CREATE A HADOOP CLUSTER AND MIGRATE 39PB DATA PLUS 150000 JOBS/DAY

Stuart Pook (s.pook@criteo.com @StuartPook)

criteo

BROUGHT TO YOU BY LAKE

Anna Savarin, Anthony Rabier, Meriam Lachkar, Nicolas Fraison, Rémy Gayet, Rémy Saissy, Stuart Pook, Thierry Lefort & Yohan Bismuth

2014, PROBLEM

A CLUSTER CRITICAL FOR BOTH STORAGE AND COMPUTE

- 39 petabytes raw storage
- 13404 CPUs (26808 threads)
- 105 terabytes RAM
- 40 terabytes imported per day
- > 100 000 jobs per day

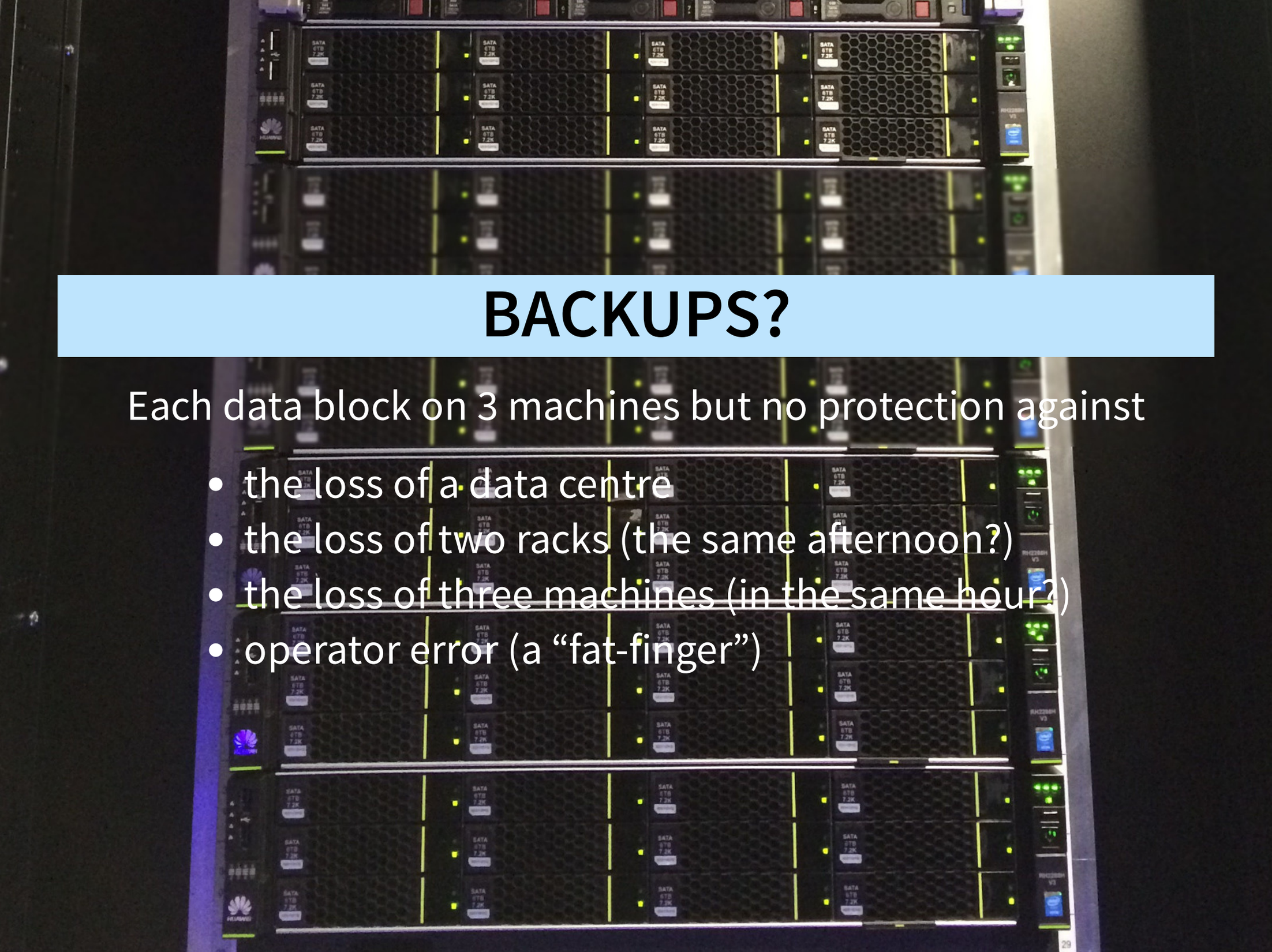
still running CDH4 and CentOS 6

data centre is full

NO DISASTER PLAN :-(

Defining a disaster: what do we have to lose?

