BIGDATA TEAM

MegaFon Course: Big Data

Big Data Introduction
Distributed File Systems

Dral Alexey (aadral@bigdatateam.org)
CEO at BigData Team, http://bigdatateam.org/
https://www.facebook.com/bigdatateam/

03.06.2019, Moscow, Russia

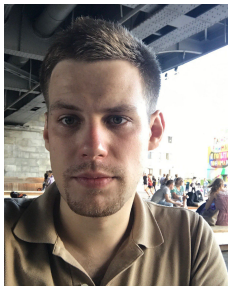
BIGDATA TEAM

Course Team

Big Data Instructors

Alexey Dral

Anton Gorokhov



Artym Vybornov

TA / Administration

Victoria

Nikolay

Oleg

BIGDATA TEAM

About Myself

- AESC MSU
- MSU
- YSDA
- Rambler
- Yandex
- Amazon AWS
- MIPT
- Sberbank
- BigData Team

Course Outline

- Distributed File Systems, HDFS
- MapReduce and Optimization Techniques
- Hive and Optimization Techniques
- Spark RDD, DataFrames and its Optimizations
- RealTime: Kafka and Spark Streaming
- NoSQL with Cassandra
- Data Ingestion and Layout Optimization
- Spark ML

BIGDATA TEAM

Learning Artefacts

The 70-20-10 rule was developed by Morgan McCall, Robert W. Eichinger and Michael M. Lombardo at the Center for Creative Leadership.

- 12 домашних заданий
- 2 тестирования

Материалы: http://rebrand.ly/mf2019q2_gdrive_public

BIGDATA TEAM

Где мы сейчас?

HDFS	HDFS	MR	MR	MR	MR	Hive	Hive	Hive	Hive	Spark	Spark	Spark	Spark	RT	RT	RT	RT	NoSQL	NoSQL
0.75	0.4	0	1	0	0.8	0	0	0	1	1	1	0	0.75	0	0	0	0	0	0
0	0	0	0.75	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0.6	0	1	0	0.8	0	0	1	1	0	1	0	1	1	0	0	0	0	0
0	0	0	0.75	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
0.75	0.4	0	1	0	0.8	0	0.5	0	1	0	0	0	1	0.67	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0.75	0.6	0	0.75	1	0.8	0	0	0	1	0	1	1	0.75	1	0	1	0.33	0	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0.75	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0.75	0	1	1	0	1	1	1	0	0	0.75	0	0	0	0	0	0
1	0	0	1	0	0	0	0	1	0	1	1	0	0	0	1	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0.75	0.4	0	1	0	0.6	1	1	0	1	1	1	0	1	0.67	0	0	0	0	0
1	0	0	1	0	1	0	0.75	1	1	0	1	0	0	0	0	0	0	0	0

BIGDATA TEAM

Где мы сейчас?

MF-2019-Q2	before	after	gain	support	<hidden>	before	after	gain	support
HDFS	28%			2	HDFS	42%	80%	38%	1
MR	28%			4	MR	35%	79%	44%	4
Hive	27%			4	Hive	30%	71%	41%	4
Spark	30%			4	Spark	38%	63%	25%	5
RT	13%			4	RT	21%	68%	47%	4
NoSQL	4%			2	NoSQL	17%	76%	59%	2
avg-score max: 20	4.6			20	avg-score max: 20	6.1	14.2	8.1	20
	23%	0%	0%			31%	71%	40%	

Outline

- Big Data: Motivation
- Distributed File System
- HDFS Workshop

Big Data: Motivation

Big Data: Motivation

Data Scientist: The Sexiest Job of the 21st Century

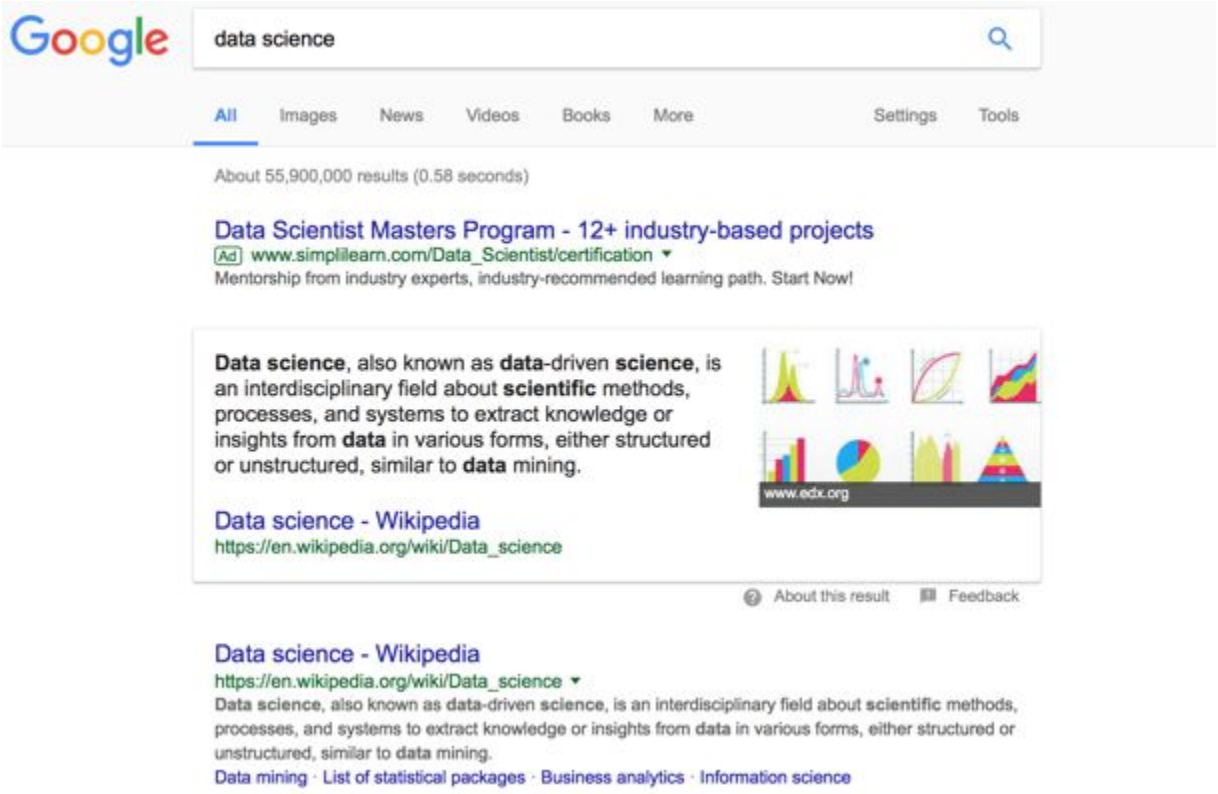
by Thomas H. Davenport and D.J. Patil

FROM THE OCTOBER 2012 ISSUE

https://hbr.org/2012/10/data-scientist-the-sexiest-job-of-the-21st-century

Откуда берутся (большие) **данные**?

Search Engines



Social Networks

Twitter, Inc.

VKontakte

Instagram

LinkedIn

facebook

...

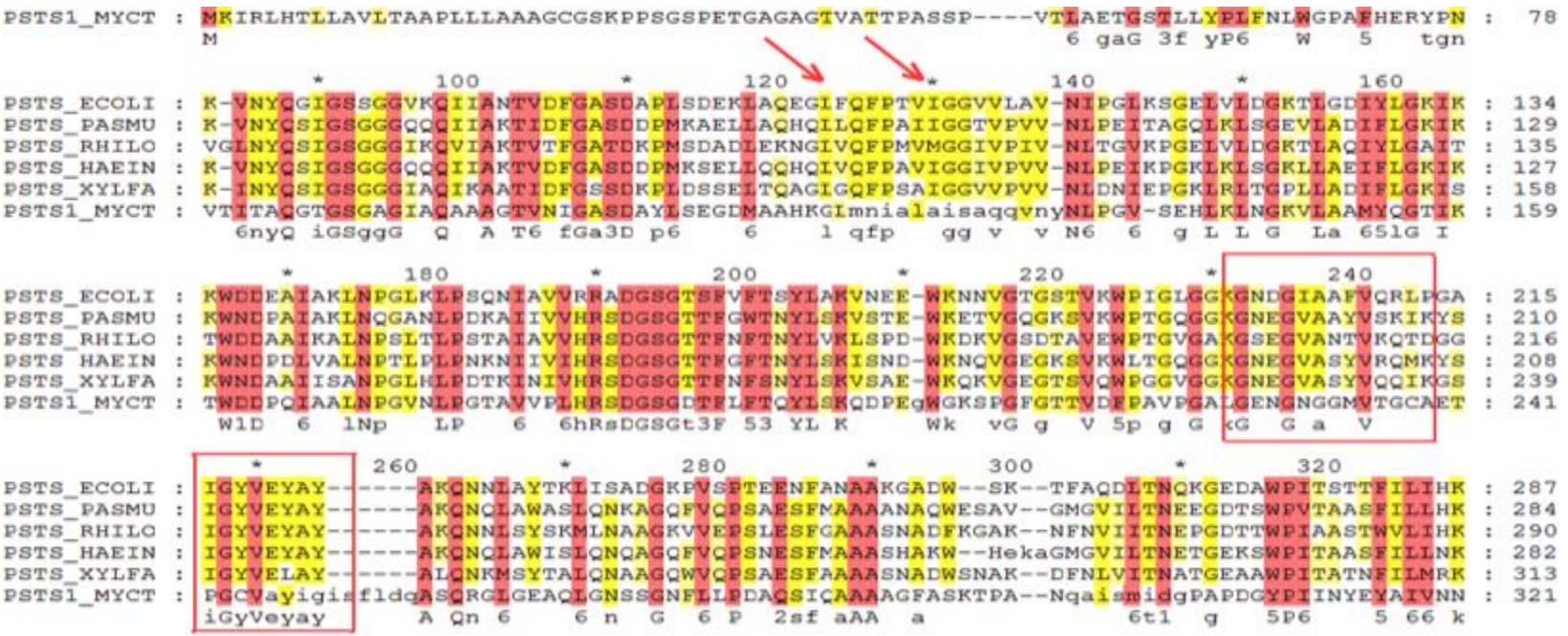
BIGDATA TEAM

Transactions

Telecom Data

BIGDATA TEAM

Bioinformatics

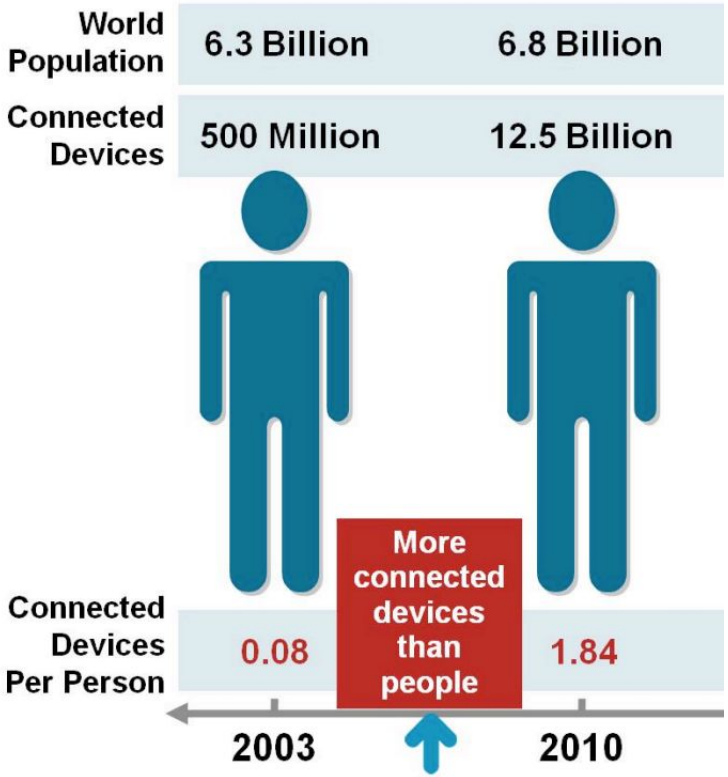


BIGDATA TEAM

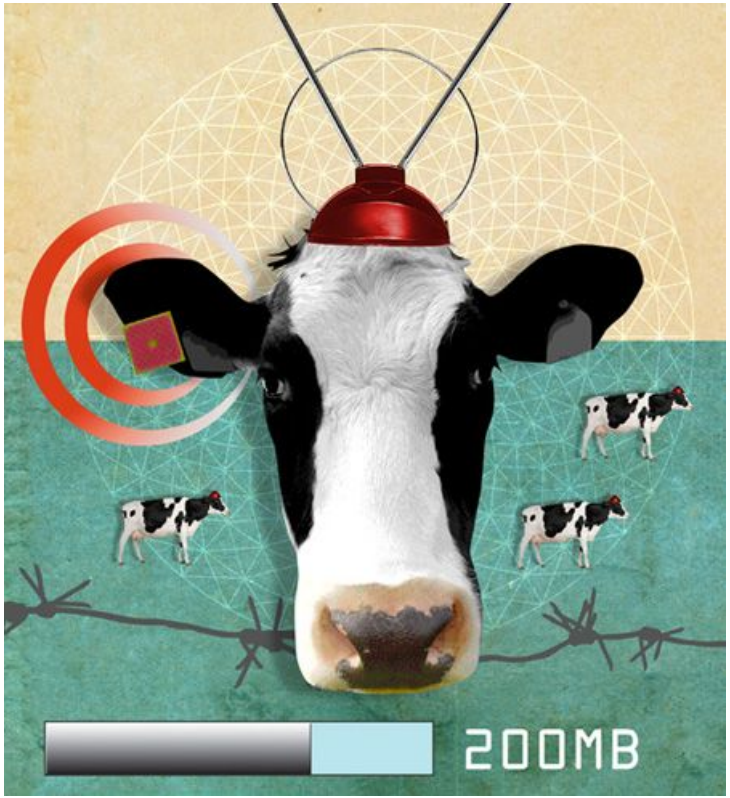
Internet of Things (IoT)

Q&A: IoT vs Human Beings?

Internet of Things (IoT)



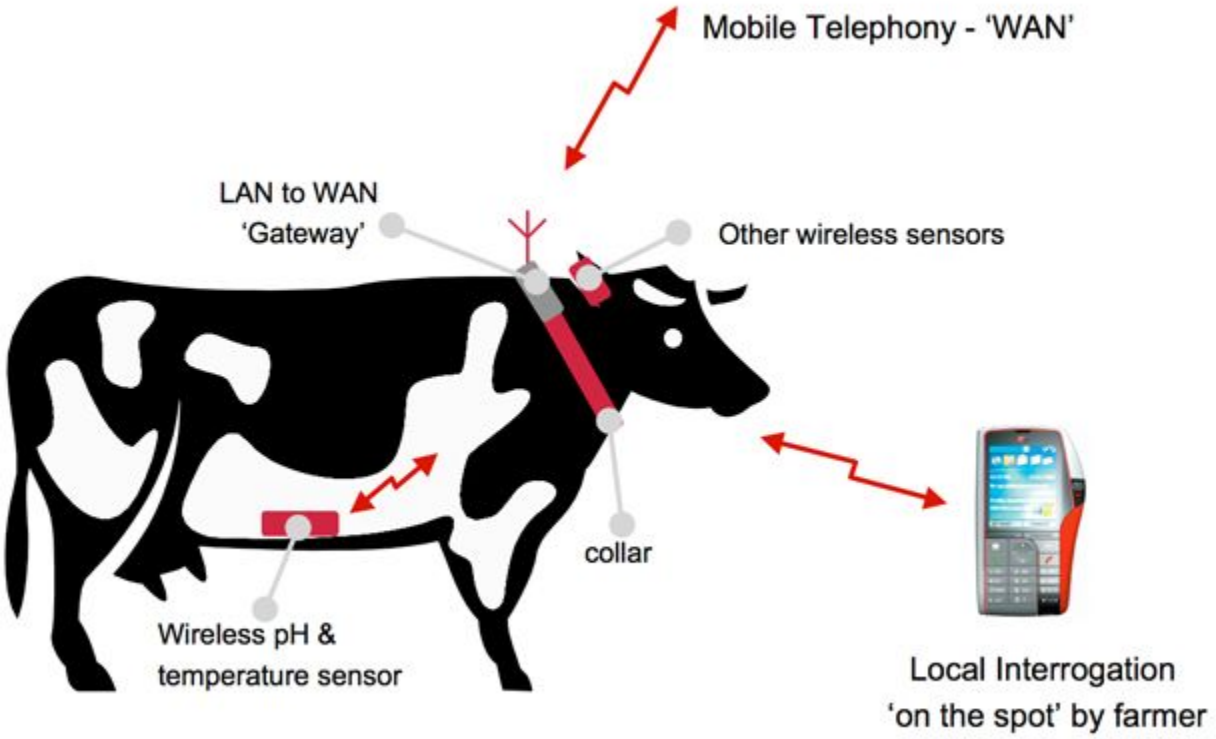
Source: Cisco IBSG, April 2011



Source: The Economist, 2010

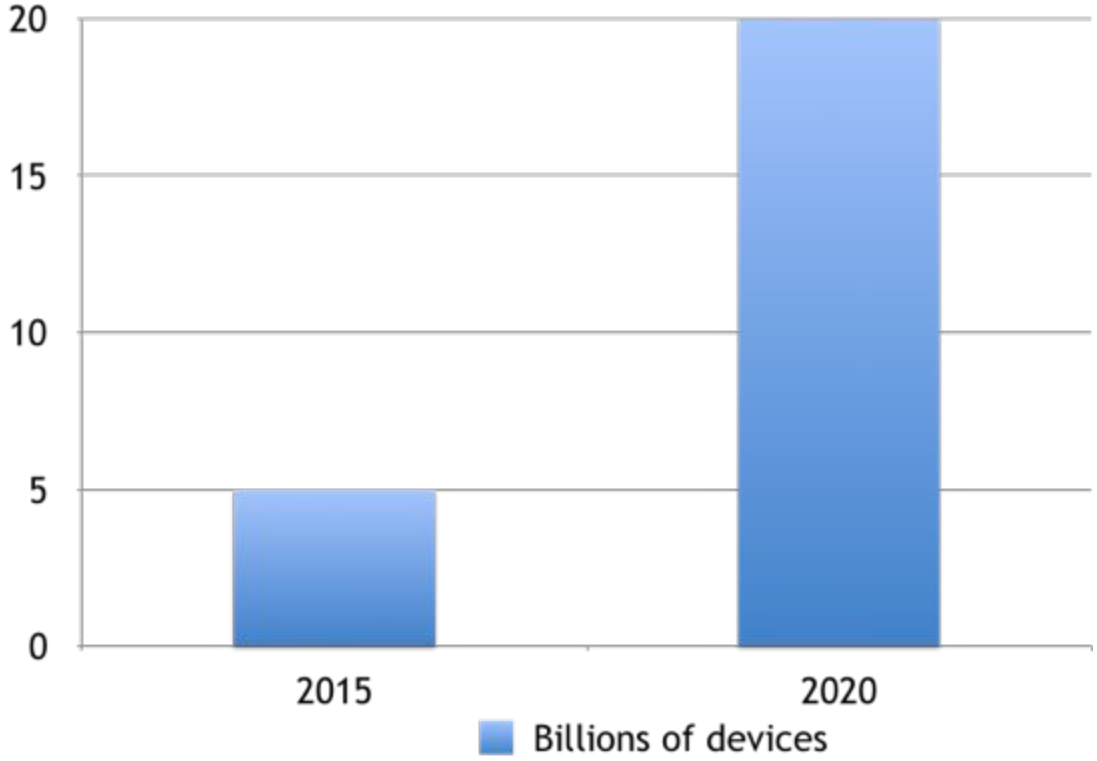
BIGDATA TEAM

Internet Cattle



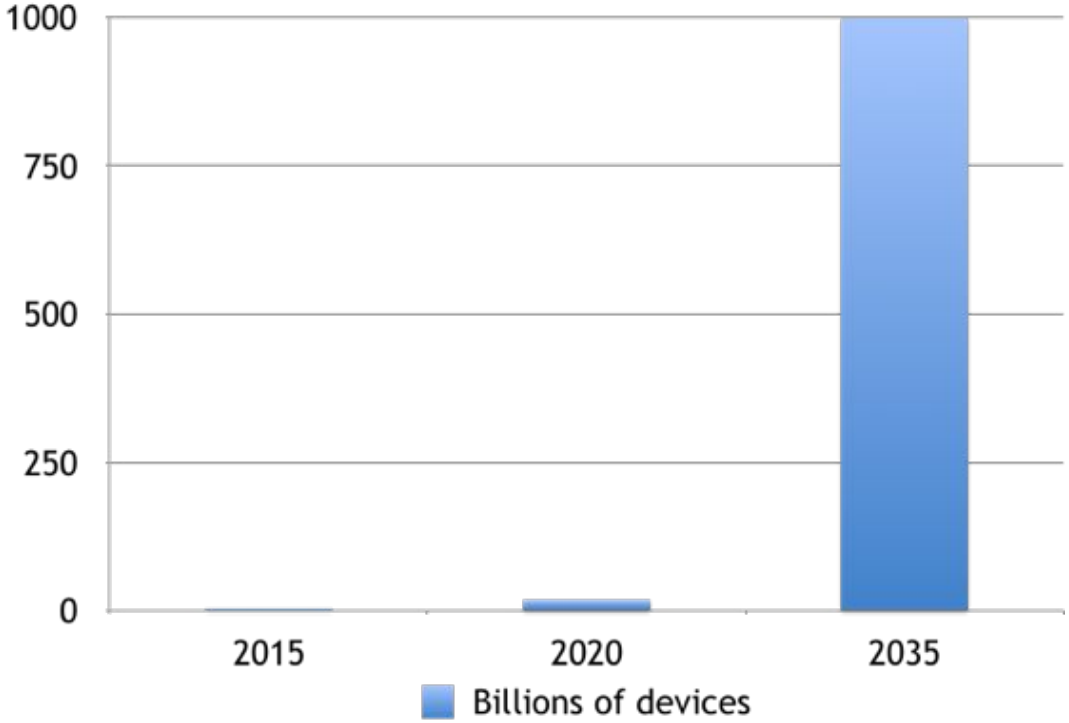
https://www.theengineer.co.uk/internet-cattle/ Scotland (16.08.2017)