- the data “data is our blood”
- write access (needed for compute)
- compute
  - 72 hours of buffers
  - 1st of month billing is sacred
  - prediction models updated every 6 hours
  - margin feedback loop



BACKUPS?

Each data block on 3 machines but no protection against

- the loss of a data centre
- the loss of two racks (the same afternoon?)
- the loss of three machines (in the same hour?)
- operator error (a “fat-finger”)

BACKUP IN THE CLOUD?

backup too long

restore too long

COMPUTE IN THE CLOUD?

- where to find 27000 threads?
- reservation too expensive
- no need for elasticity as 100% charge
- growth too fast: 200 → 2600 (×12 in 2½ years)
- Requires reserved instances → same price
- Hadoop needs network for east-west traffic
- Criteo already has datacentres
- Criteo likes bare metal
- In-house is cheaper and better

A NEW CLUSTER TO:

- protect the data
- duplicate the compute
- migrate to CDH5 and CentOS 7
  - new functionality → Spark
- implement our capacity planning
  - increase the compute
  - increase storage capacity

DIGRESSION: WHEN TO MIGRATE?

Migrating from CDH4 to CDH5 seems easy, but:

- Huge fsimage (9 GB)
- Downtime must be limited

A migration can go wrong

- We have a new cluster
- Use it for an offline upgrade

Do OS migration to CentOS 7 later

- Do not do everything at once
- Impact on users as python, mono etc change

HOW TO BACKUP: A SPARE CLUSTER?

- No, two clusters master/slave whose roles can swap
- Essential jobs run on the master
- Other jobs split between the two clusters
- Copy output from essential jobs to the slave

IMPORT NEW DATA

We have a L

- kafka feeds a cluster
- copy to the other

We could have used a Y but

- overload intercontinental links
- different data imported

→ two clusters master/slave

THE MASTER FAILS …

- Turn the L
- Move essential jobs
- Stop non-essential jobs
- Stop development & ad-hoc jobs
- To be implemented

AT CRITEO HADOOP GREW FROM POC TO PROD

AD-HOC FIRST CLUSTER

- created as a POC
- grew without a master plan
- more users
- more essential
- now the datacentre is full