Internet of Things



Gartner

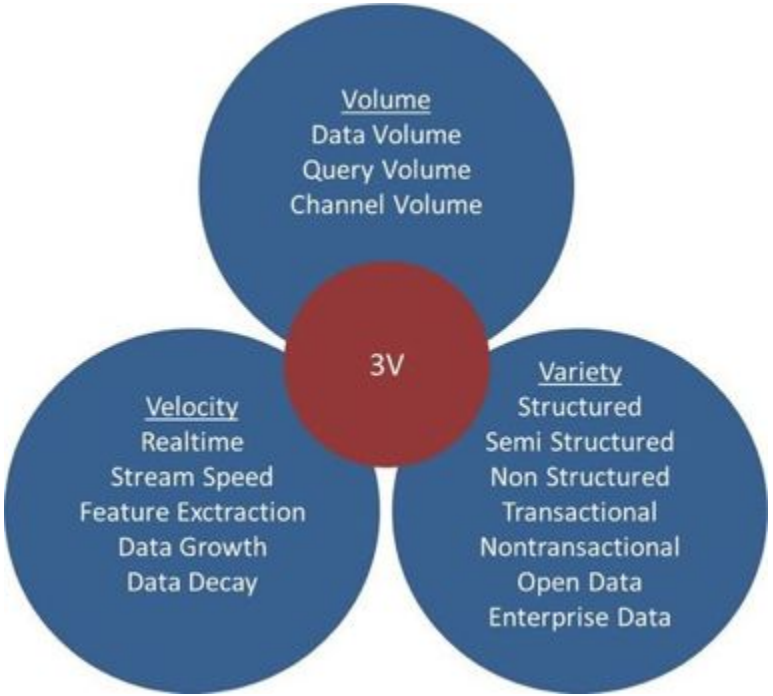
Internet of Things

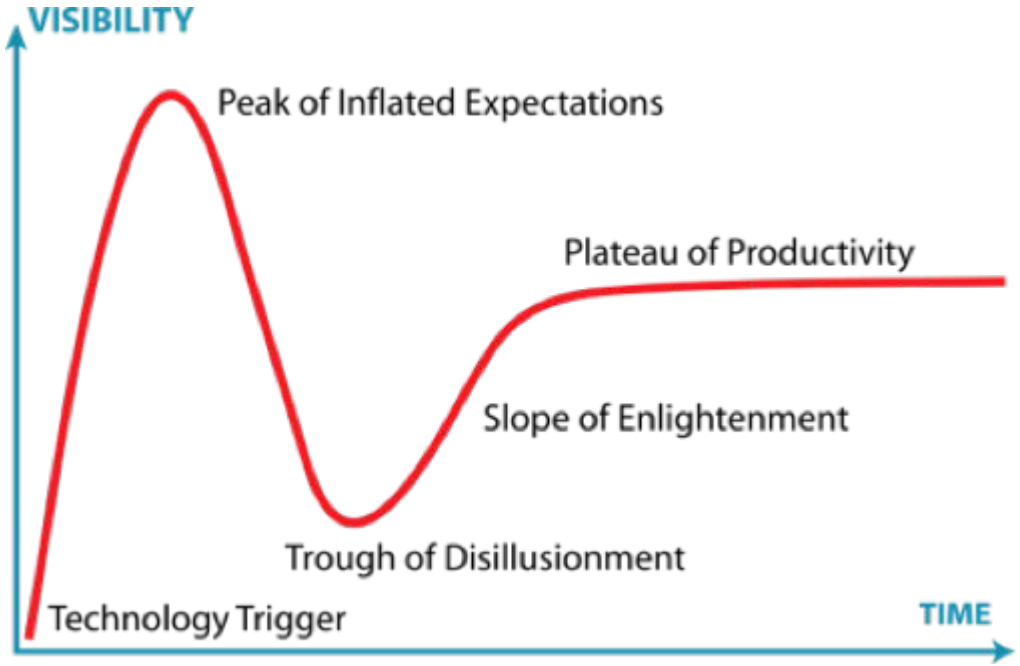


Gartner

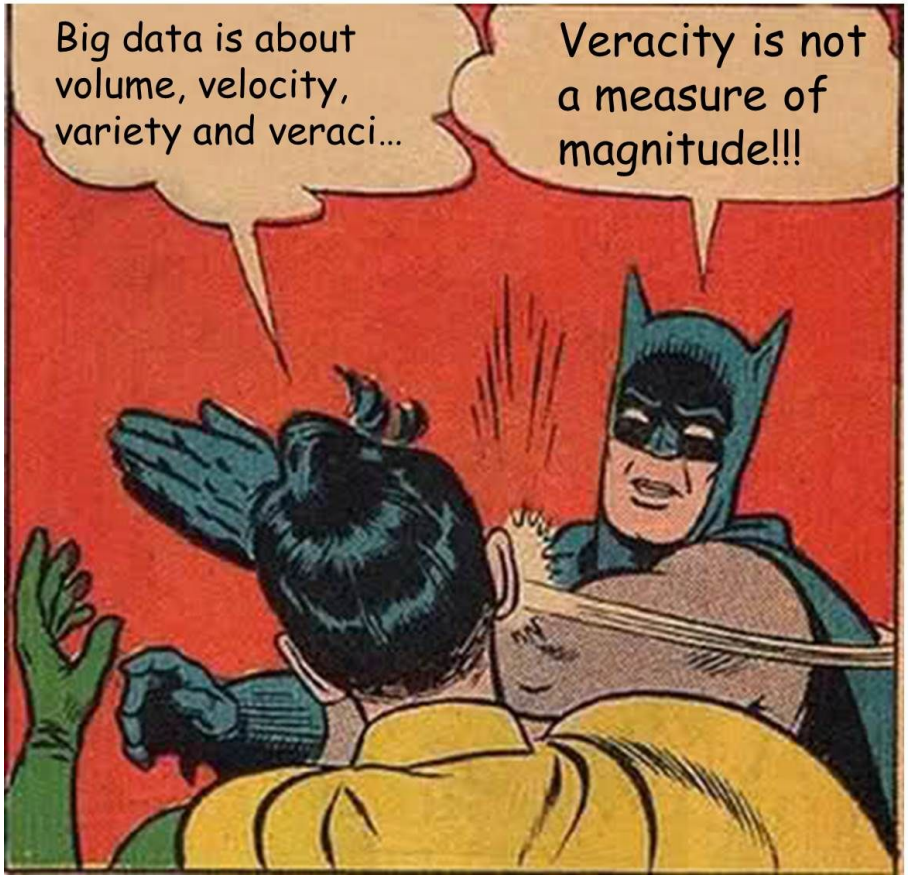
Q&A: Что такое Big Data?

Big and Small Data





4+ Vs of Big Data

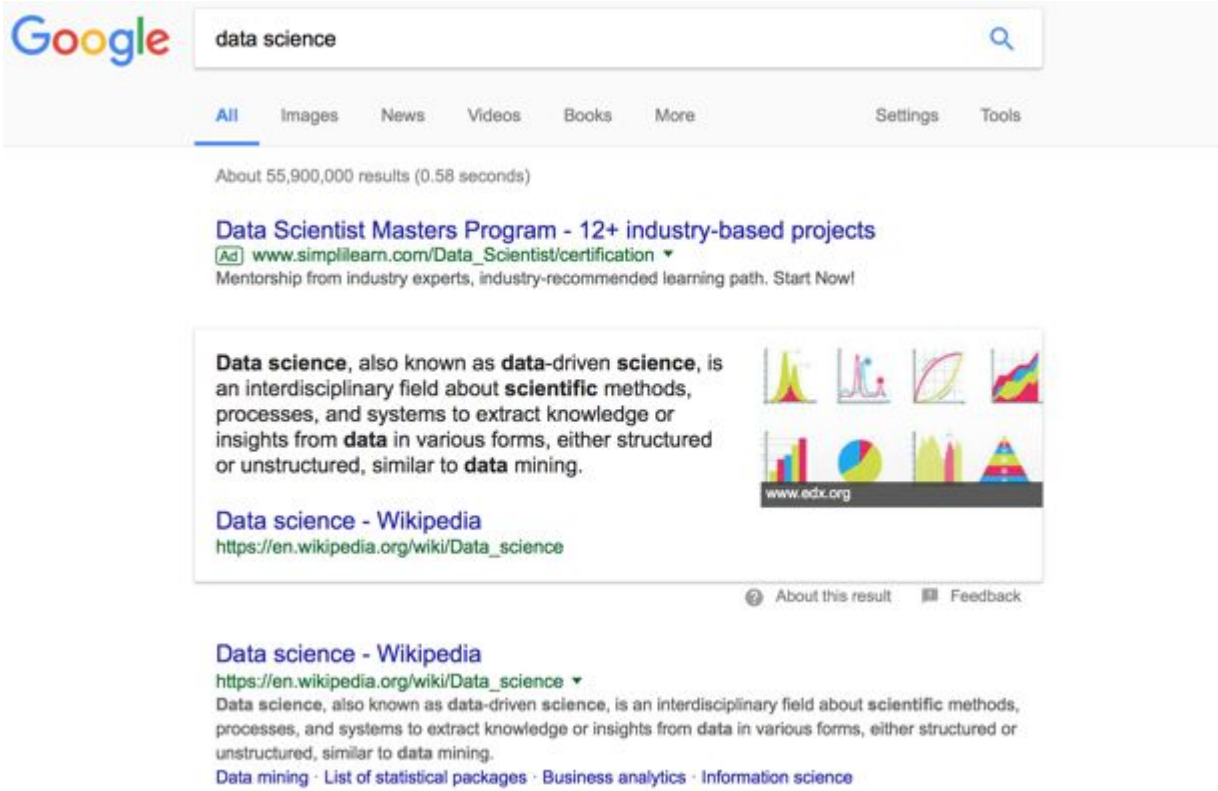


https://blogs.gartner.com/doug-laney/batman-on-big-data/

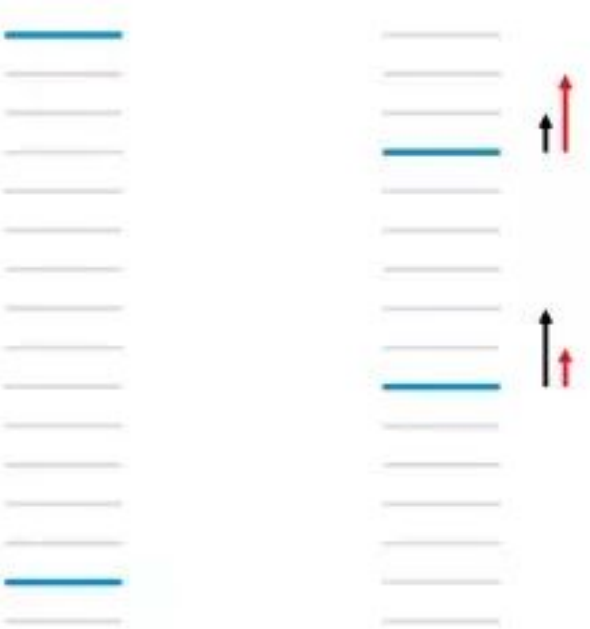
Где применяется (data) **science**?

Why Big Data?

Search Optimization



Learning to Rank



University of Illinois at Urbana-Champaign

ILLINOIS

Text Retrieval and Search Engines

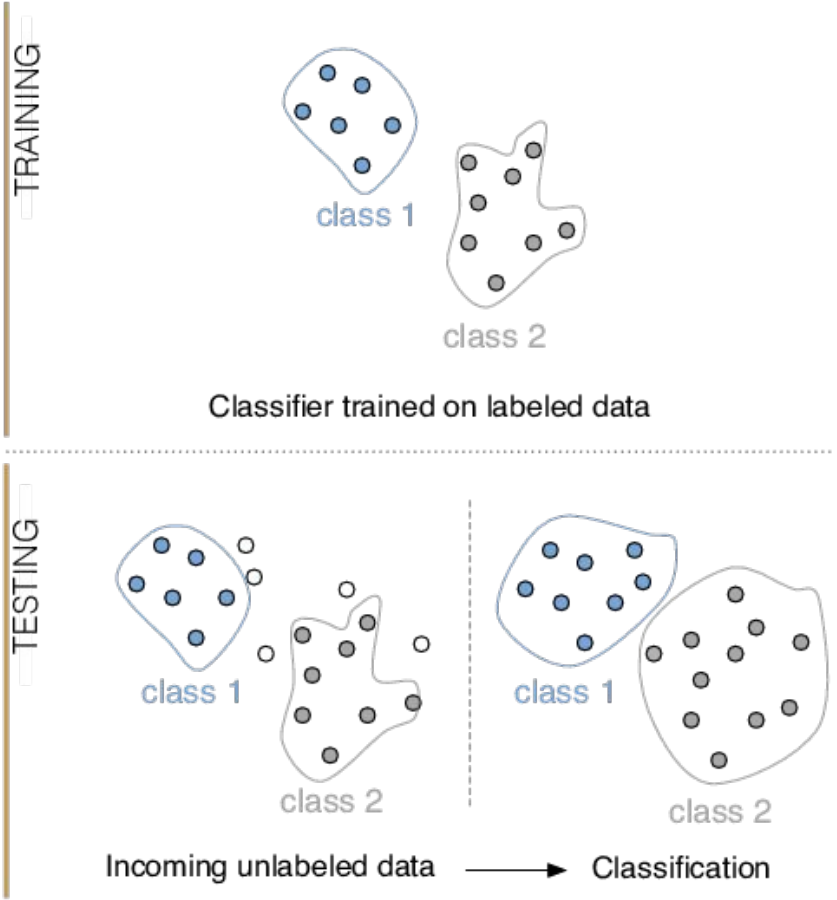
358 ratings

Course 2 of 6 in the Specialization Data Mining

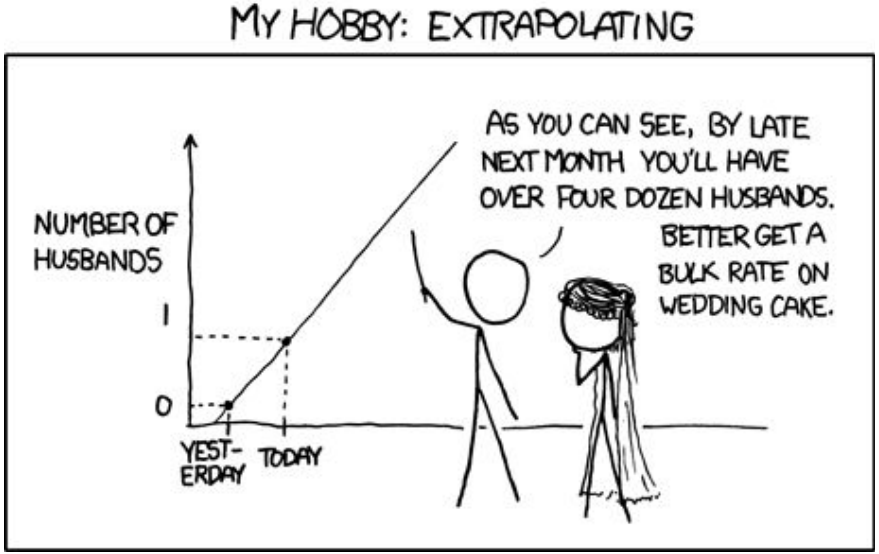
https://en.wikipedia.org/wiki/Learning_to_rank

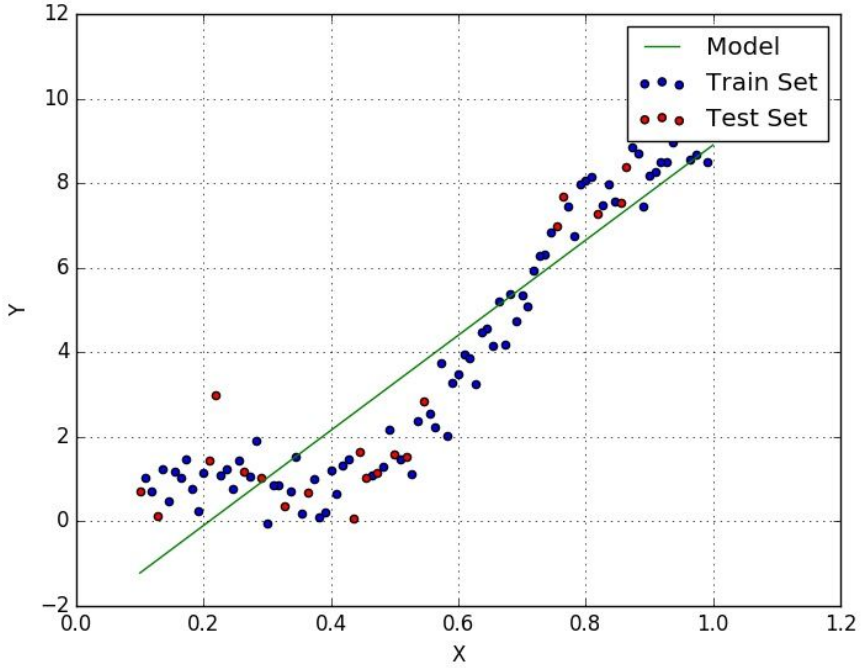
Credit Scoring

Classification



Regression


MY HOBBY: EXTRAPOLATING
NUMBER OF HUSBANDS
1
0
YEST-ERDAY
TODAY
AS YOU CAN SEE, BY LATE NEXT MONTH YOU'LL HAVE OVER FOUR DOZEN HUSBANDS.
BETTER GET A BULK RATE ON WEDDING CAKE.


Model
Train Set
Test Set
Y
X
12
10
8
6
4
2
0
−2
0.0
0.2
0.4
0.6
0.8
1.0
1.2

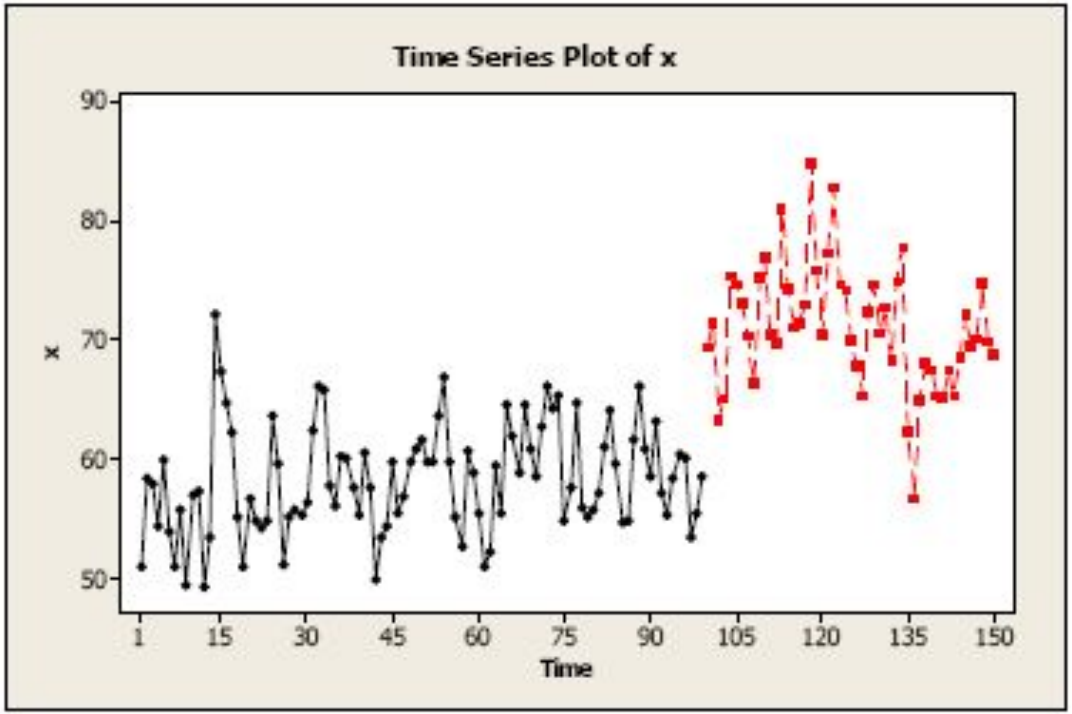
Churn Prediction

BIGDATA TEAM

Demand Forecast

BIGDATA TEAM

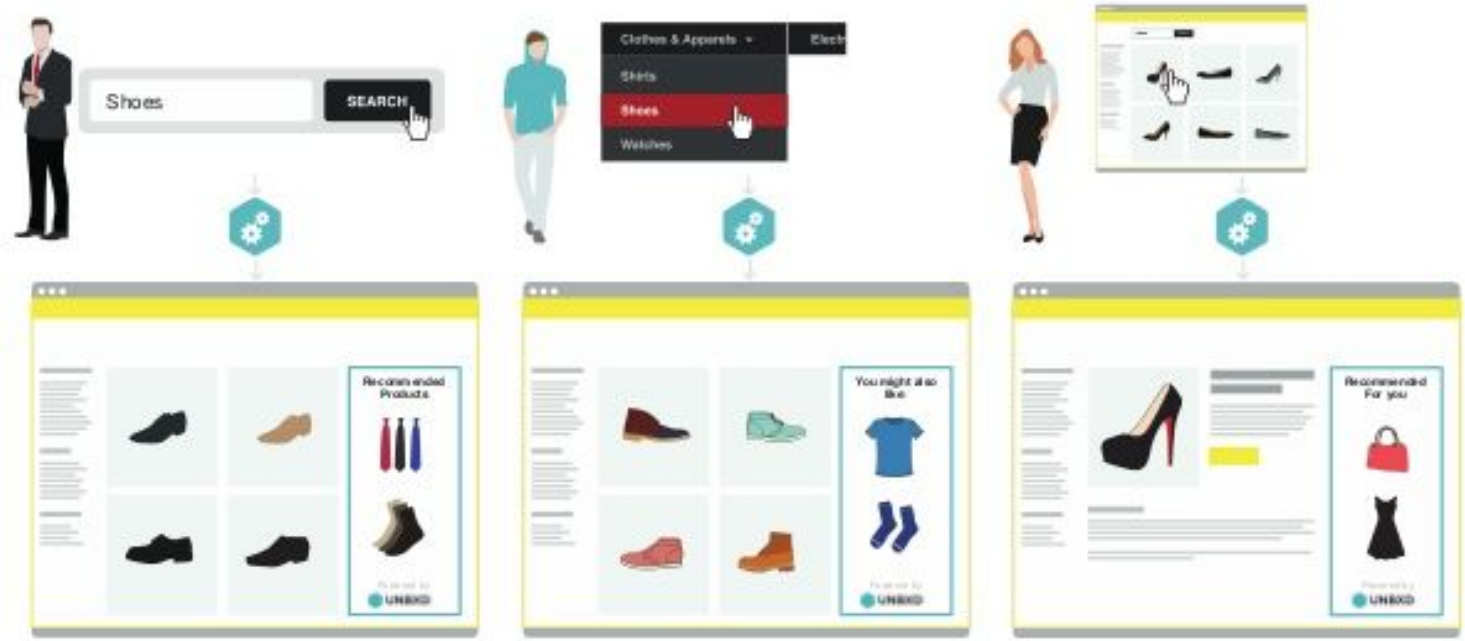
Time Series Analysis



https://en.wikipedia.org/wiki/Time_series

BIGDATA TEAM

Personal Recommendations



Recommender Systems

NETFLIX

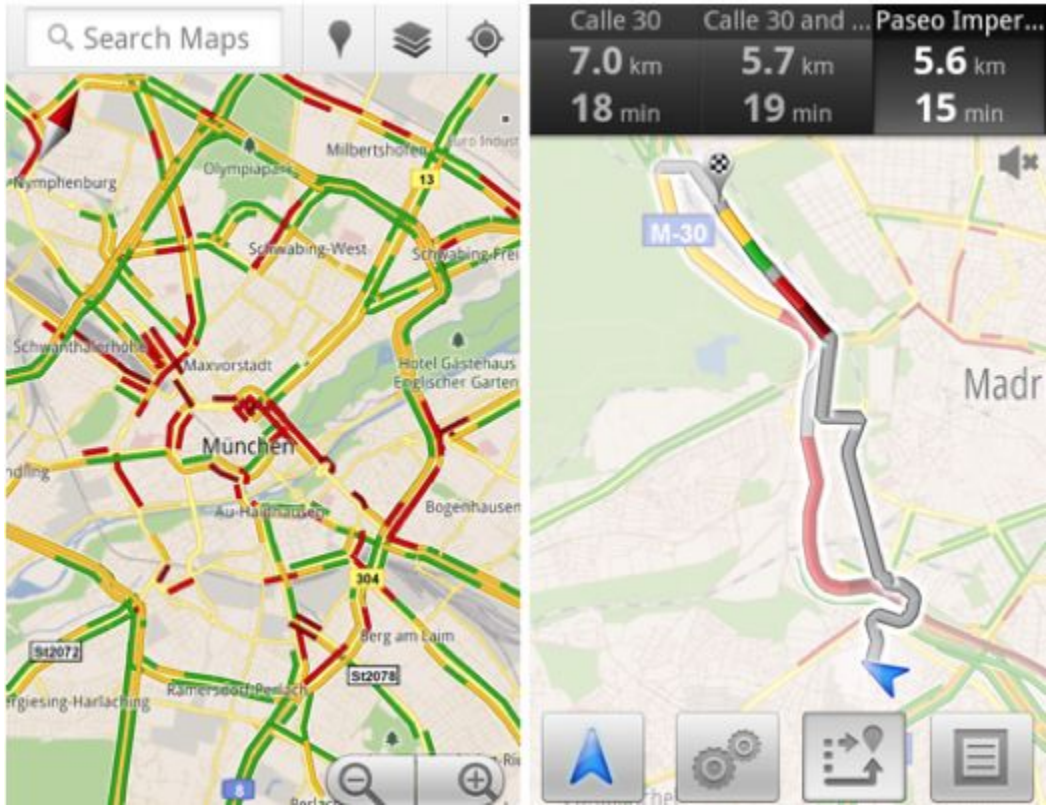
BIGDATA TEAM

Manufacturing

https://yandex.ru/blog/company/stal-neft-i-iskusstvennyy-intellekt-yandex-data-factory-o-novoy-promyshlennoy-revolyutsii

BIGDATA TEAM

Traffic Jam Prediction & Routing



BIGDATA TEAM

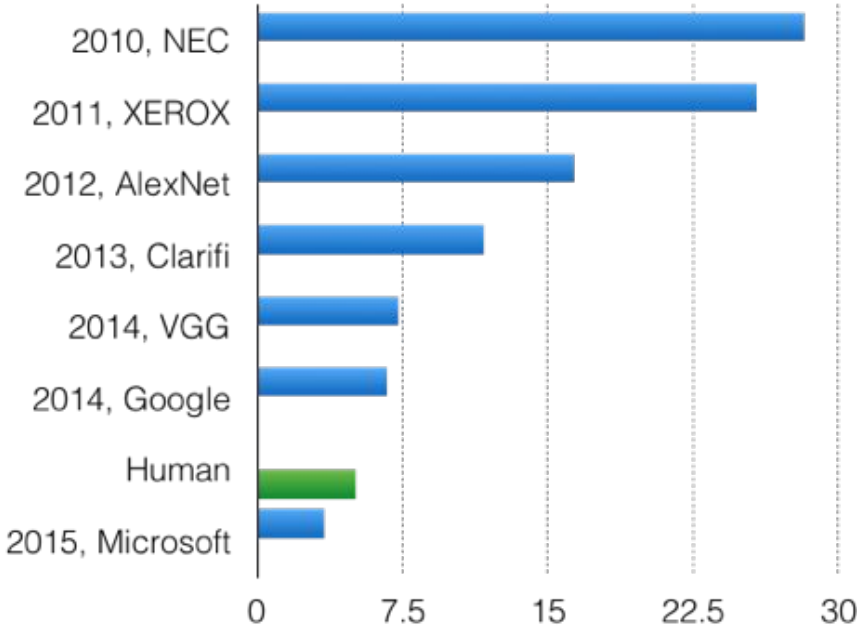
Self-Driving Cars

BIGDATA TEAM

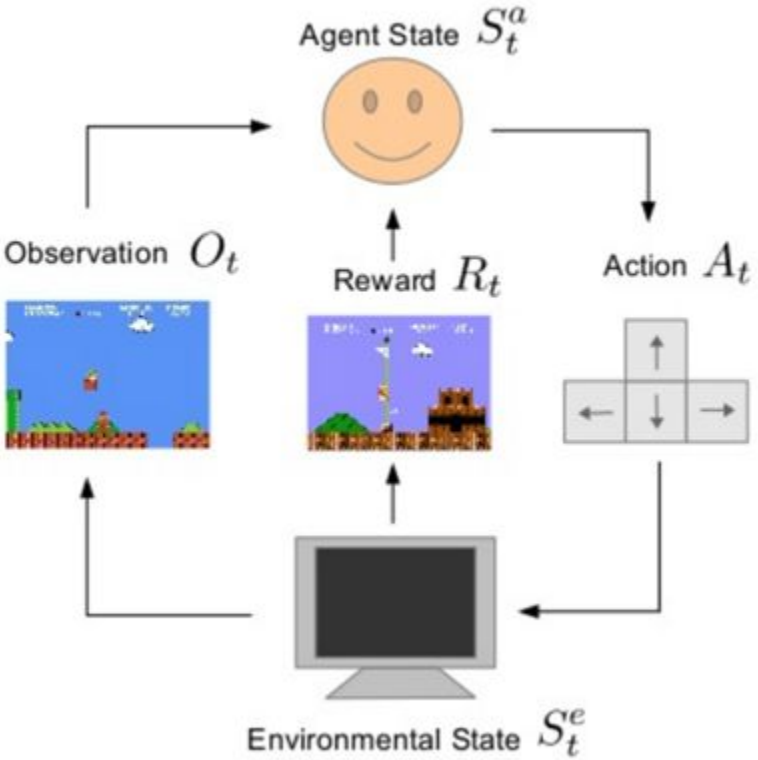
Urban Air-Transport

BIGDATA TEAM

Image Recognition: ImageNet



Reinforcement Learning

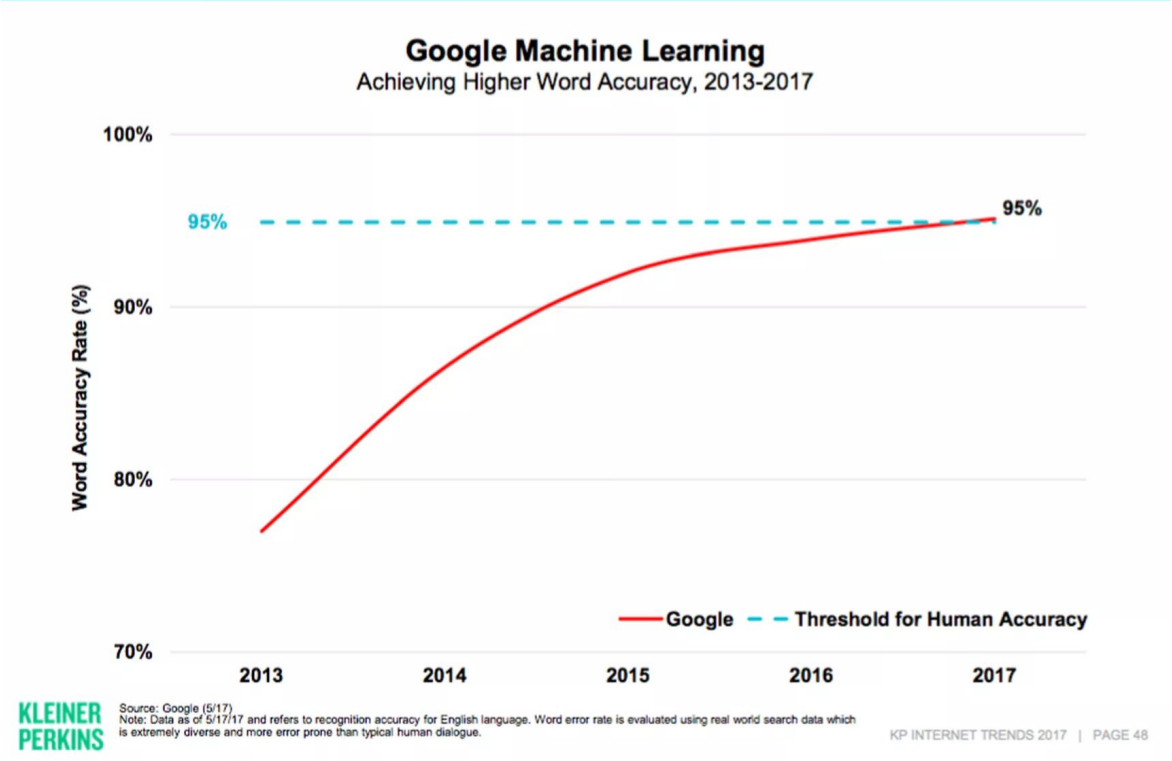


https://en.wikipedia.org/wiki/Reinforcement_learning

Data Science for People

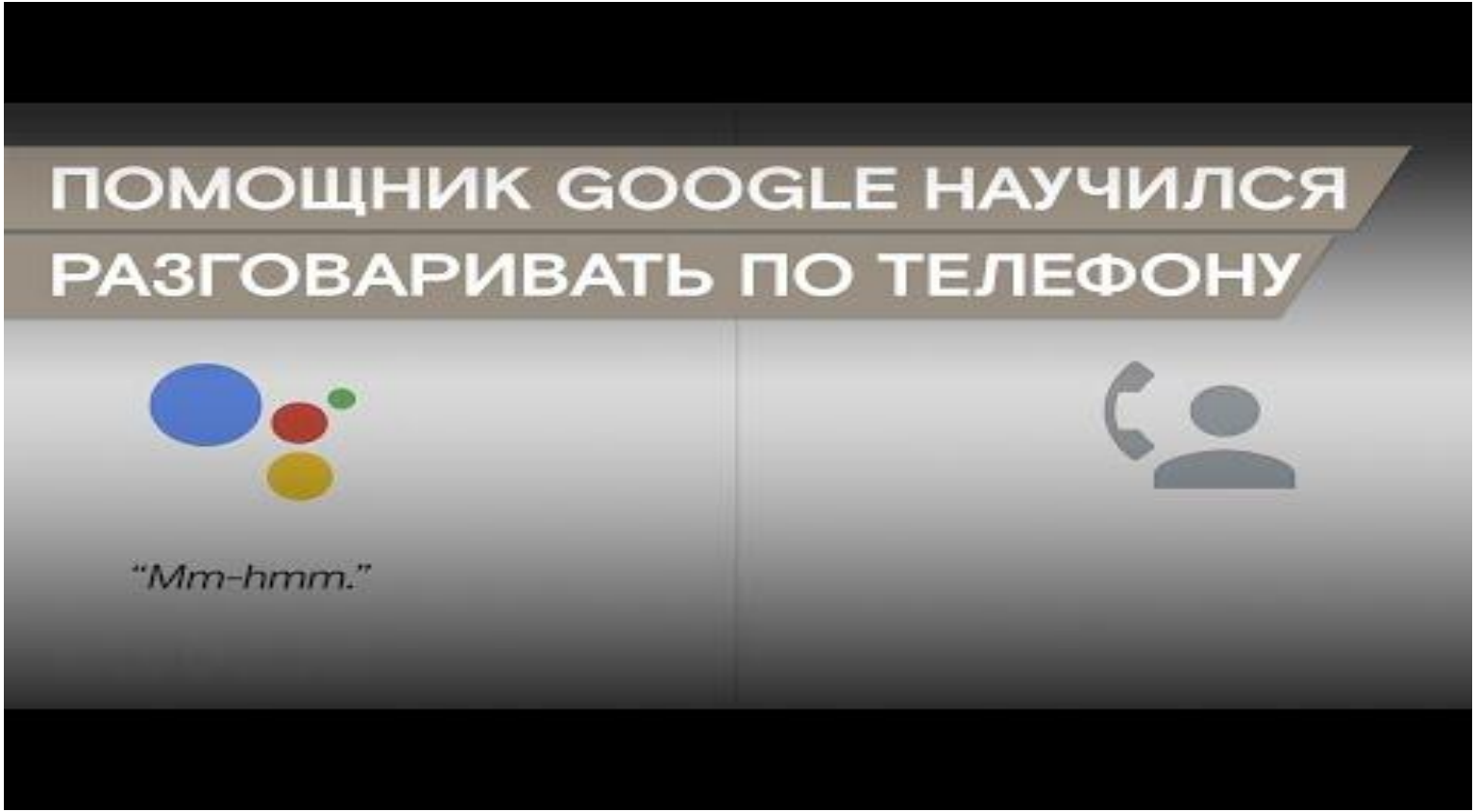
Voice Recognition

…Voice-Based Platform *Back-Ends* = Voice Recognition Accuracy Continues to Improve



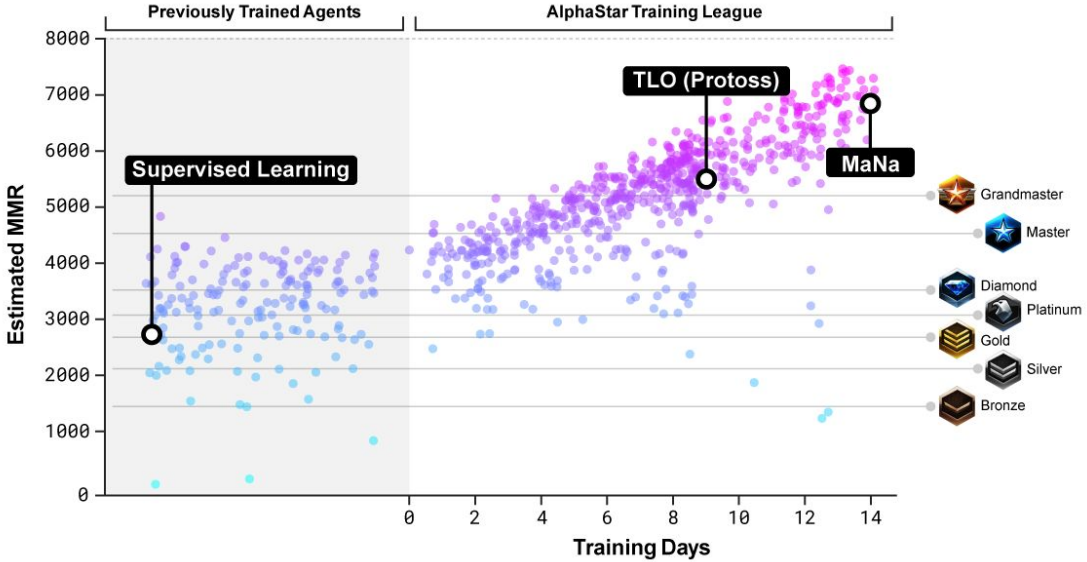
BIGDATA TEAM

Google Duplex (8.05.2018)



BIGDATA TEAM

DeepMind, AlfaStar (24.01.2019)



- Training - 14 days, 16 TPUs / agent
- 200 years of real-time StarCraft play / agent
- Technical description is being prepared for publication

https://deepmind.com/blog/alphastar-mastering-real-time-strategy-game-starcraft-ii/

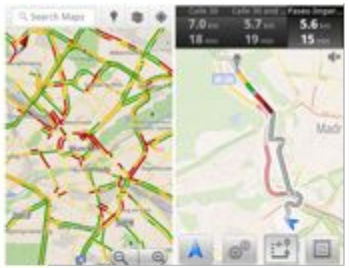
BIGDATA TEAM

Agriculture

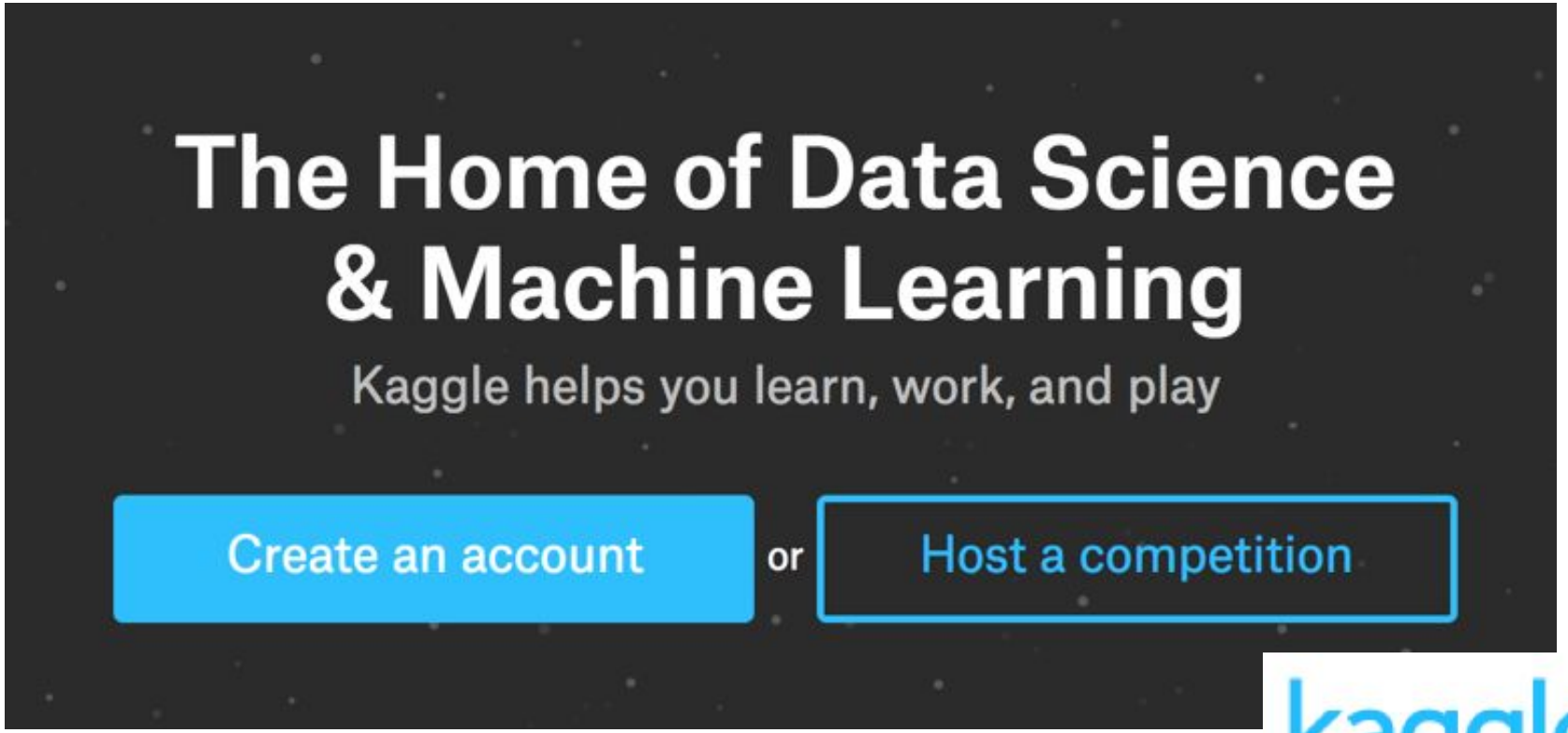
http://aggeek.net/ru/technology/id/top-5-naibolee-mnogoobeschajuschih-tehnologij-v-smart-agriculture-431/

BIGDATA TEAM

Data Science for Business



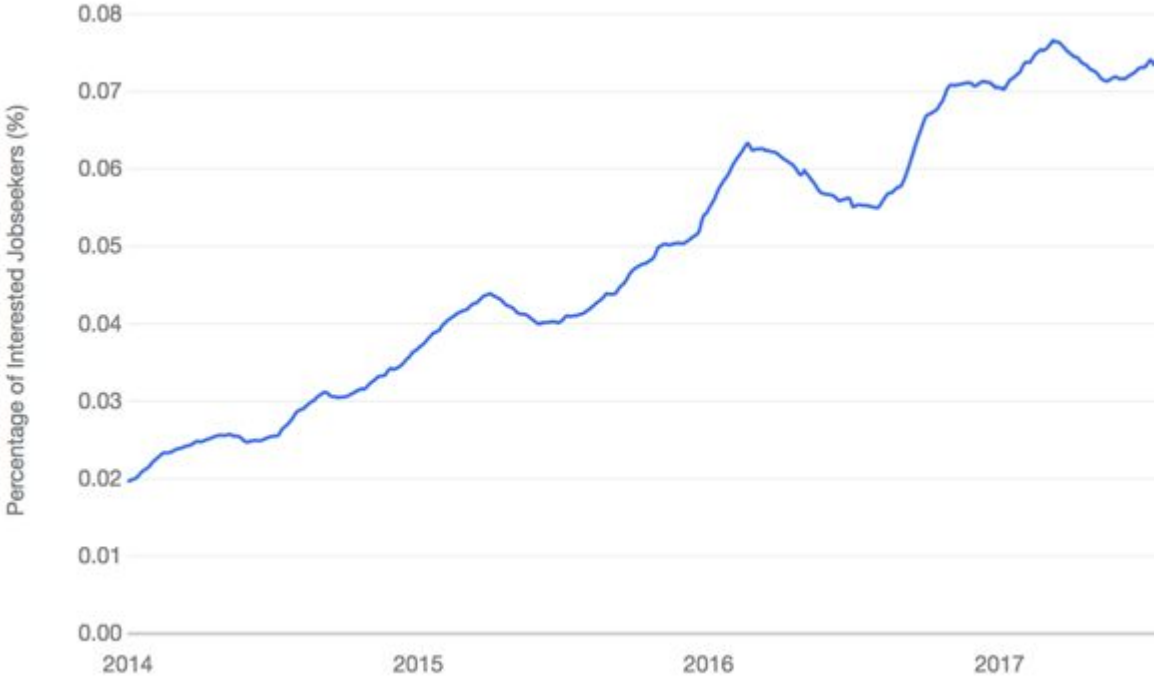
Data Science: Outsourcing



kaggle

Big Data: Motivation, v2

Data Science Job Index



indeed®

Data Scientist? Data Engineer? Data ...?