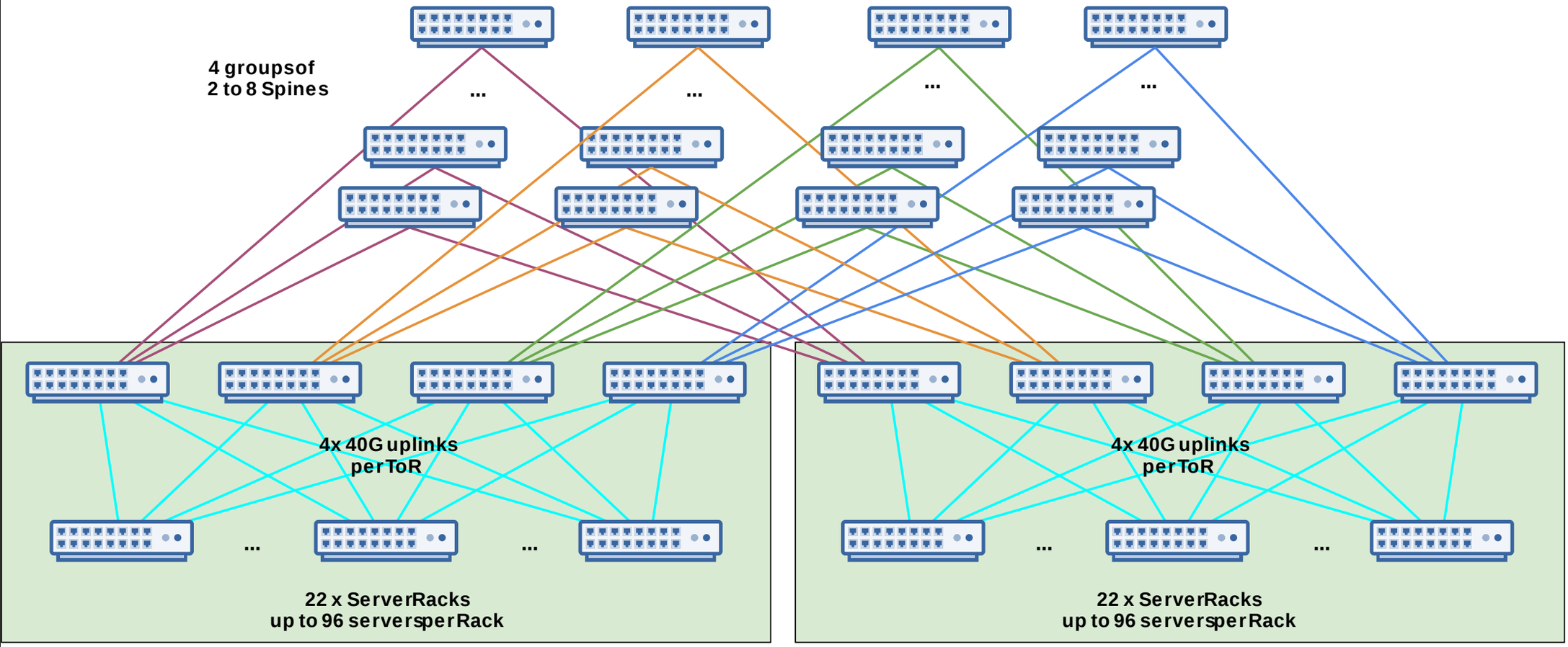
THIS TIME MULTI-YEAR PLAN

The cluster will start big and

it will grow …

THE PLAN

A NEW DATA-CENTRE WITH A NEW NETWORK



BUILD A NEW DATA-CENTRE IN 9 MONTHS

- it's impossible but we didn't know
- so we did it
- new constructor
- choose one server for everything
- choose a bad RAID card
- saturated 10 Gb/s links

CHANGE THE HARDWARE ...

call for tenders

three bidders

we bought three 10 node clusters

- 16 (or 12) 6 TB SAS disks
- 2 Xeon E5-2650L v3, 24 cores, 48 threads
- 256 GB RAM
- Mellanox 10 Gb/s network card

TEST THE HARDWARE ...

we tested:

- disk bandwidth
- network bandwidth

by executing

- teragen, with :
  - zero replication to saturate the disks
  - high replication (×9) to saturate network
- terasort

STRESS THE HARDWARE ...

load the hardware

similar performance

a manufacturer eliminated because

- 4 disks delivered broken
- other disks failed
- 20% electrical overconsumption

We choose the most dense and the least expensive → Huawei

HARDWARE MIX ...

- HP in existing data-centres
- Huawei and HP in the new data-centres
- Both in the old preprod cluster
- Huawei in the new preprod cluster

integrating several manufacturers is an investment

→ guarantees our liberty at each order

THE NEW MACHINES ARRIVE …

- We rack
- We configure using our Chef cookbooks
- Infrastructure is code in git
- Automate everything to scale
- System state in RAM (diskless)

A DIGRESSION: WHY DISKLESS?

+ Disks for the data not for the system
+ Chef convergence assured at each reboot
+ Manual operations are lost
+ Maximises storage density
− Longer reboots
− More infrastructure required for boot (Chef server, RPMs, etc.)
− 3 GB memory used for rootfs
− Nobody else at Criteo does it

We are going on-disk for master nodes

DON’T MIX O/S AND DATA ON THE SAME DISK

we did and it didn’t work

- HDFS is I/O bound
  - O/S traffic slows it down
- HDFS traffic is sequential
- O/S traffic is random
- mixing leads to extra seeks
  - disk failures

WE TEST HADOOP WITH 10 HOUR PETASORTS

- one job that uses all the resources
  - increase all per job limits
- one user that uses all the resources
  - increase all per user limits
- 250 GB application master
- trade-off between size of containers & number of containers

IT CRASHES

- Linux bug → namenode crashes
- Need to update kernel
  - On all 650 nodes
  - But we already have some users

ROLLING REBOOT WITHOUT DOWNTIME

- Good training
- Confirms reboot of a 650 node cluster is possible
- Found configuration errors at first reboot after build
- Rack by rack → OK
  - Jobs killed once
- Node by node → fail
  - Some jobs killed 600 times
- Applies to datanode and nodemanager restarts
- Now we can restart a 1100 node cluster without no impact