indeed

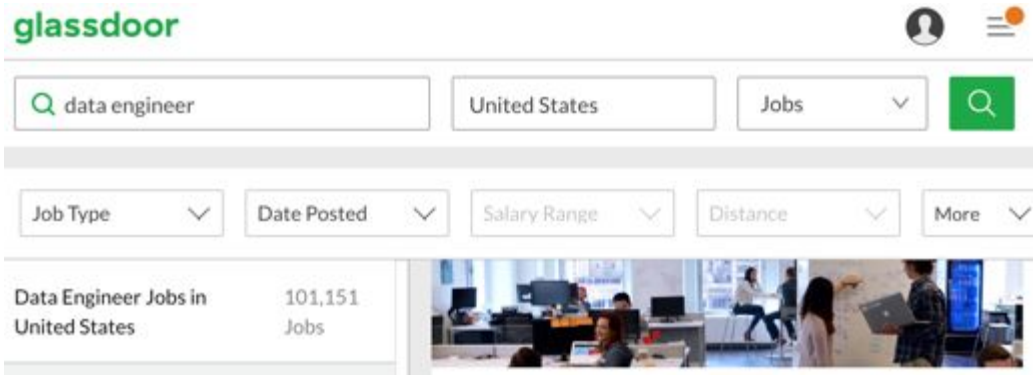
Index jobs

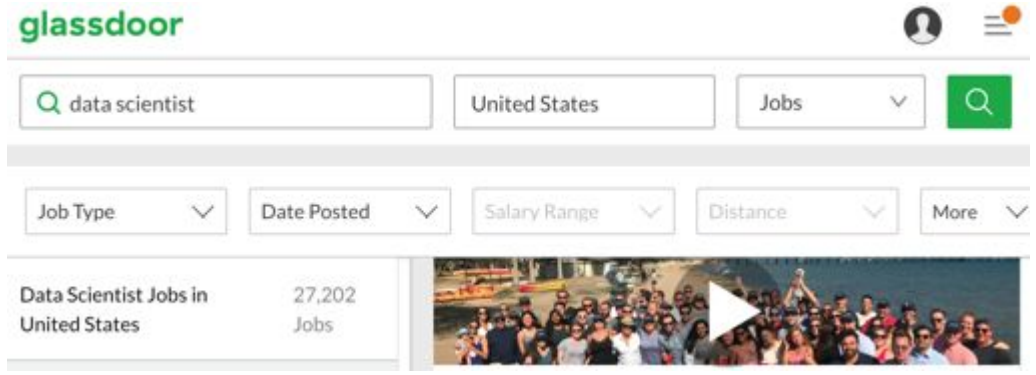
My recent searches

data engineer - 34,228 new

data scientist - 9,279 new

4x





BIGDATA TEAM

Coursera Specialization on Big Data

Capstone Project

Hadoop, Spark **Hive, Spark** **Spark ML** **Real Time**

Yandex https://www.coursera.org/specializations/big-data-engineering

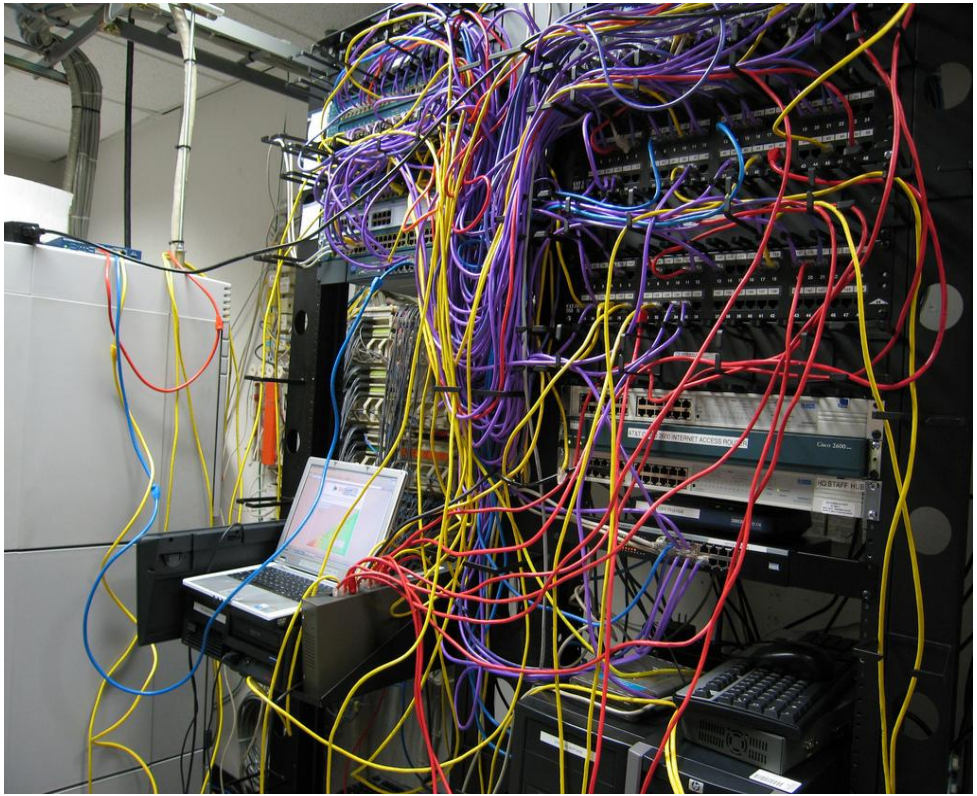
Telegram Channel

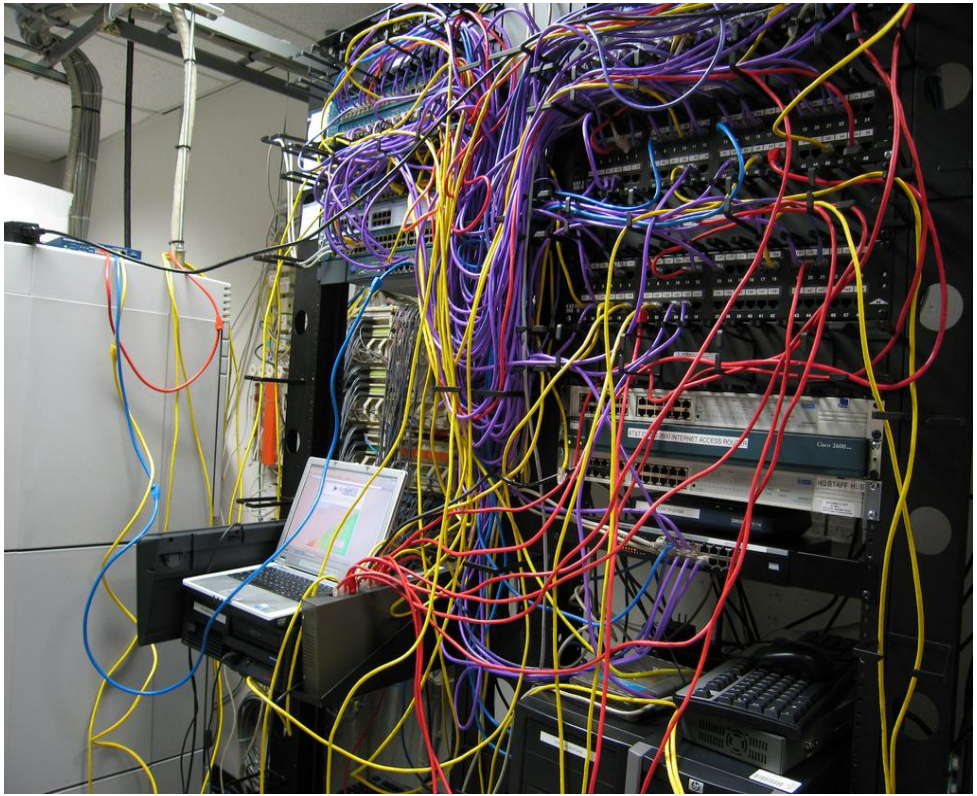
http://rebrand.ly/mf2019q2_telegram_join

Tea / Coffee Break

Distributed File Systems

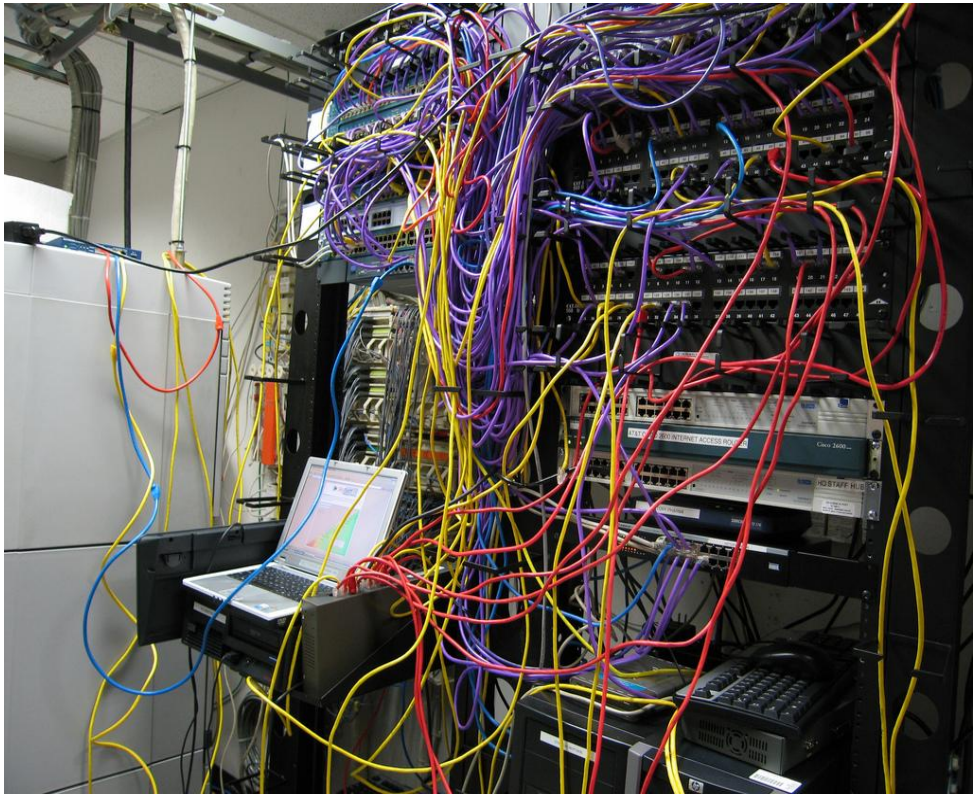
Q: what can break?





Node Failures

- Fail-Stop
- Fail-Recovery
- Byzantine



Link Failures

- Perfect Link
- Fair-Loss Link
- Byzantine

(a) Synchronous Model

- Clock Drift
- Clock Skew

Multiprocess Computational Systems

- Parallel Computing
- Distributed Computing
- Grid Computing

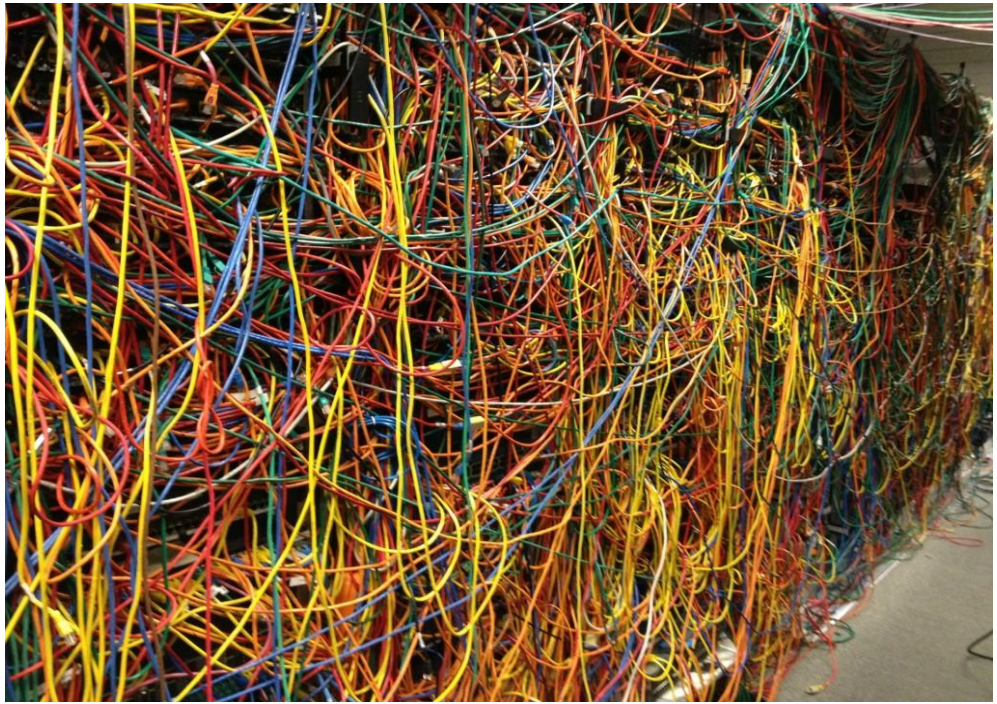
Parallel Computing

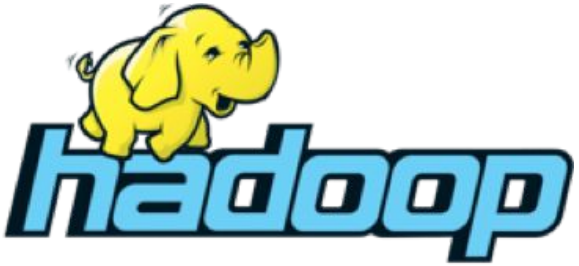
OpenMP

MPI

Fail-Stop + Perfect Link + Synchronous

Distributed Computing



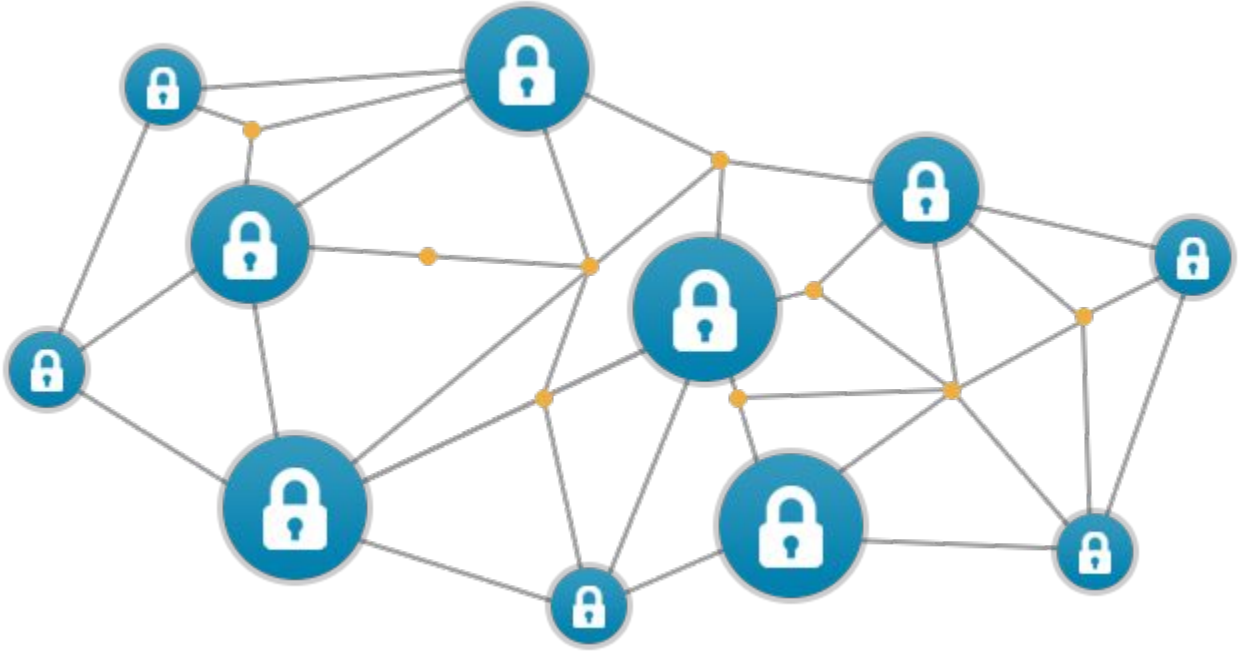


Fail-Recovery + Fair-Loss Link + Asynchronous

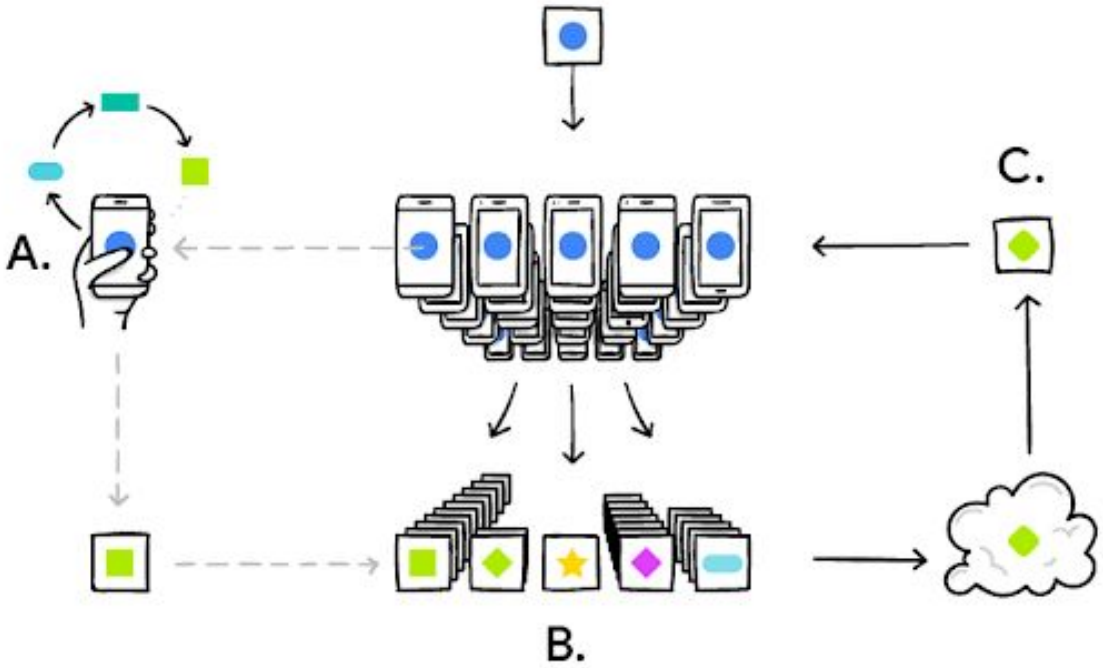
Grid Computing

Byzantine-Failure + Byzantine Link + Asynchronous

Blockchain



Federated Machine Learning



https://ai.googleblog.com/2017/04/federated-learning-collaborative.html

GFS / HDFS

Distributed File Systems

The Google File System

Sanjay Ghemawat, Howard Gobioff, and Shun-Tak Leung

Google*

ABSTRACT

We have designed and implemented the Google File System, a scalable distributed file system for large distributed data-intensive applications. It provides fault tolerance while running on inexpensive commodity hardware, and it delivers high aggregate performance to a large number of clients.

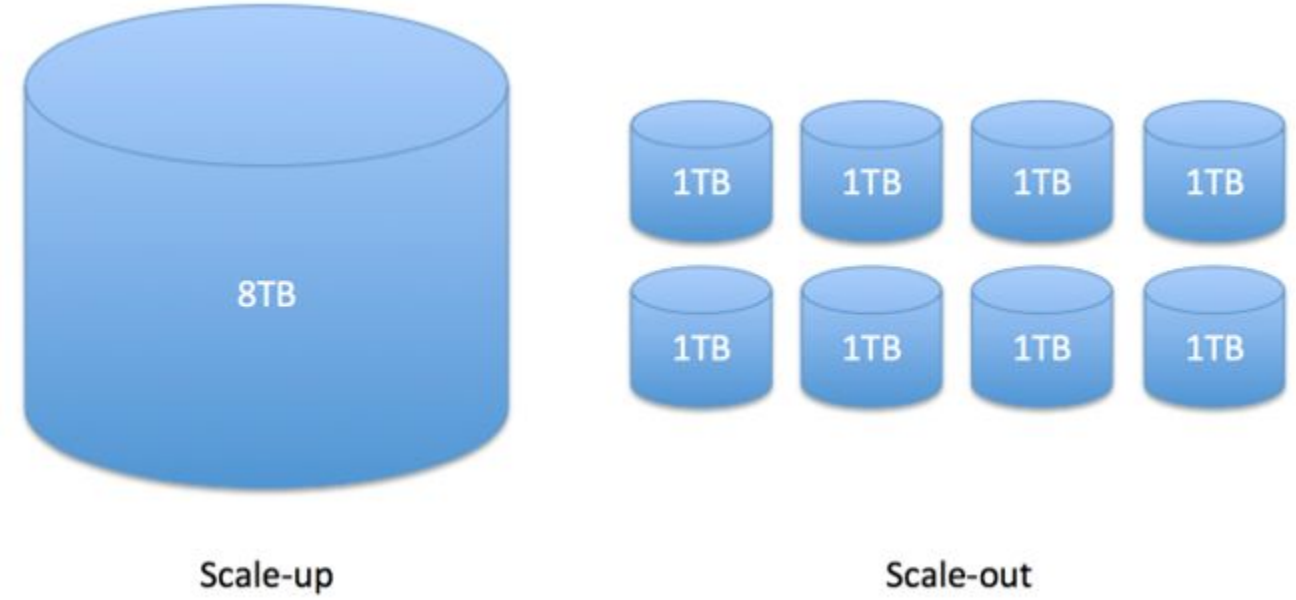
While sharing many of the same goals as previous distributed file systems, our design has been driven by observations of our application workloads and technological environment, both current and anticipated, that reflect a marked departure from some earlier file system assumptions. This has led us to reexamine traditional choices and explore rad-

1. INTRODUCTION

We have designed and implemented the Google File System (GFS) to meet the rapidly growing demands of Google's data processing needs. GFS shares many of the same goals as previous distributed file systems such as performance, scalability, reliability, and availability. However, its design has been driven by key observations of our application workloads and technological environment, both current and anticipated, that reflect a marked departure from some earlier file system design assumptions. We have reexamined traditional choices and explored radically different points in the design space.

The Google File System, Symposium on Operating Systems Principles (SOSP, 2003)

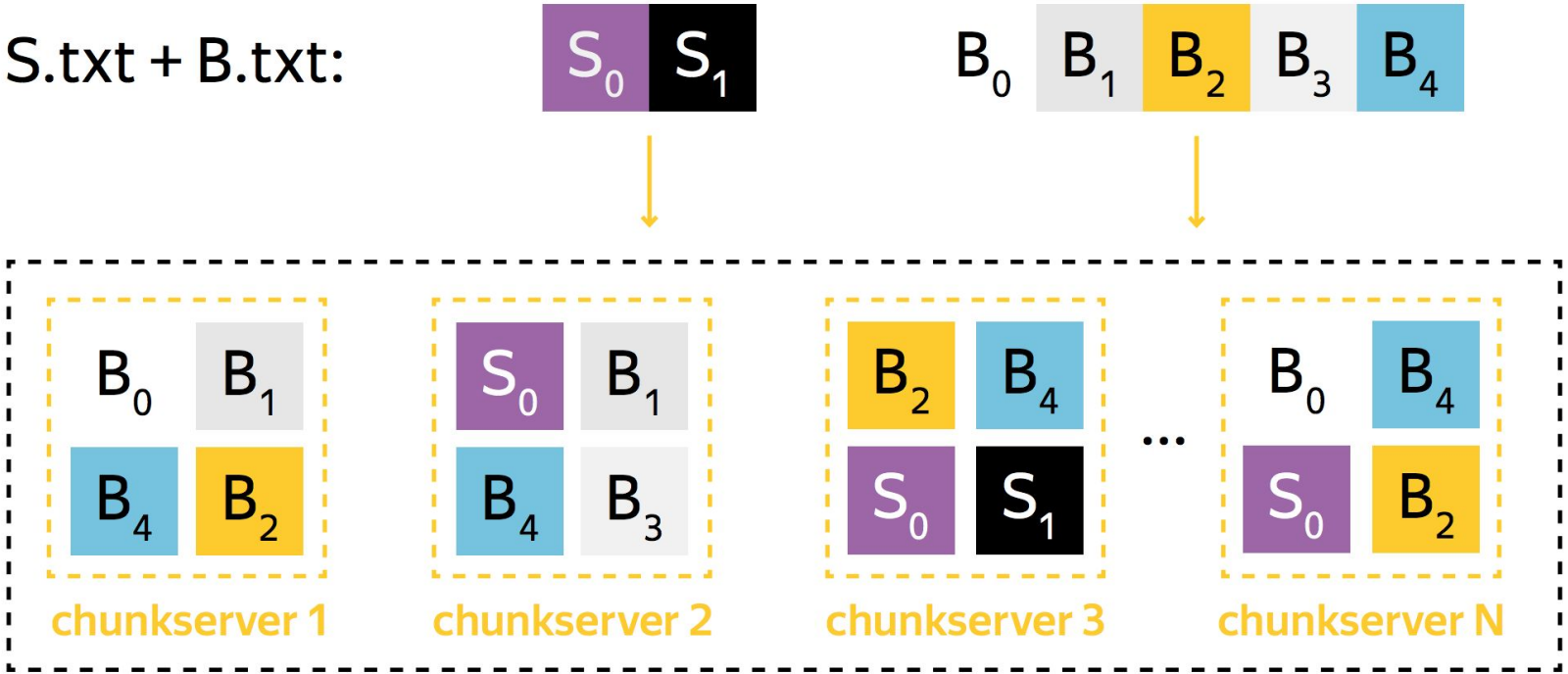
Distributed File Systems



Q&A

- Q: Как выживать в случае поломок серверов?
- Q: Как решать проблему равномерной нагрузки серверов в случае наличия файлов разных размеров (2TB vs 10GB)?

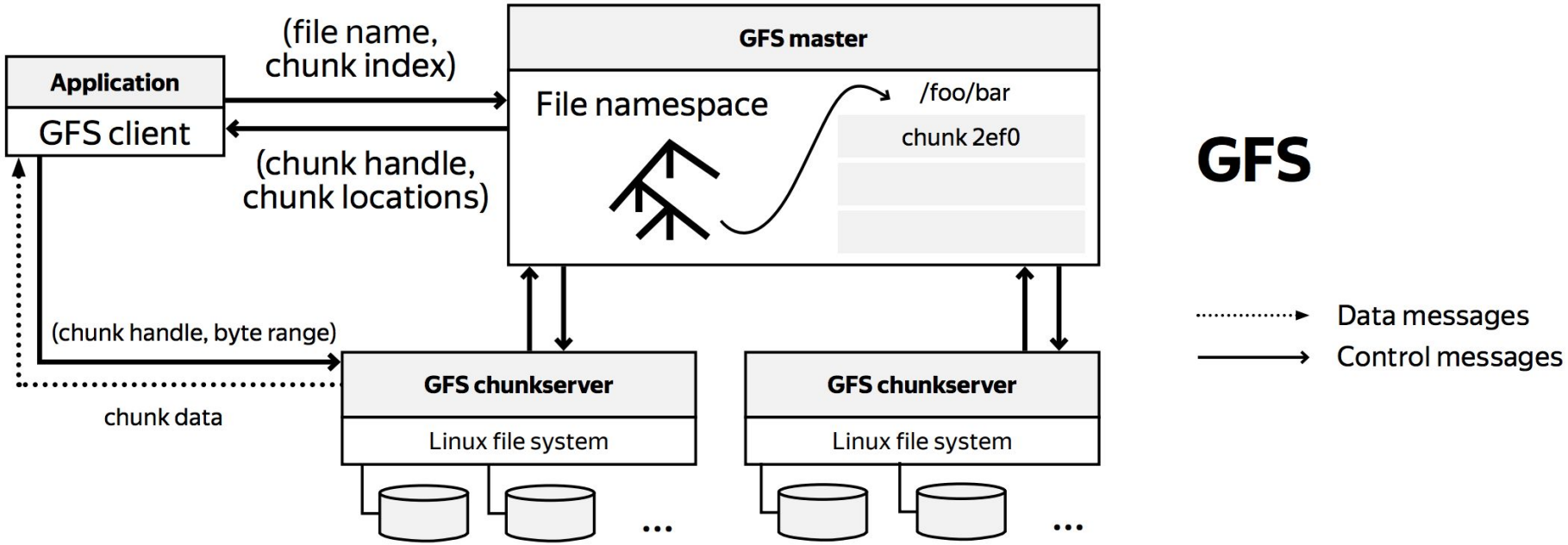
Distributed File Systems



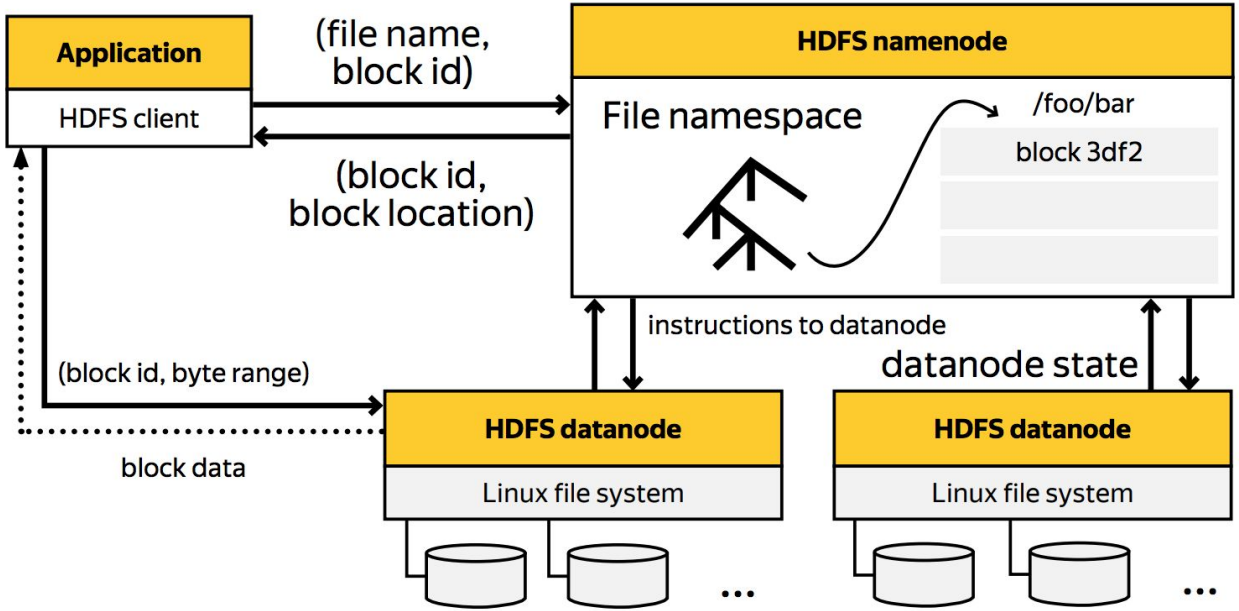
DFS Key Components

- Components failures are a norm → replication
- Even space utilization
- Write-once-read-many semantic

GFS Data Flow

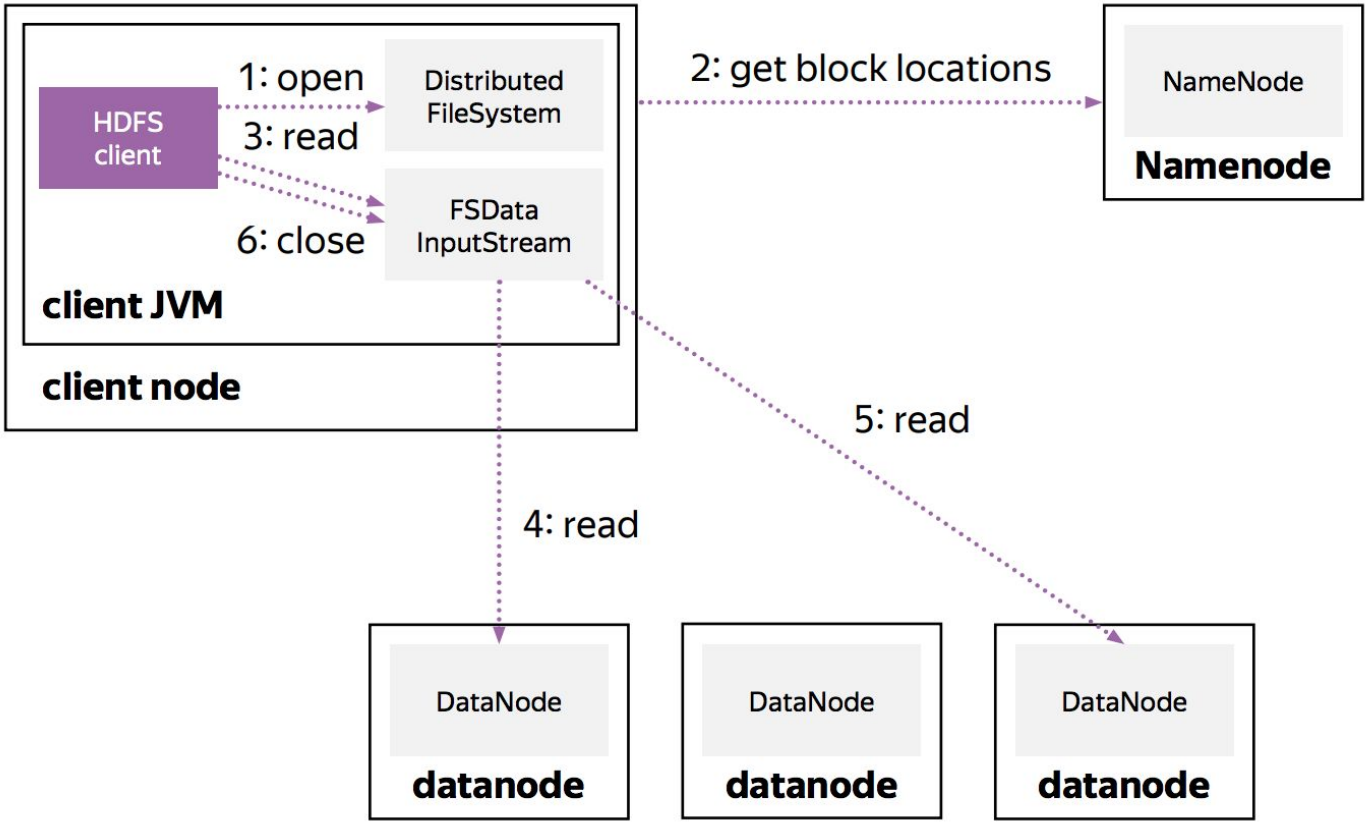


HDFS Data Flow

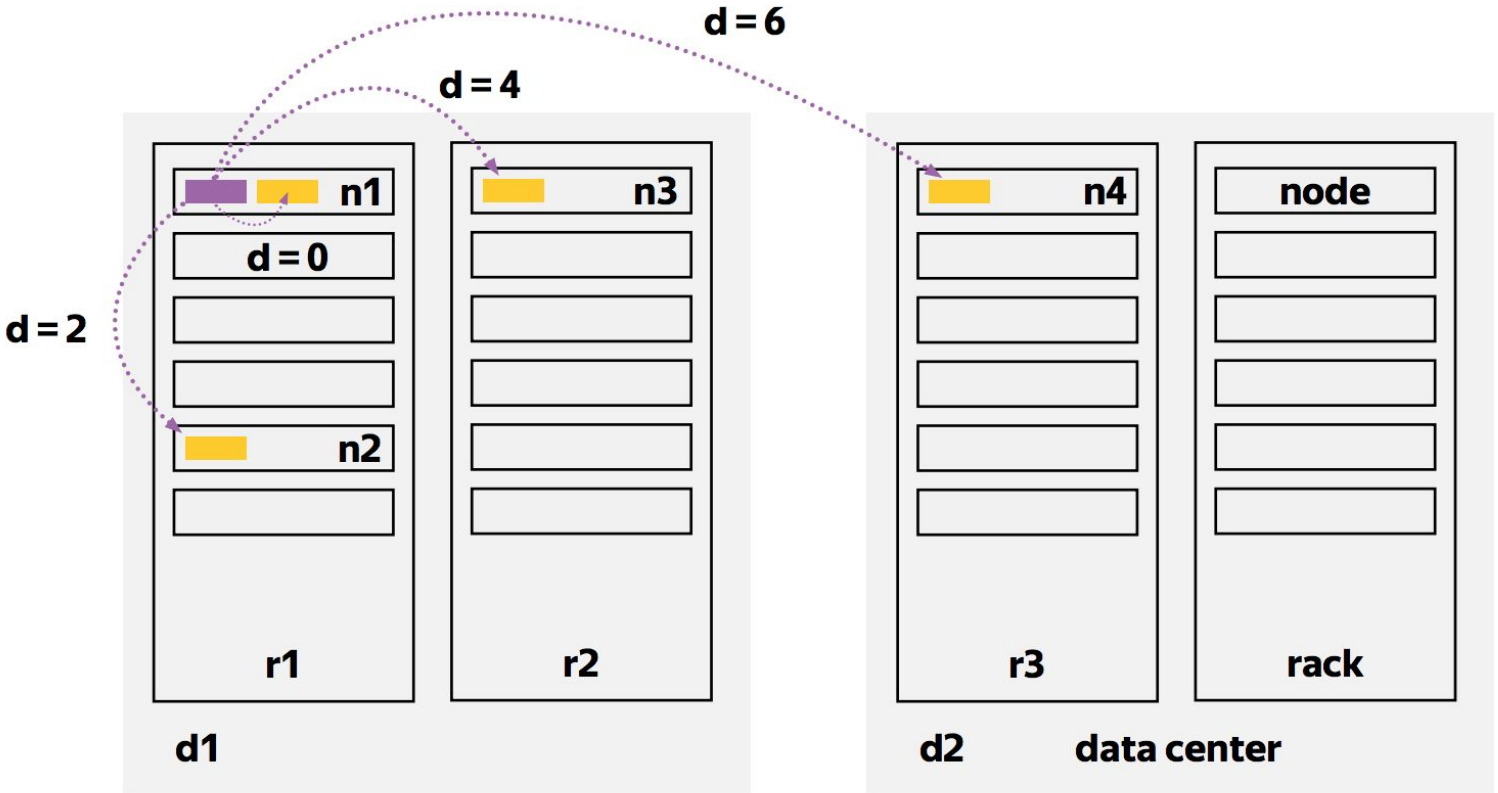


HDFS

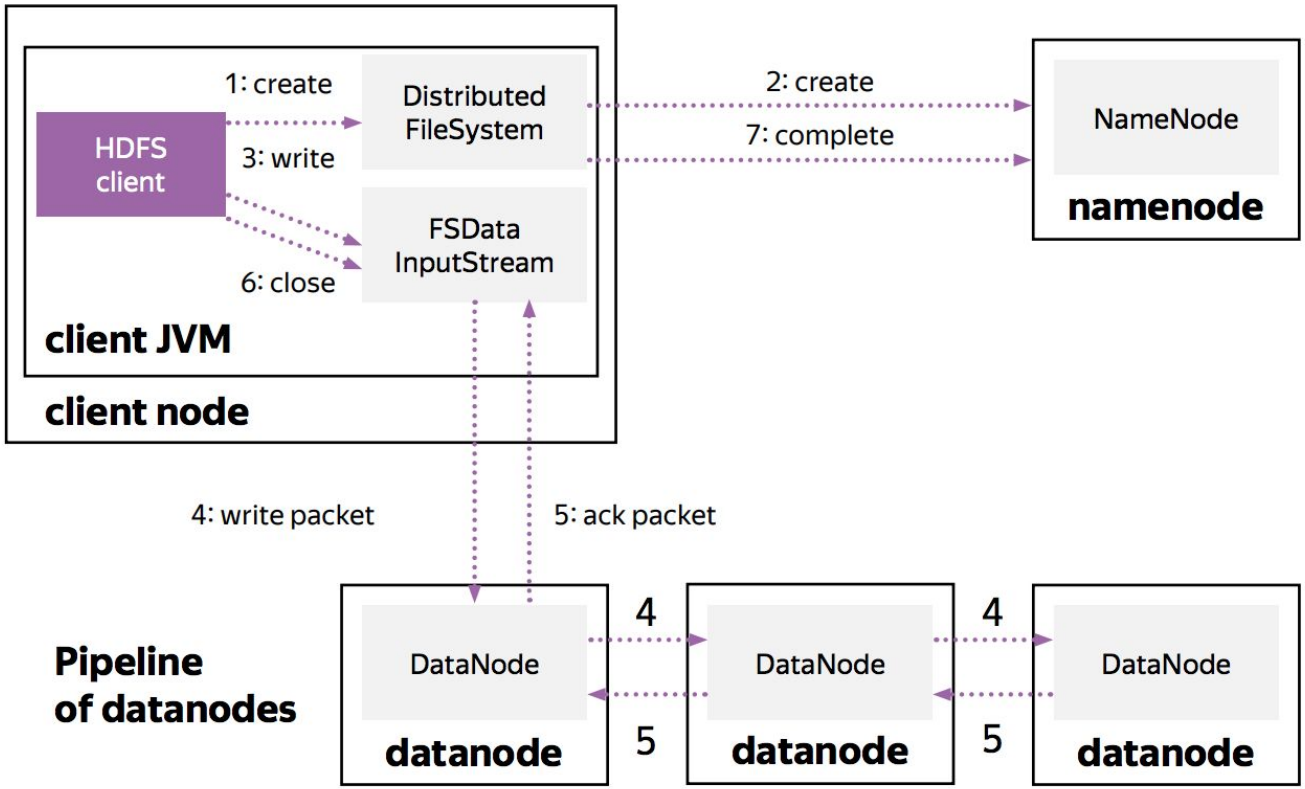
HDFS read


HDFS client
1: open
3: read
6: close
Distributed FileSystem
FSData InputStream
client JVM
client node
2: get block locations
NameNode
Namenode
4: read
5: read
DataNode
datanode
DataNode
datanode
DataNode
datanode

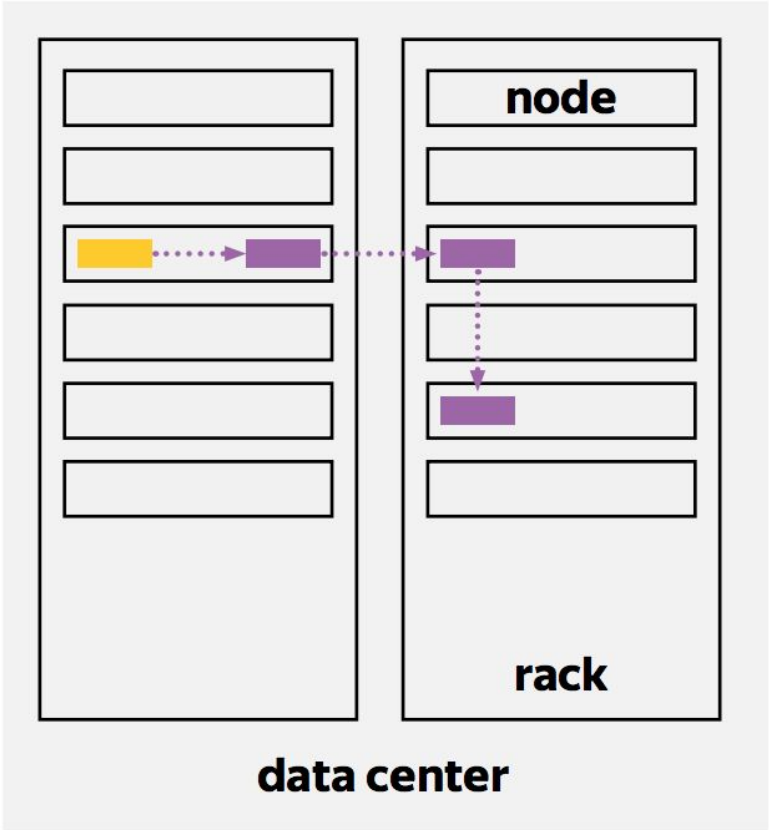
HDFS “distance”



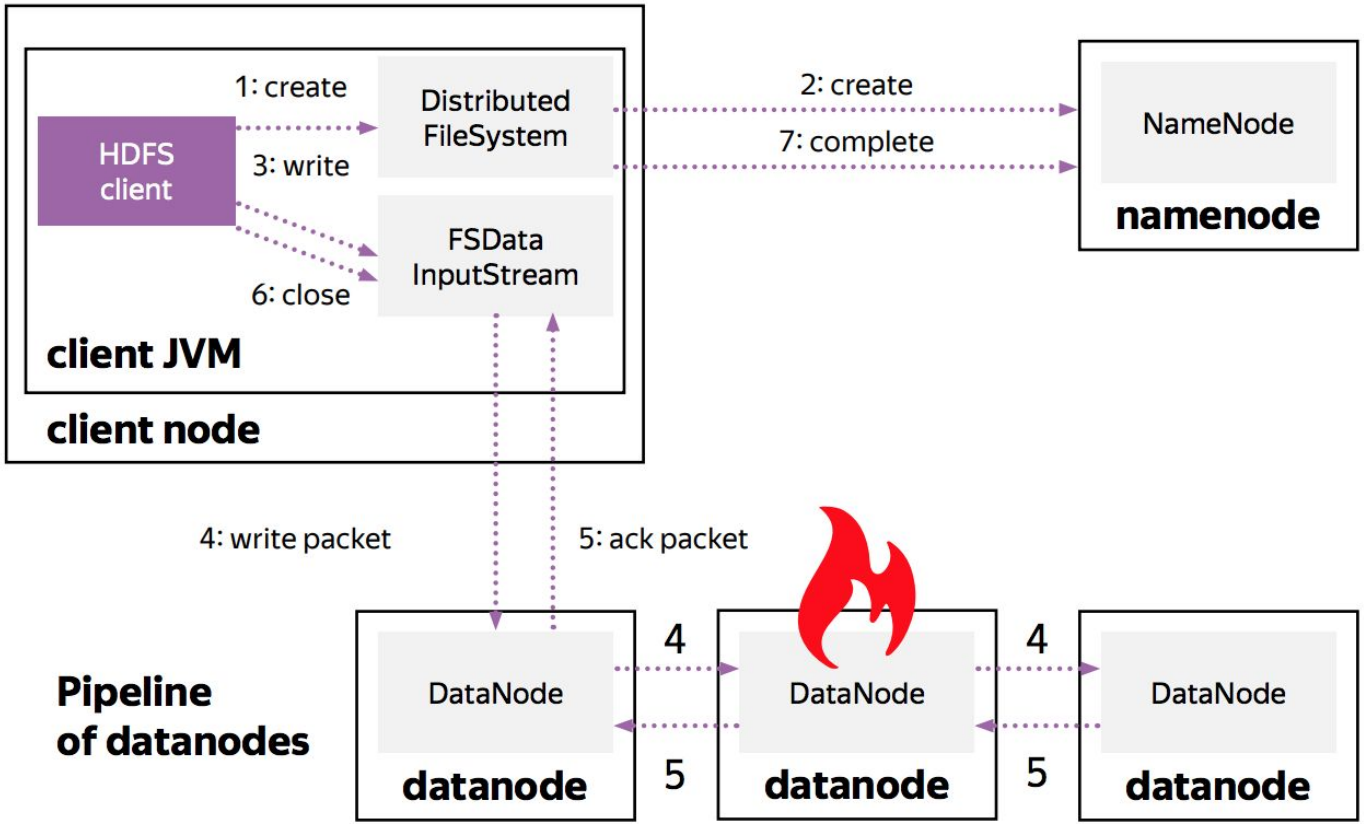
HDFS write


1: create
3: write
6: close
HDFS client
Distributed FileSystem
FSData InputStream
client JVM
client node
2: create
7: complete
NameNode
namenode
4: write packet
5: ack packet
Pipeline of datanodes
DataNode
datanode
4
5
DataNode
datanode
4
5
DataNode
datanode