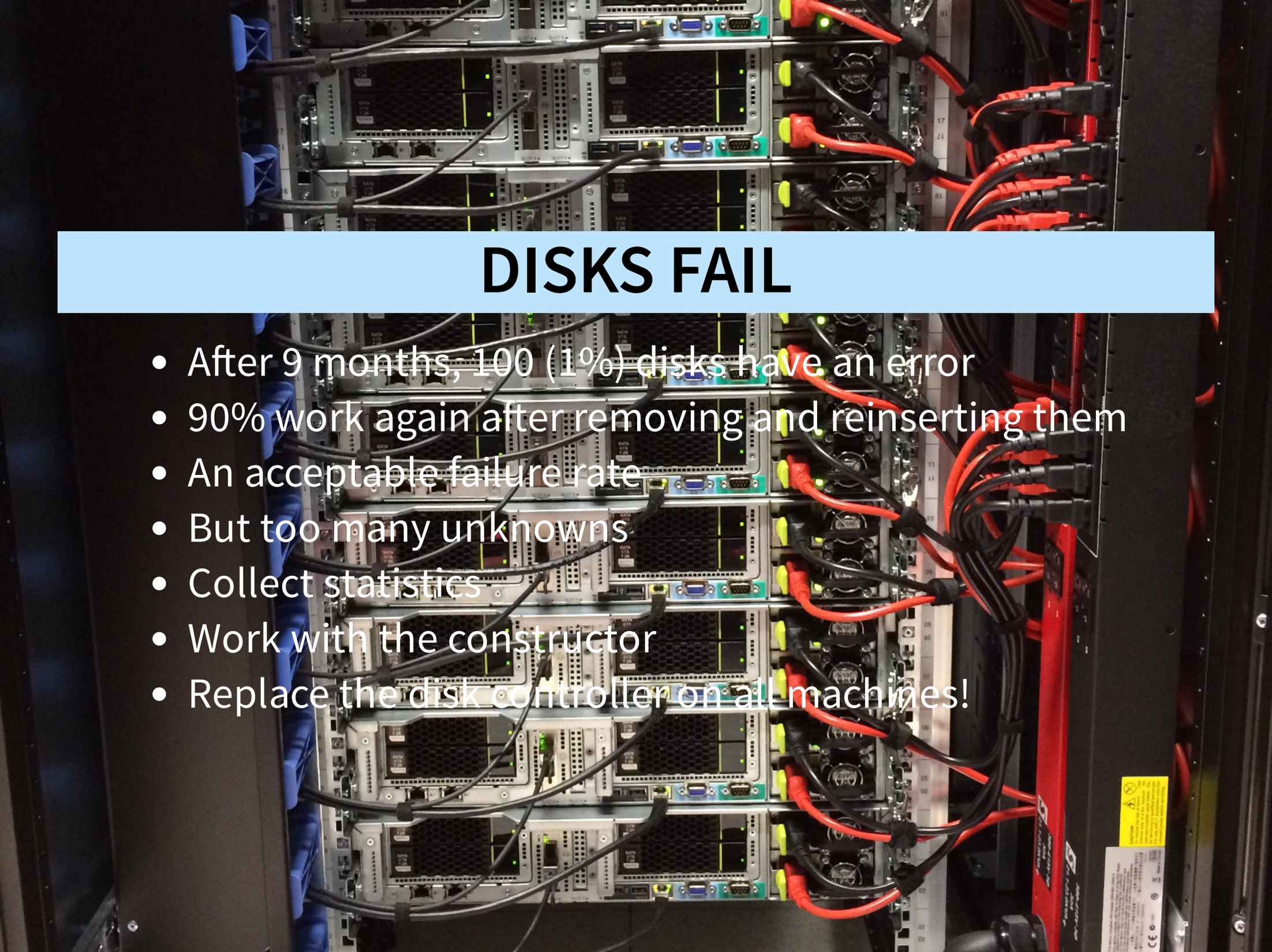
TUNE HADOOP CONFIGURATION

Lots of petasorts and petagens

Run as many jobs as possible

Adapt the configuration to the scale of our cluster

- namenode has 152 GB RAM for 238 266 000 blocks
- increase bandwidth and time limit for checkpoint



DISKS FAIL

- After 9 months, 100 (1%) disks have an error
- 90% work again after removing and reinserting them
- An acceptable failure rate
- But too many unknowns
- Collect statistics
- Work with the constructor
- Replace the disk controller on all machines!