HDFS Replica Placement

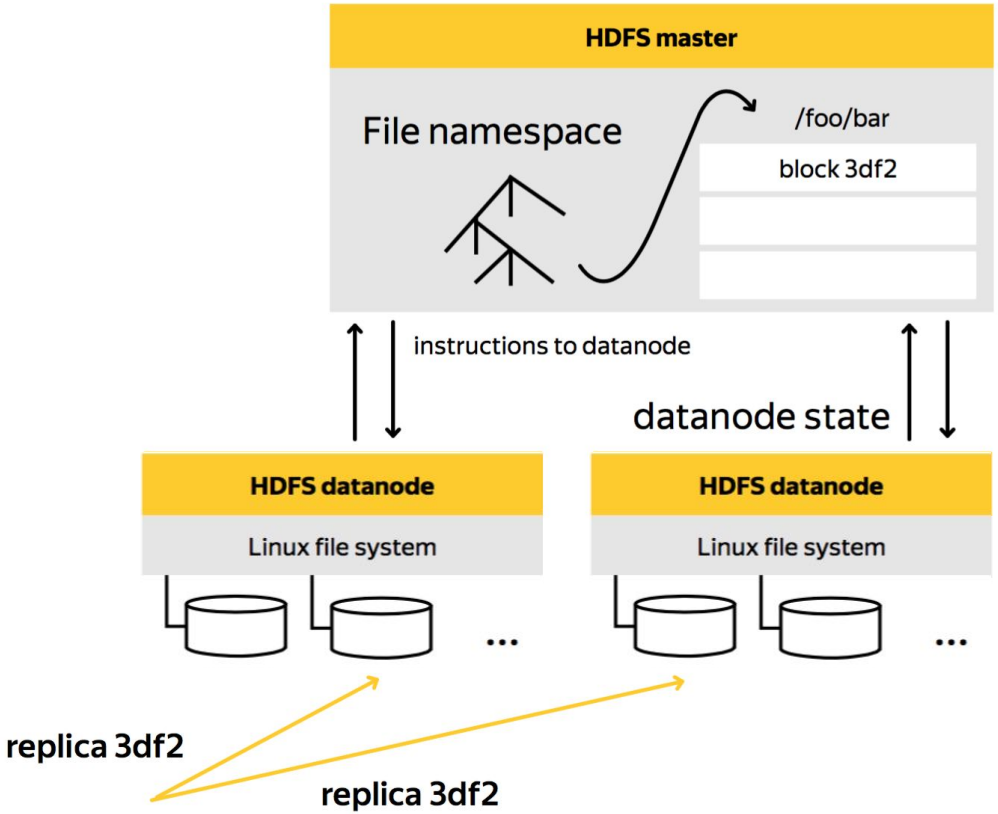


HDFS write v.2

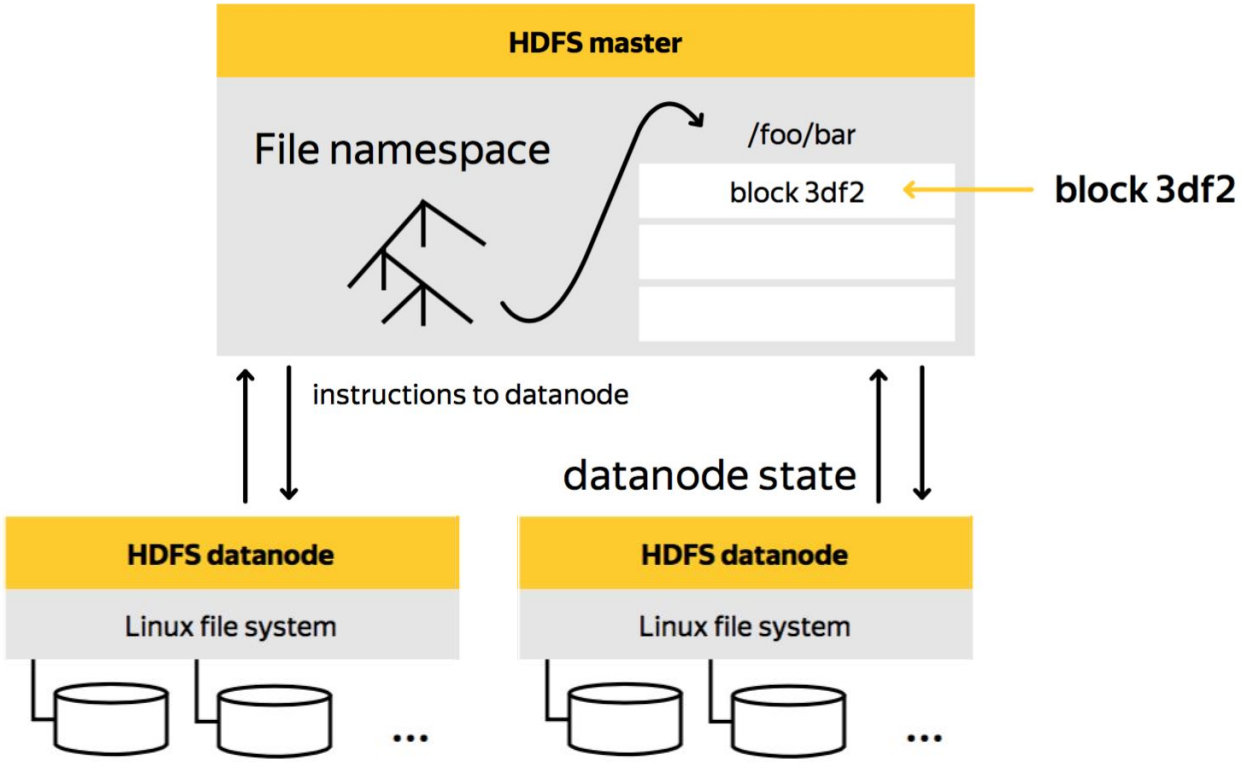


Blocks & Replicas

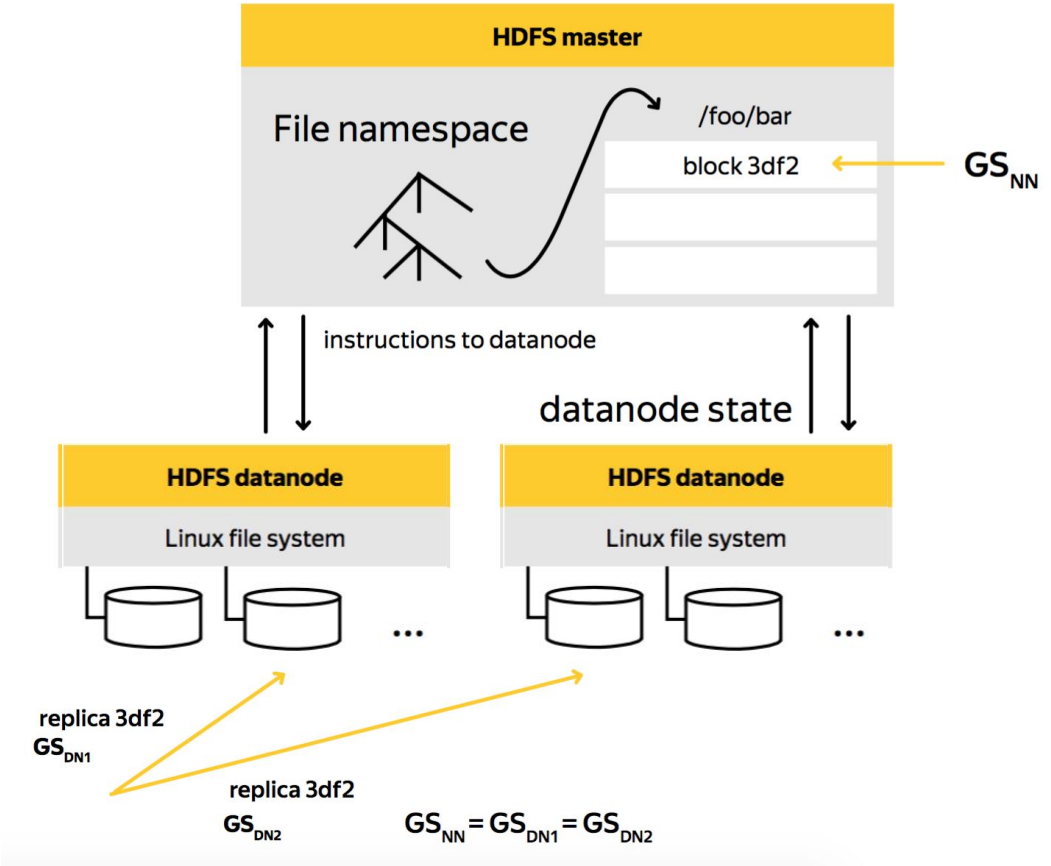
HDFS Replica



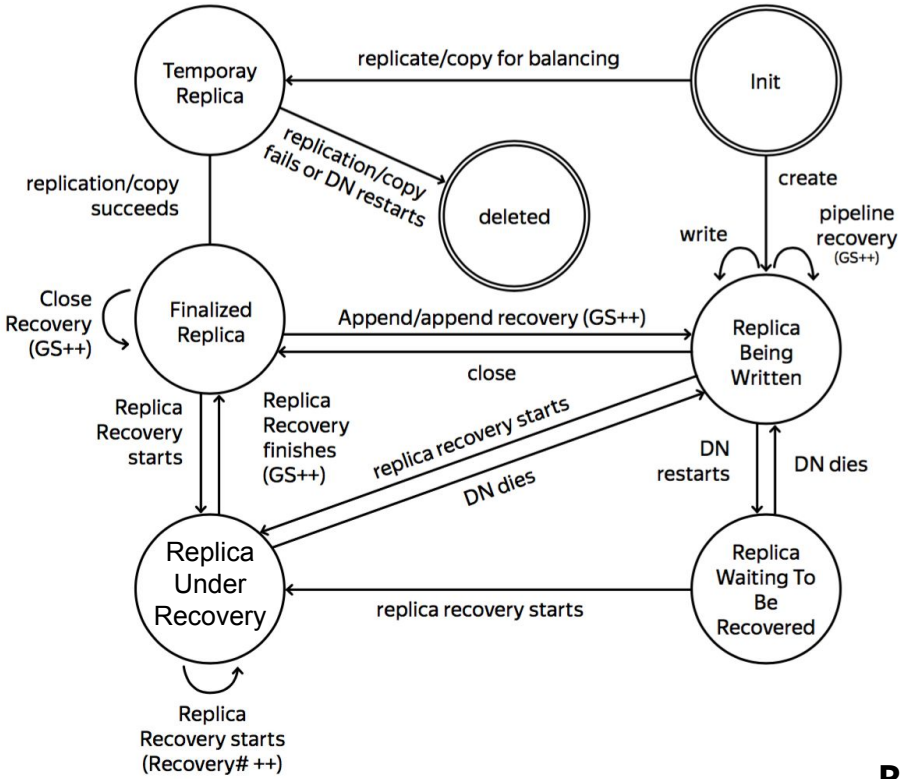
HDFS Block

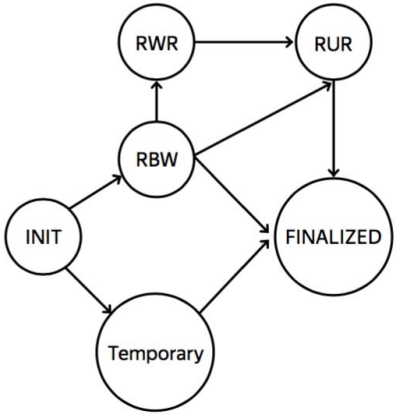

HDFS master
File namespace
/foo/bar
block 3df2
block 3df2
instructions to datanode
datanode state
HDFS datanode
Linux file system
HDFS datanode
Linux file system

Finalized Replica



Replica Recovery Process



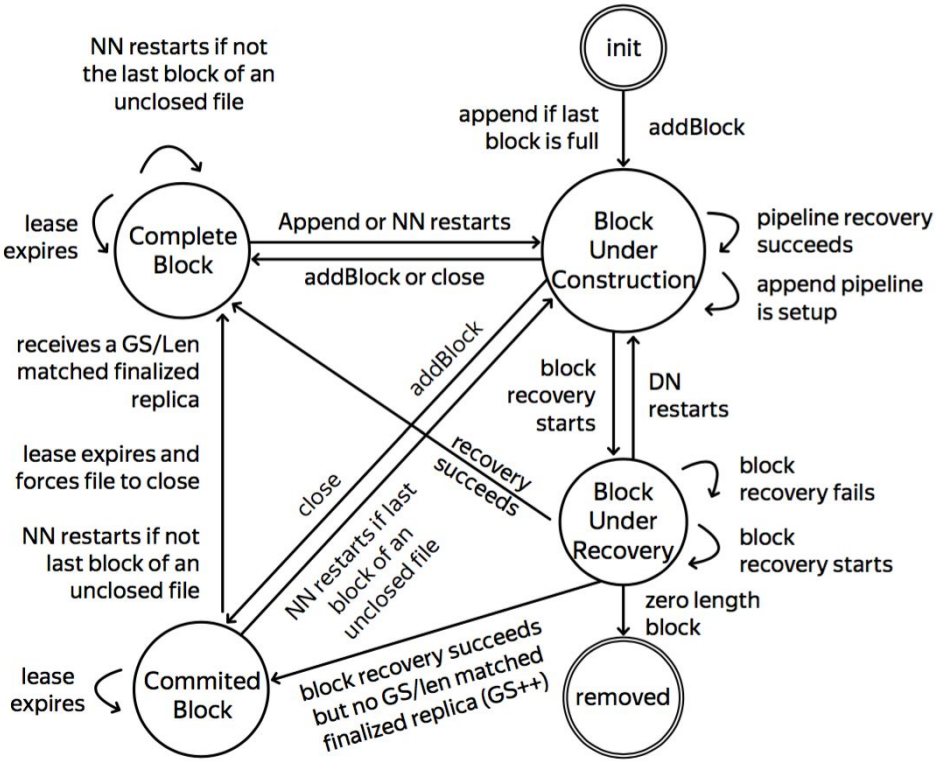


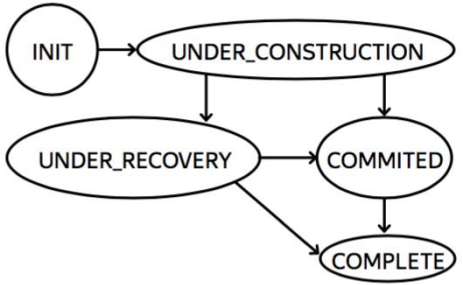
Simplified Replica State Transition

RBW / RWR / RUR / Temporary / ...

BIGDATA TEAM

Simplified Block State Transition

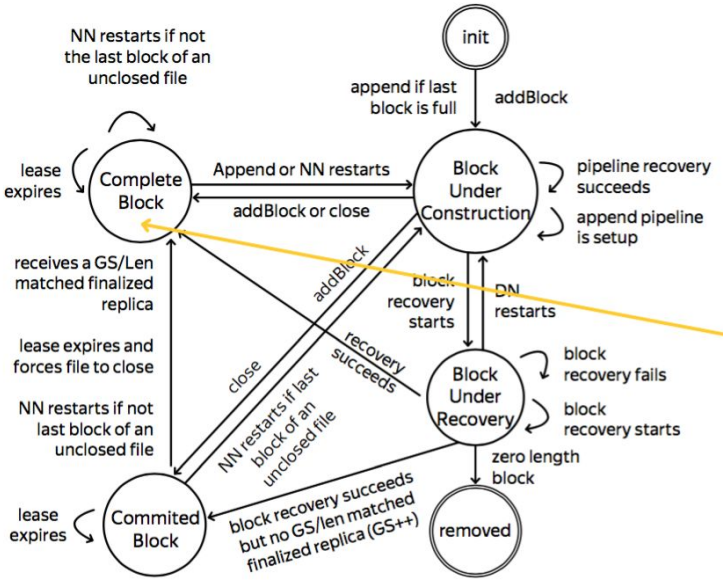


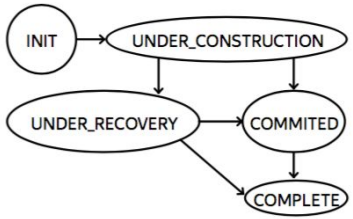


Simplified Block State Transition

BIGDATA TEAM

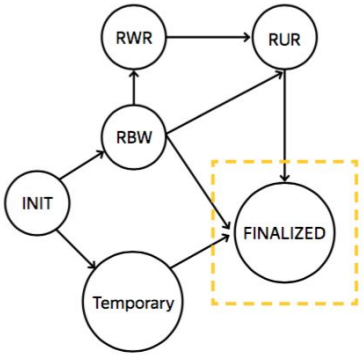
Block & Replica States Matching



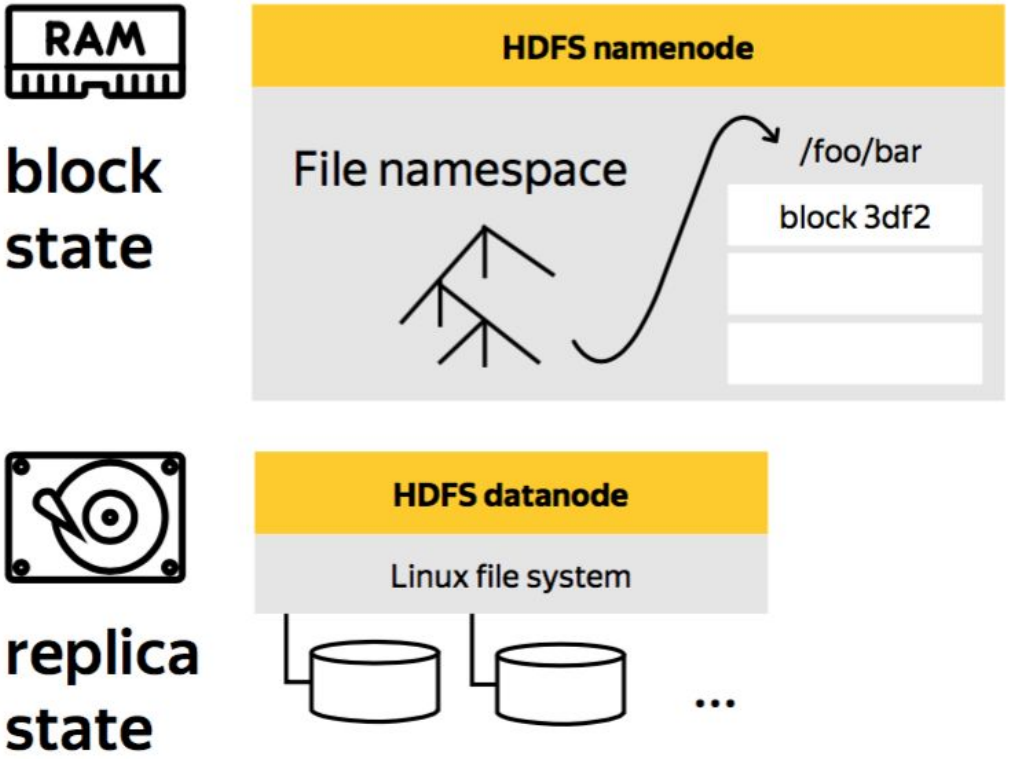


Simplified Block State Transition

complete

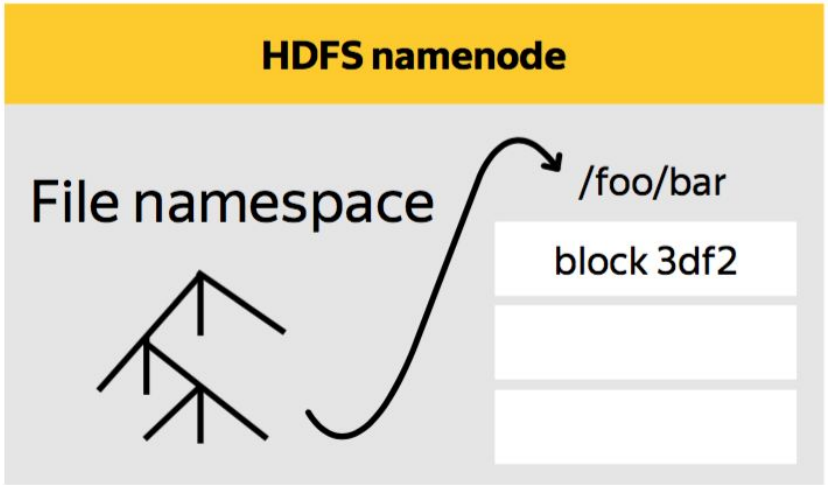


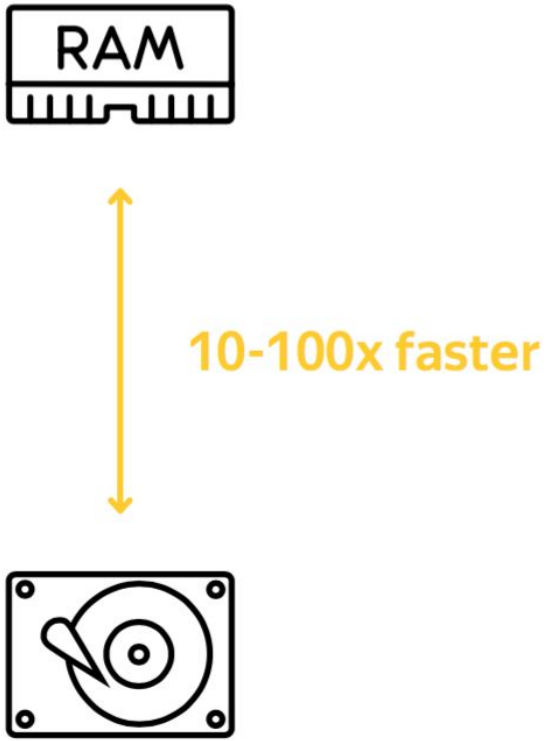
Memo: Block vs Replica



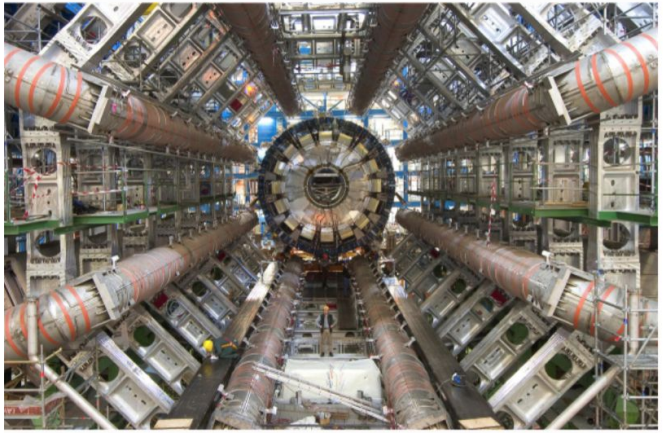
HDFS Namenode

Namenode Design Choices

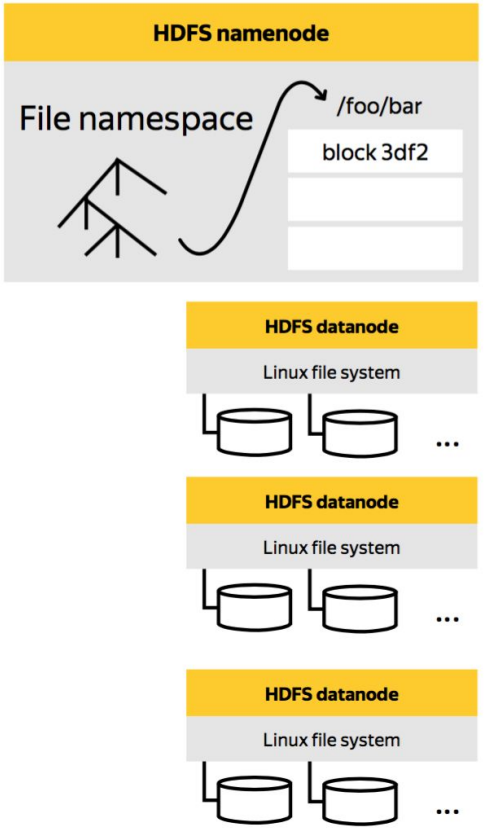




Hadoop Sizing

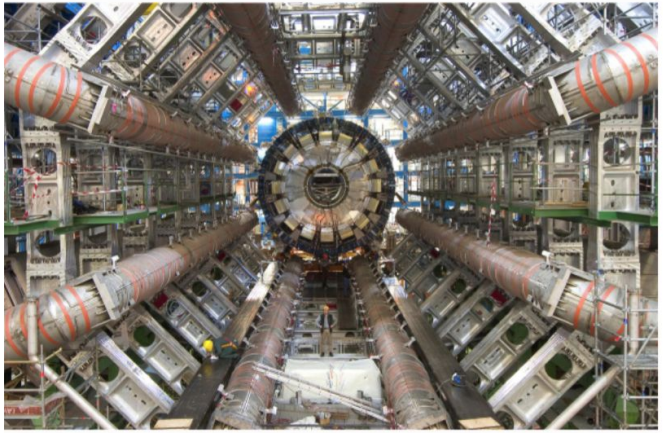


1 year ~ 10 PB



BIGDATA TEAM

Hadoop Sizing: Small Files Problem



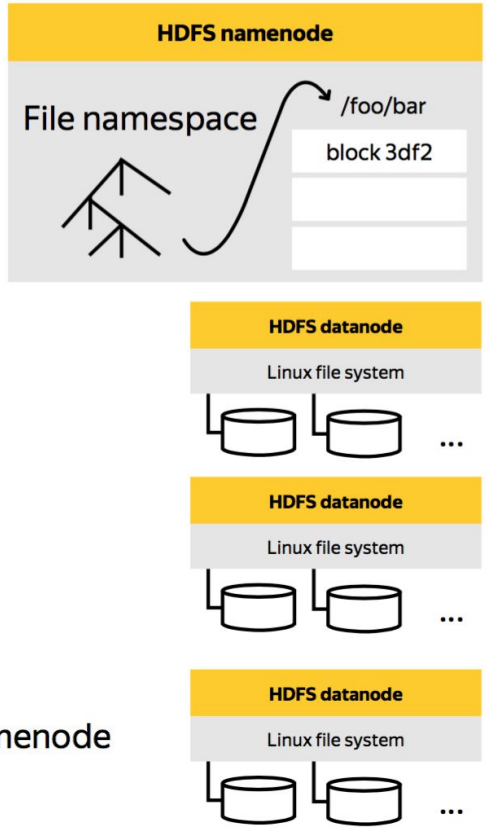
1 year ~ 10 PB

10 PB / 2 TB * 3 ~ 15 k

RAM

150 B - average block size on Namenode

https://issues.apache.org/jira/browse/HADOOP-1687



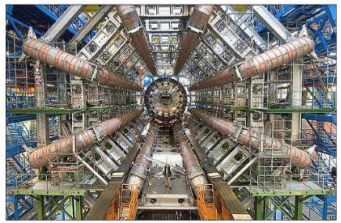
Speeding Up the Cluster

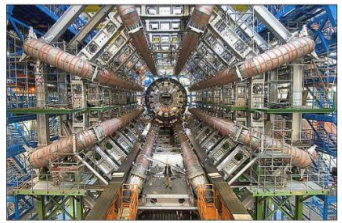
2 TB

vs

1 TB

1 TB





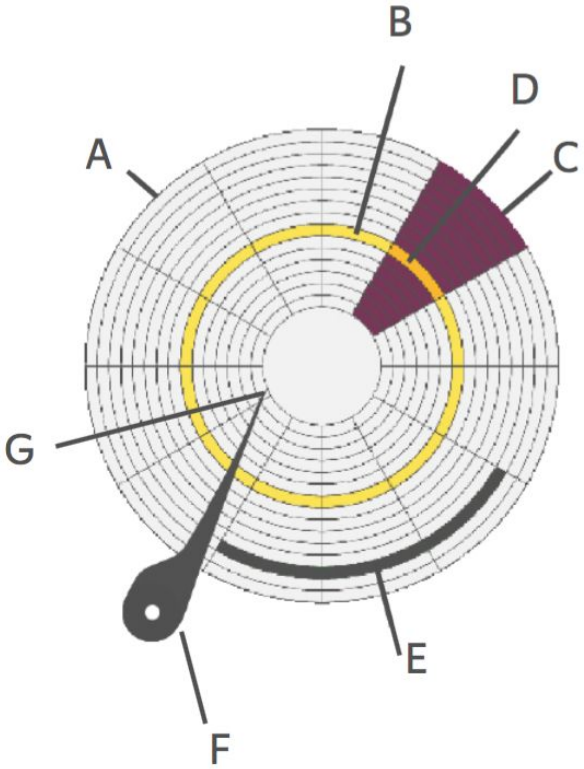
1 year ~ 10 PB 

35 days

1 year ~ 5 PB  + 5 PB 

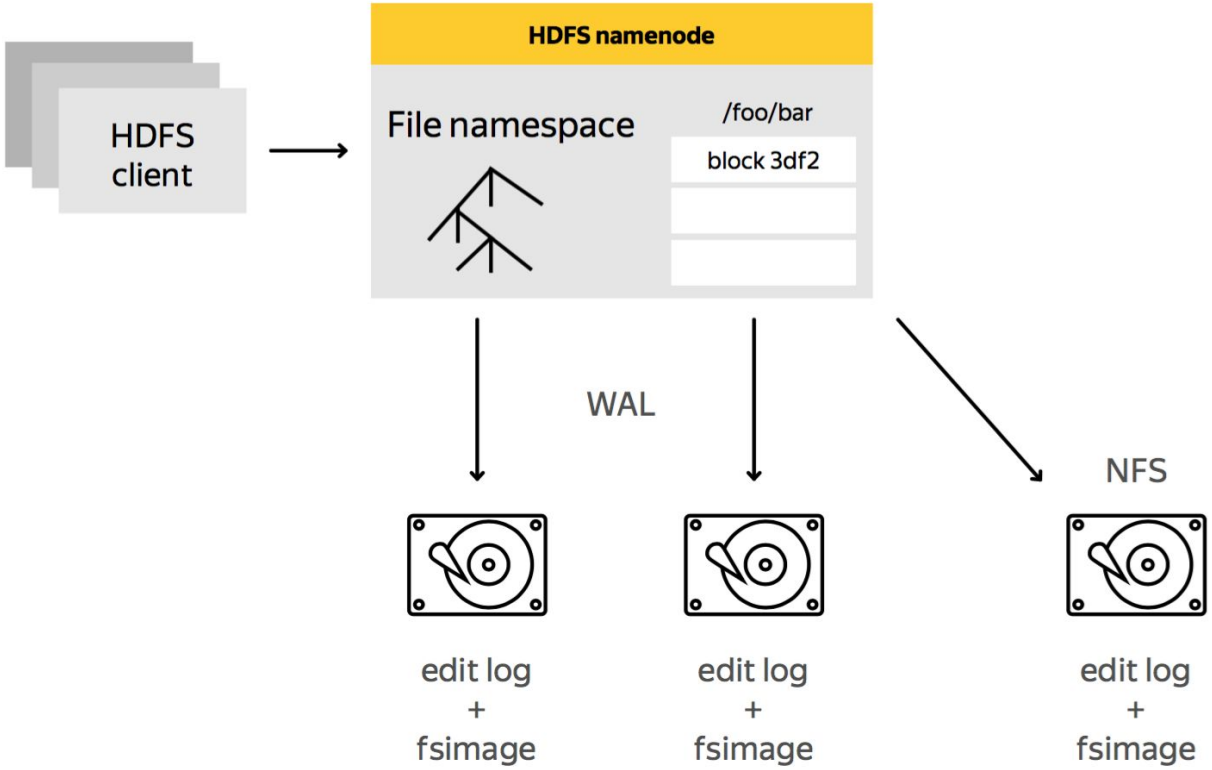
17.5 days

Default Block Size

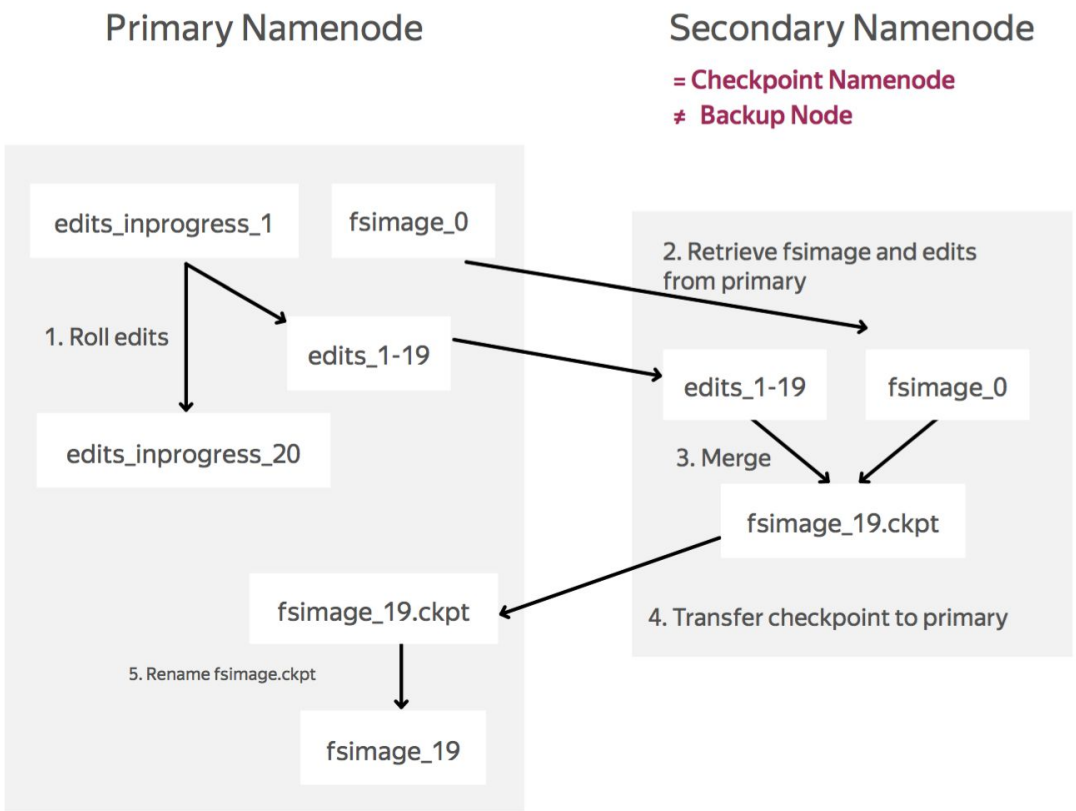


- reading speed - 3.5 GB/sec
- 128 MB - 30-40 ms
- seek time: 0.2-0.8 ms

Namenode - SPoF



Checkpoint Namenode



Checkpoint Namenode

Hadoop and NoSQL Downfall Parody
https://www.youtube.com/watch?v=hEqQMLSXQIY

Summary

- Вы можете привести примеры приложения Data Science, а также где можно наработать ML-навыки
- Вы можете перечислить 3 типа Многопроцессорных Вычислительных Систем, а также их типовые приложения
- Вы можете нарисовать диаграмму переходов состояний для блока и реплики
- Вы можете объяснить “design choice” архитектуры Namenode, а также указать на разницу между различными типами Namenode: Primary / Secondary / Checkpoint / Backup.
- Вы можете оценить ресурсы, необходимые для Hadoop-кластера (Hadoop sizing) для решения вашей задачи, а также можете объяснить что такое small files problem

Thank you! Questions?

Feedback: http://rebrand.ly/mf2019q2_feedback_01_hdfs

Dral Alexey (aadral@bigdatateam.org)
CEO at BigData Team, http://bigdatateam.org/
https://www.linkedin.com/in/alexey-dral
https://www.facebook.com/bigdatateam/

Appendix

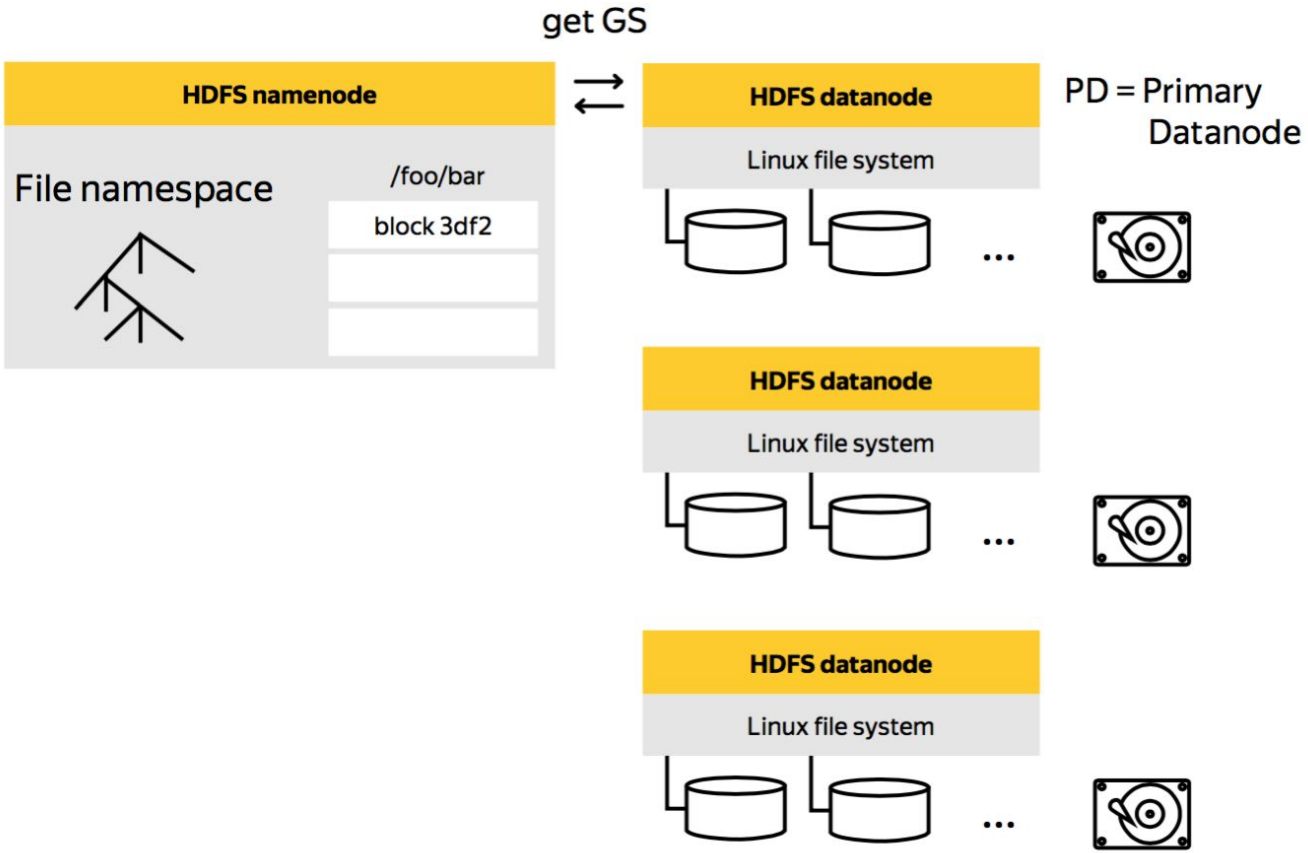
Course Resources User Guide

http://rebrand.ly/mf2019q2_user_guides

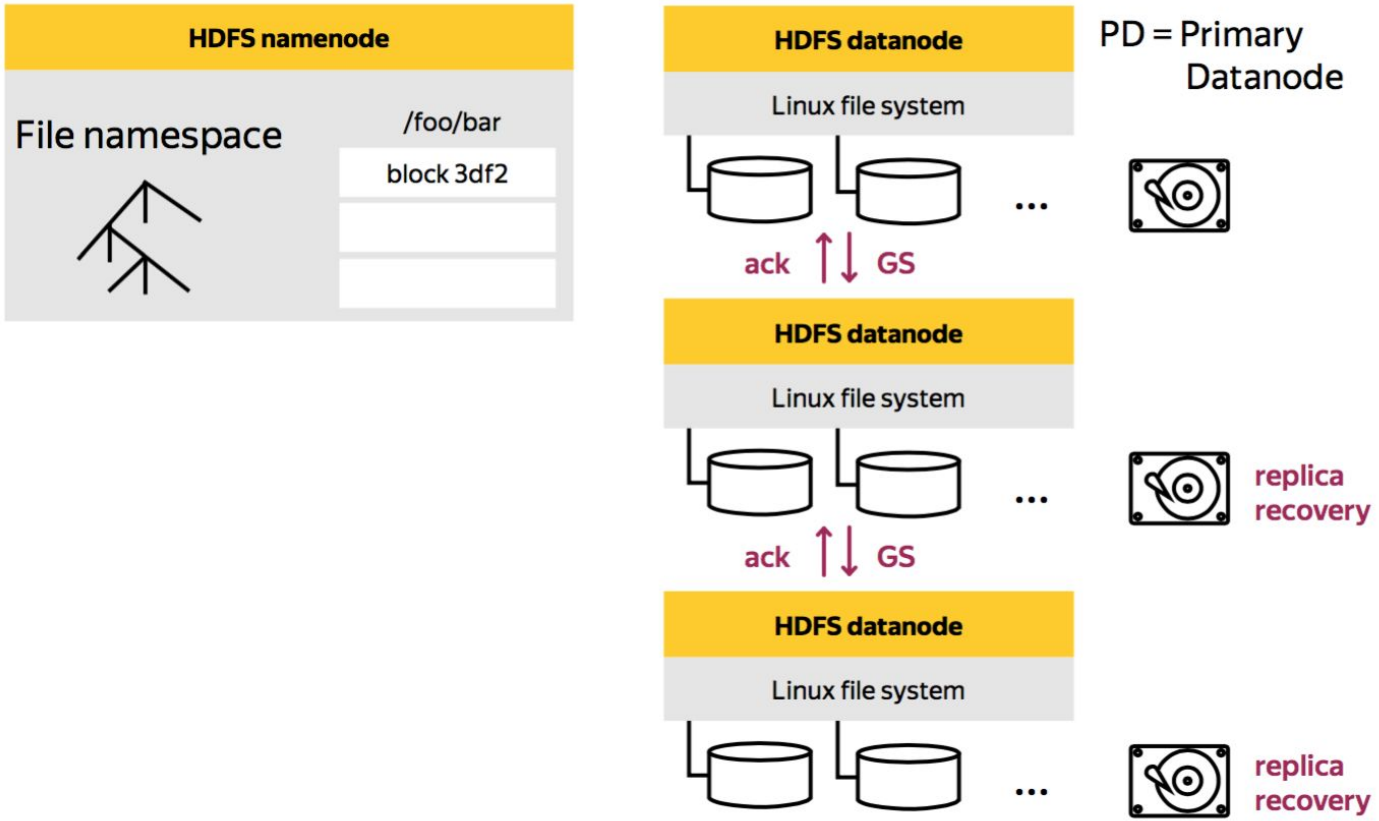
Recovery Process

- Lease Recovery
  - Block Recovery
    - Replica Recovery
- Pipeline Recovery

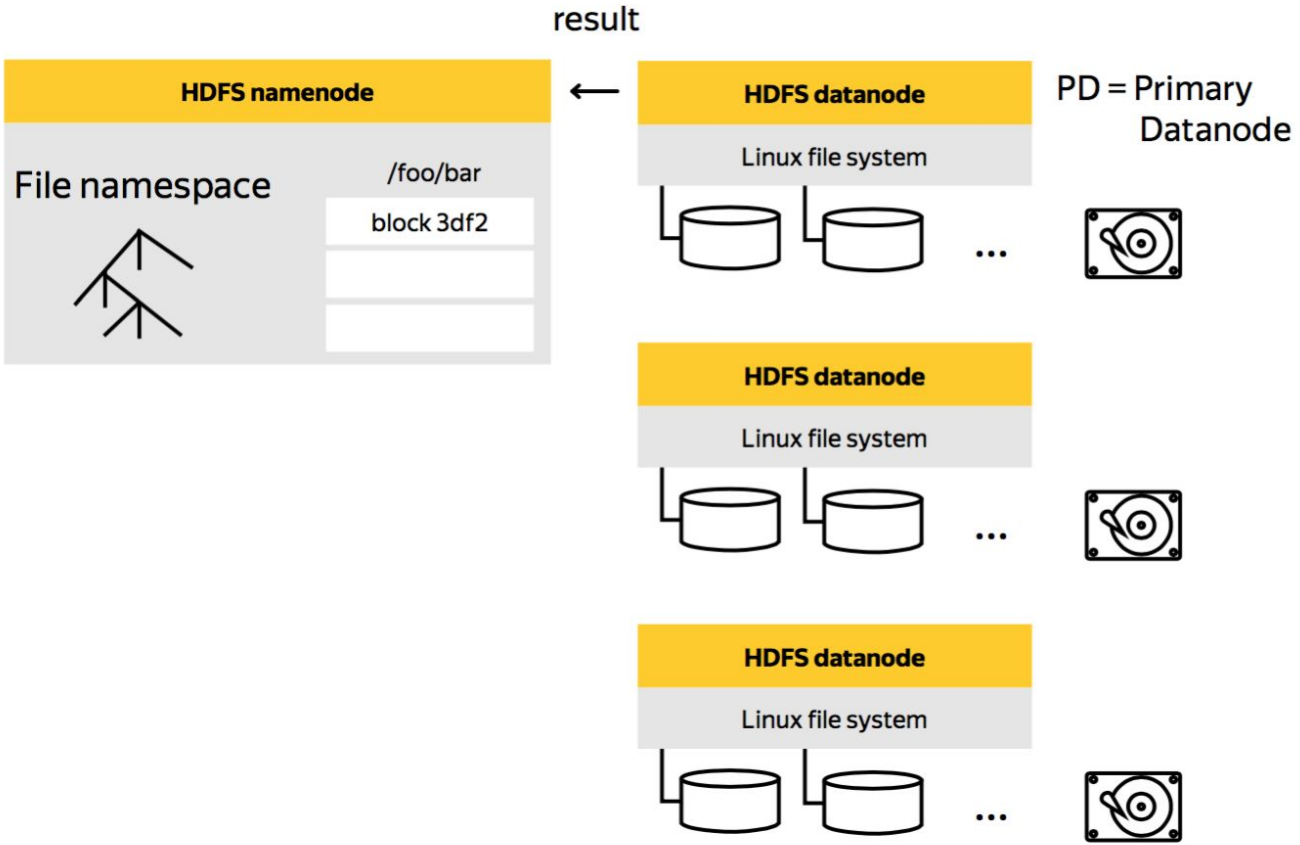
Block Recovery



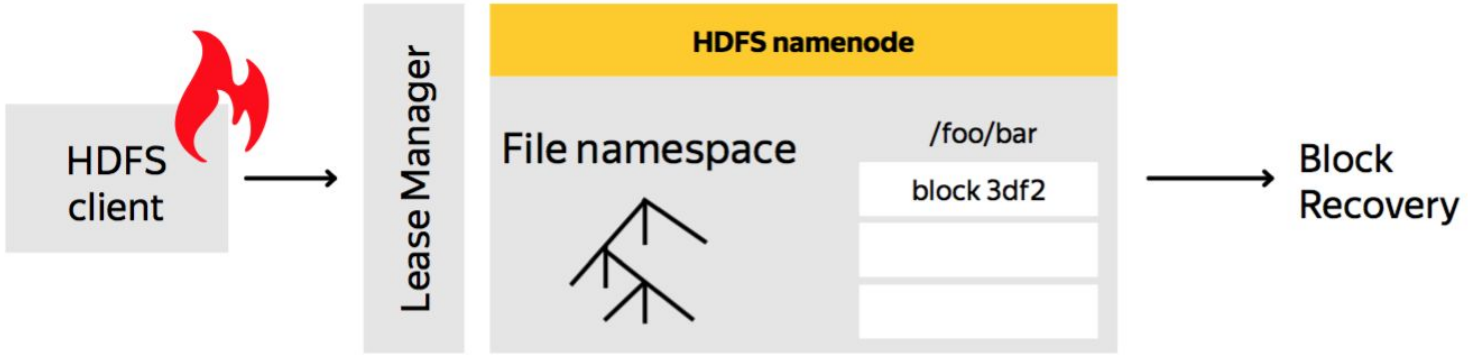
Block Recovery



Block Recovery

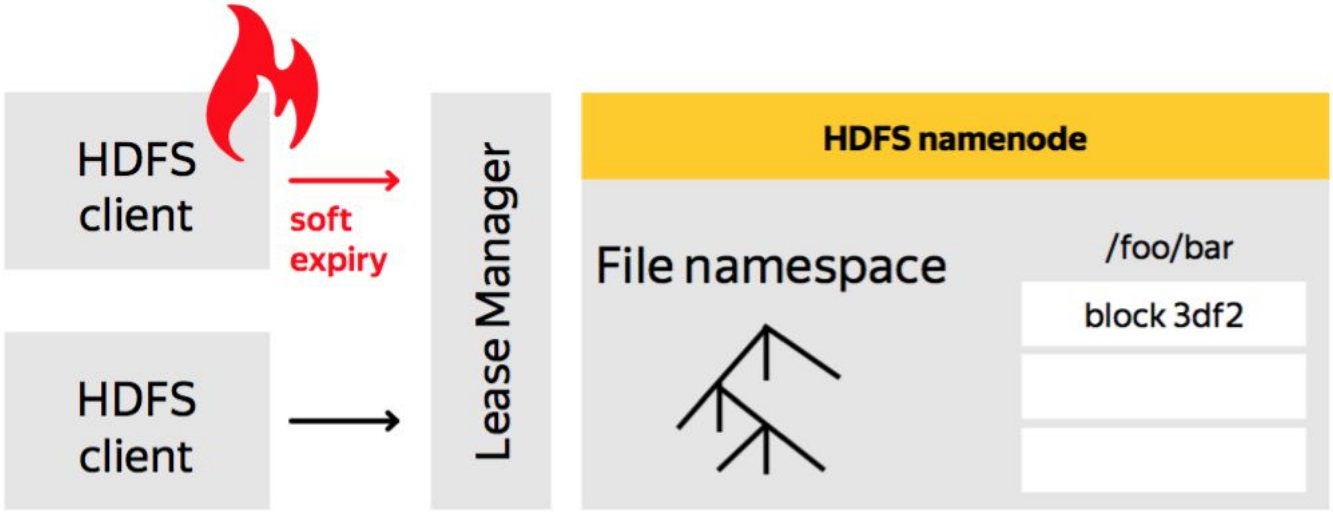


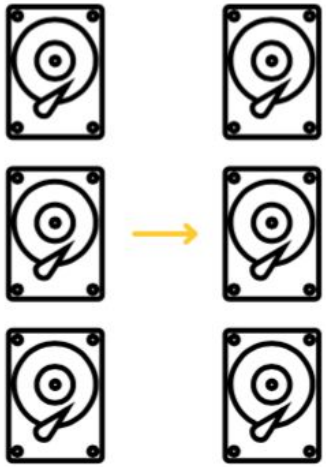
Lease Recovery



user$_1$, /path/1, lease **(soft + hard)**
user$_1$, /path/2, lease (soft + hard)
user$_2$, /path/3, lease (soft + hard)
user$_3$, /path/4, lease (soft + hard)
...

Guarantees

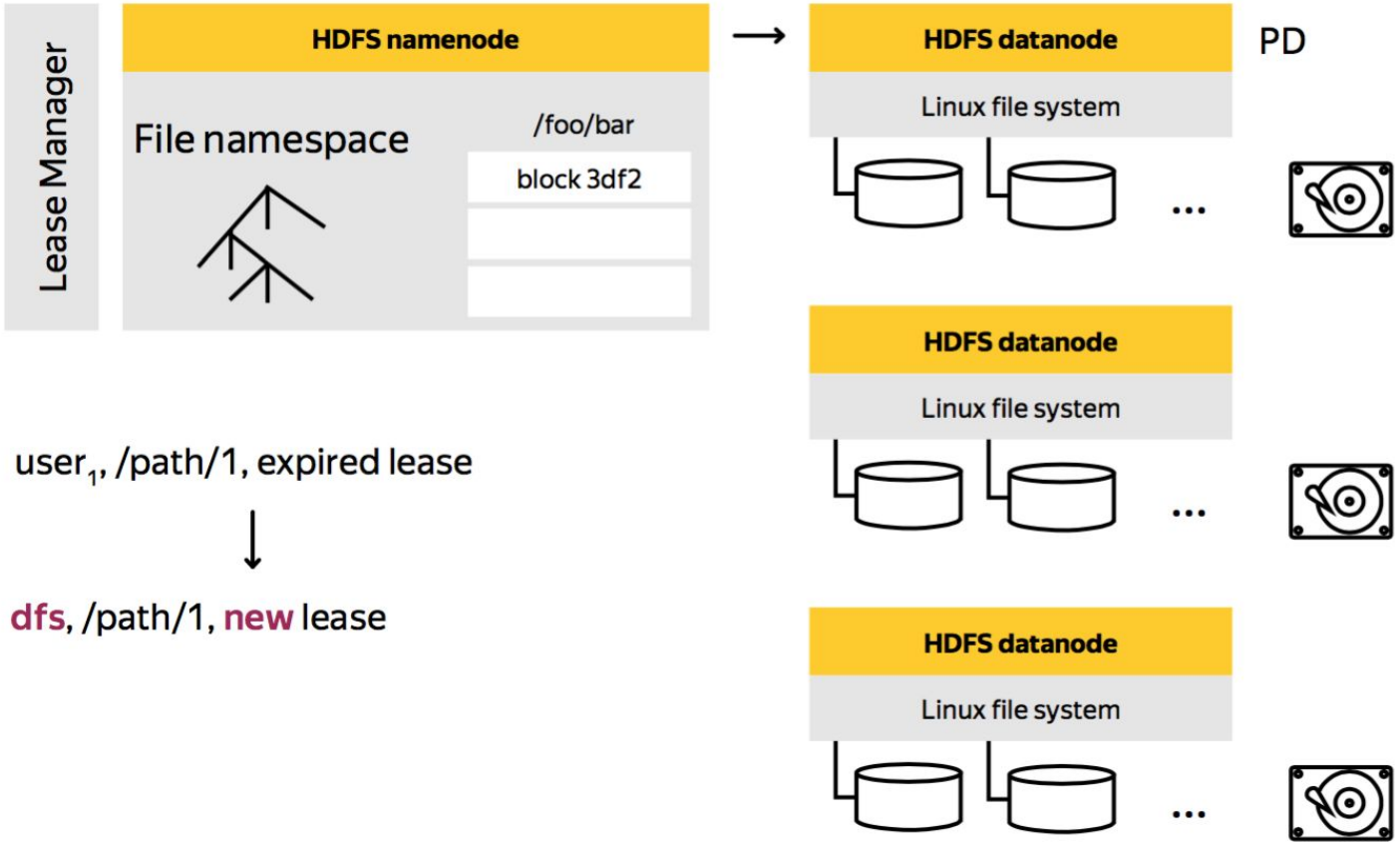




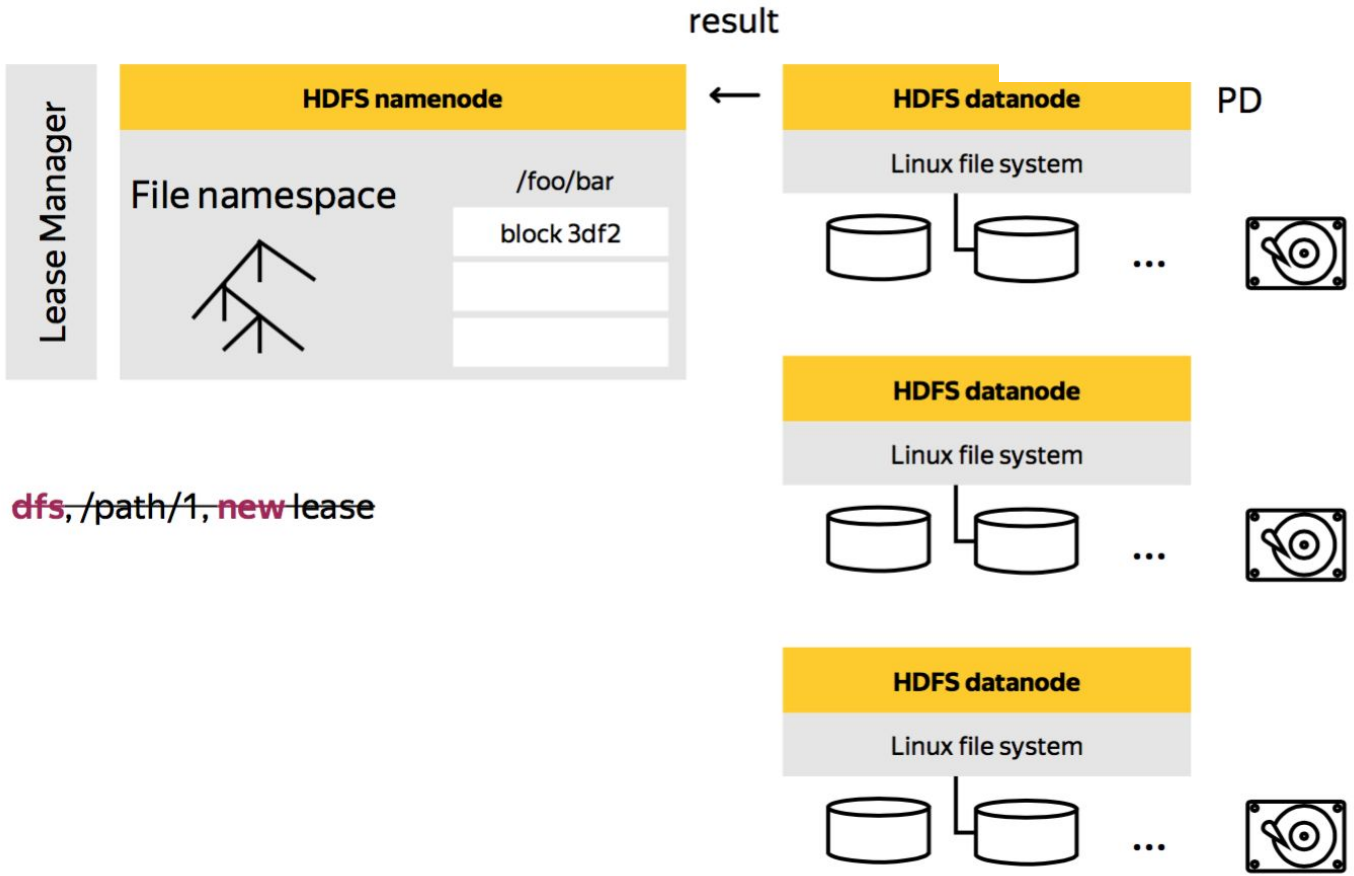
consistency guarantee

- Concurrency control
- Consistency guarantee

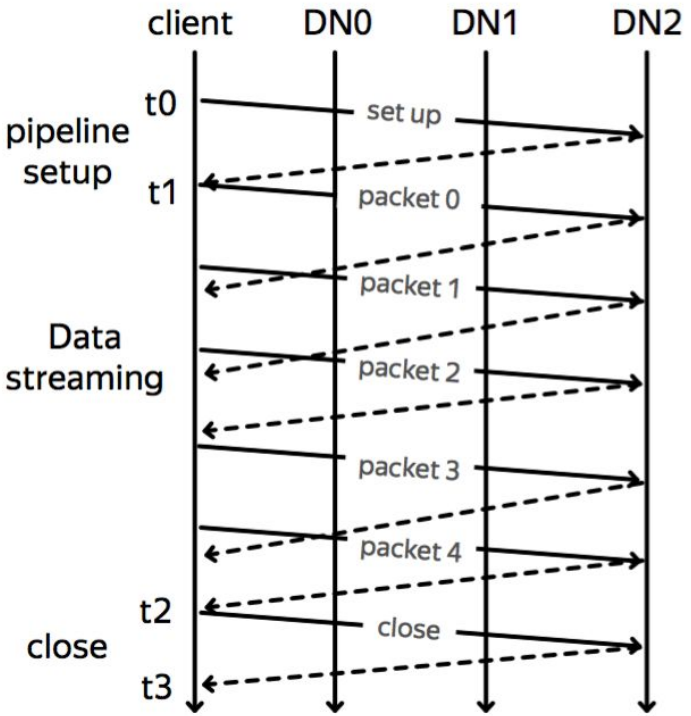
Lease Recovery

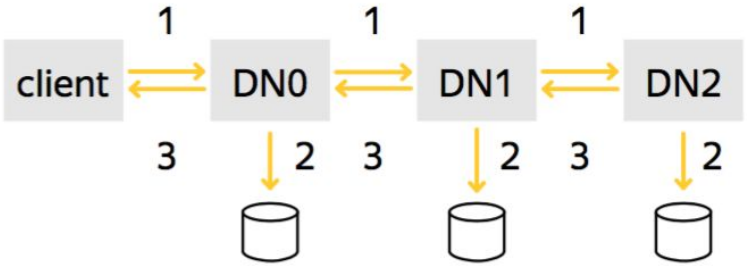


Lease Recovery

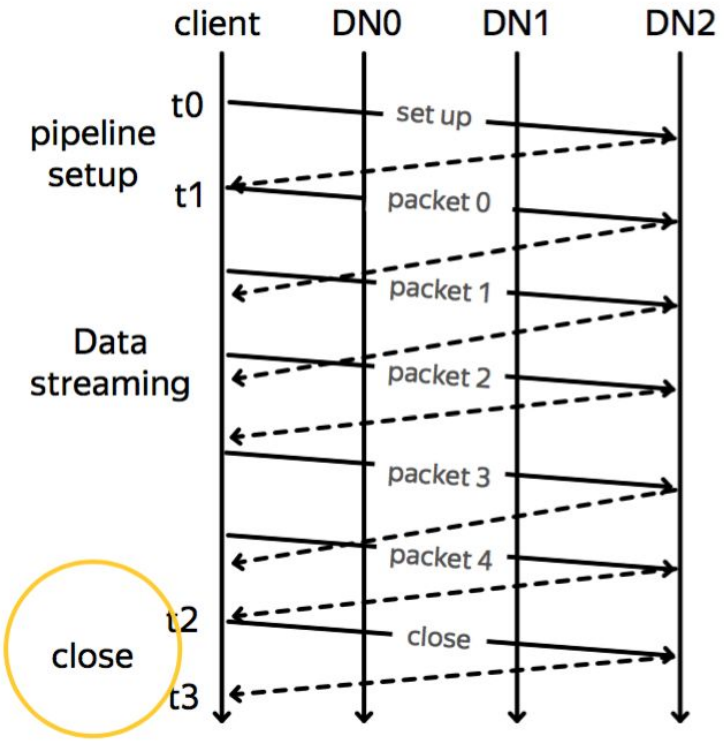


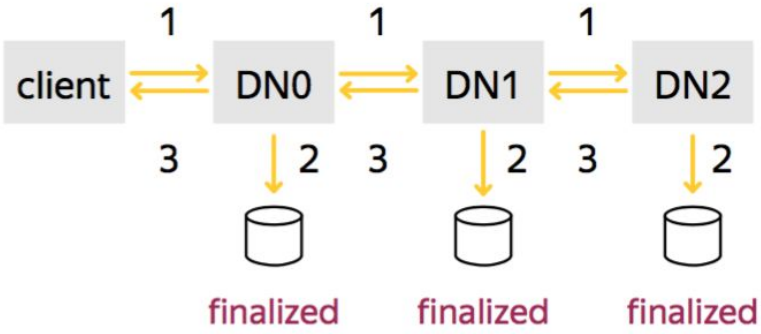
Pipeline Recovery



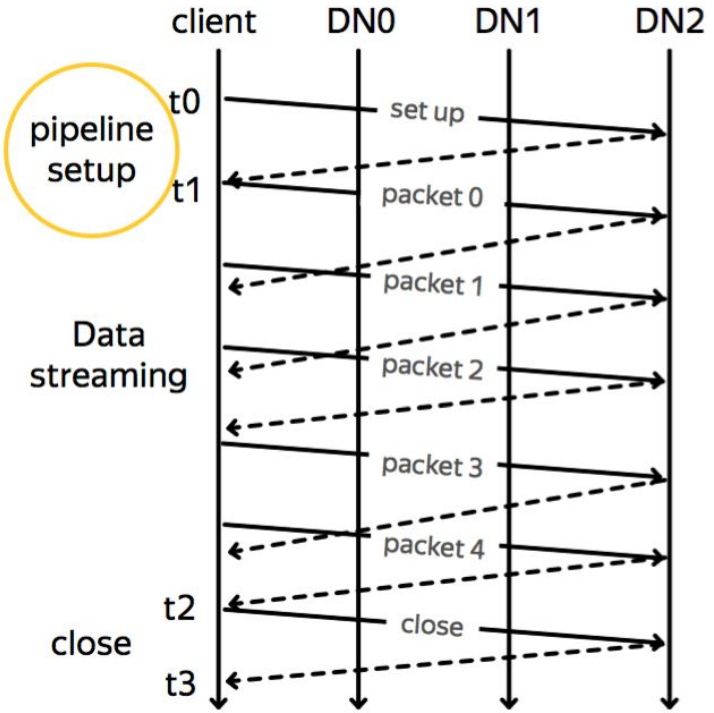


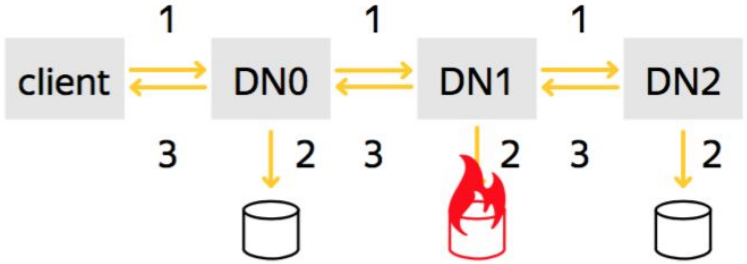
Pipeline Stages





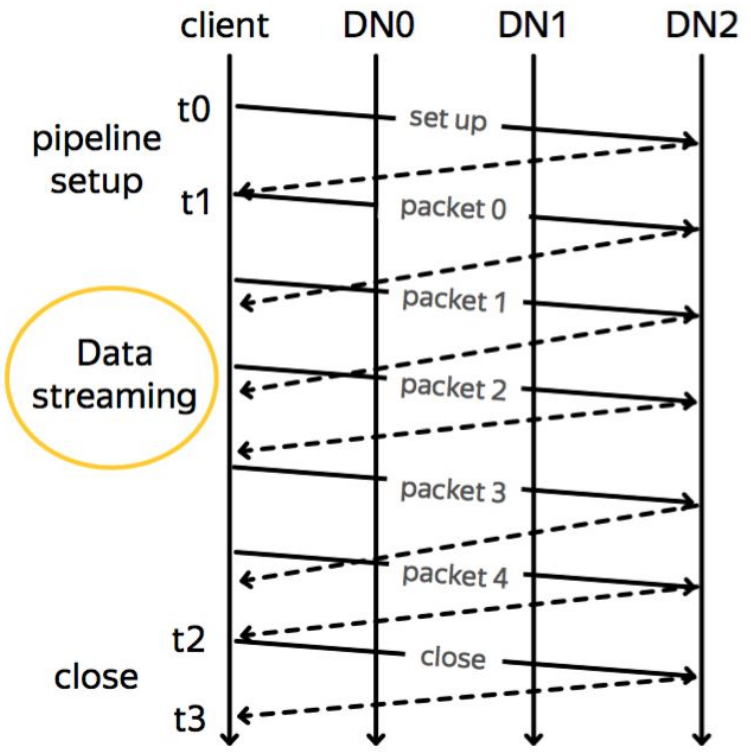
Q&A: Что будем делать?

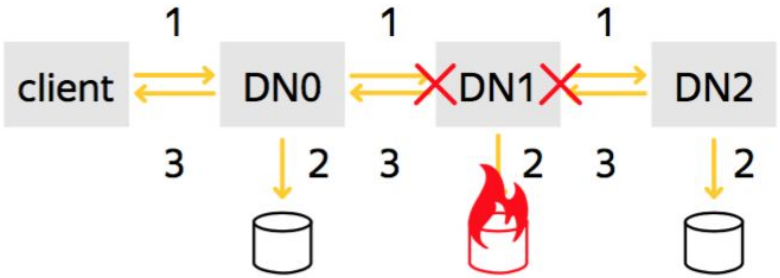




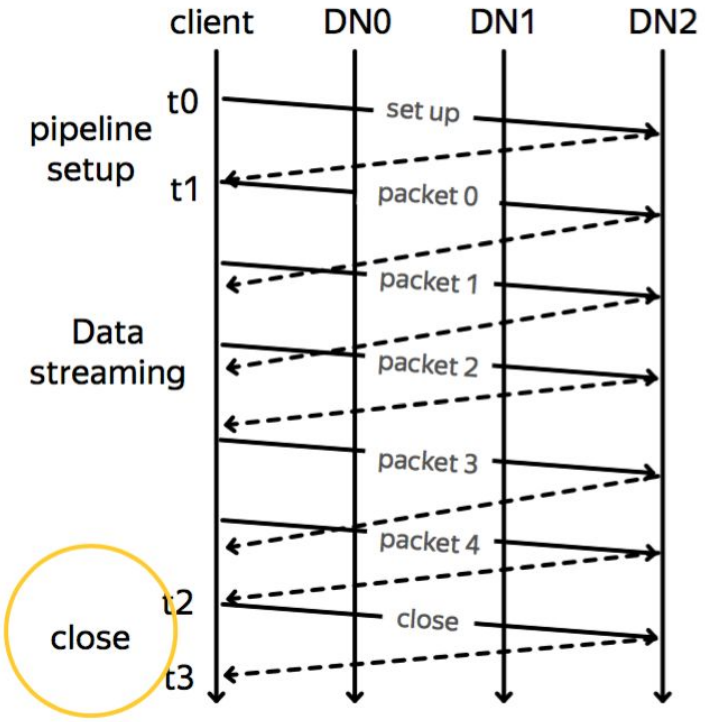
cases:
1. new file
2. append mode

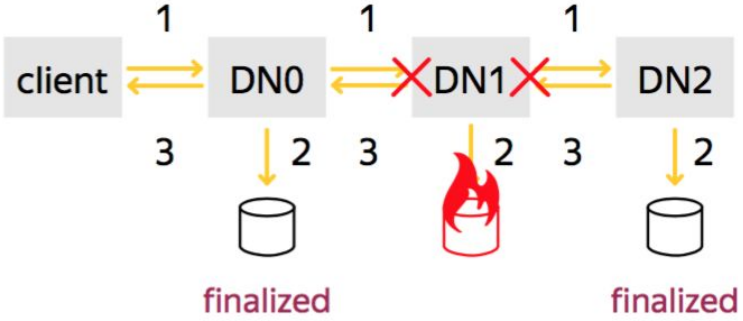
Q&A: Что будем делать?





Q&A: Что будем делать?





Fallacies of Distributed Computing

1. The network is reliable
2. Latency is zero
3. Bandwidth is infinite
4. The network is secure
5. Topology doesn’t change
6. There is one administrator
7. Transport cost is zero
8. The network is homogeneous