DON’T PANIC

- Lots of disks and machines → problems expected
- 11000 disks → even rare errors will happen
- machines running at full charge 24/24 7/7 → even rare errors will happen

ADD A MANUFACTURER TO OUR PARK

The manufacturers are motivated

We need to:

- Understand the problems for their hardware
- Find solutions with their help
- Update the firmware (rack by rack reboot)
- Train the DevOps for interventions
- Estimate size of stock of spare parts

HADOOP IS ONLINE

- We need to help Criteo use the cluster
- The real problems start when the clients arrive
- The clients test and then we need to go live

MAKE THE NEW CLUSTER PROD READY

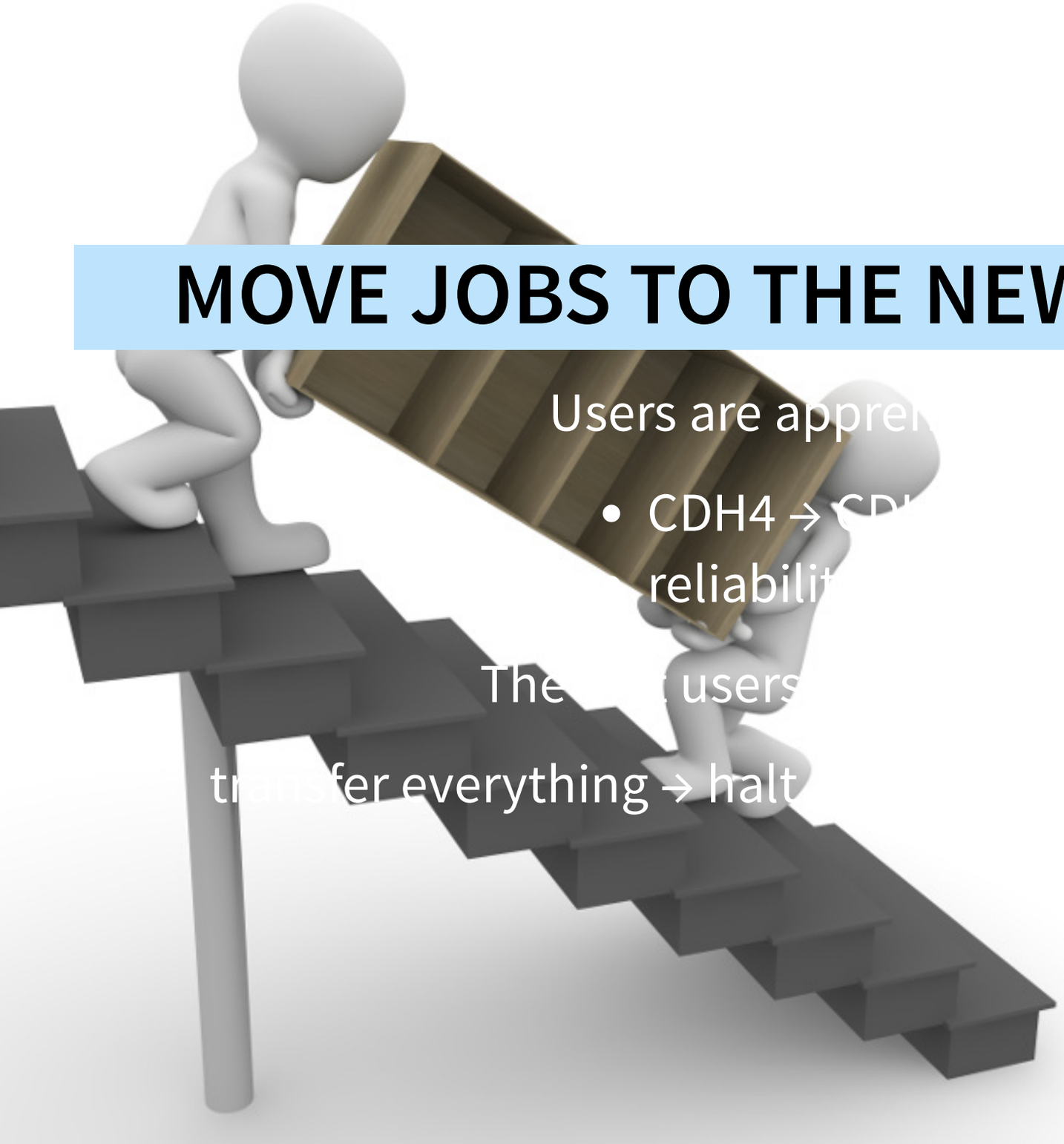
We copy 10 PB with a 20 Gb/s connection

- > 48 days all goes well
- But we have problems on the network side
- The copy is blocking prod traffic
- We have a routing problem (for a month)
- Workaround with Linux traffic control (tc) on each node

MONITOR EVERYTHING

- HDFS
  - space, blocks, files, quotas, probes
  - namenode: missing blocks, GC, checkpoints, safemode, QPS, datanode lists
  - datanodes: disks, read/write throughput, space
- YARN
  - job duration, queue length, memory & CPU usage
  - ResourceManager: apps pending, QPS
- zookeeper: availability, probes
- network
- hardware

MOVE JOBS TO THE NEW CLUSTER



Users are appre

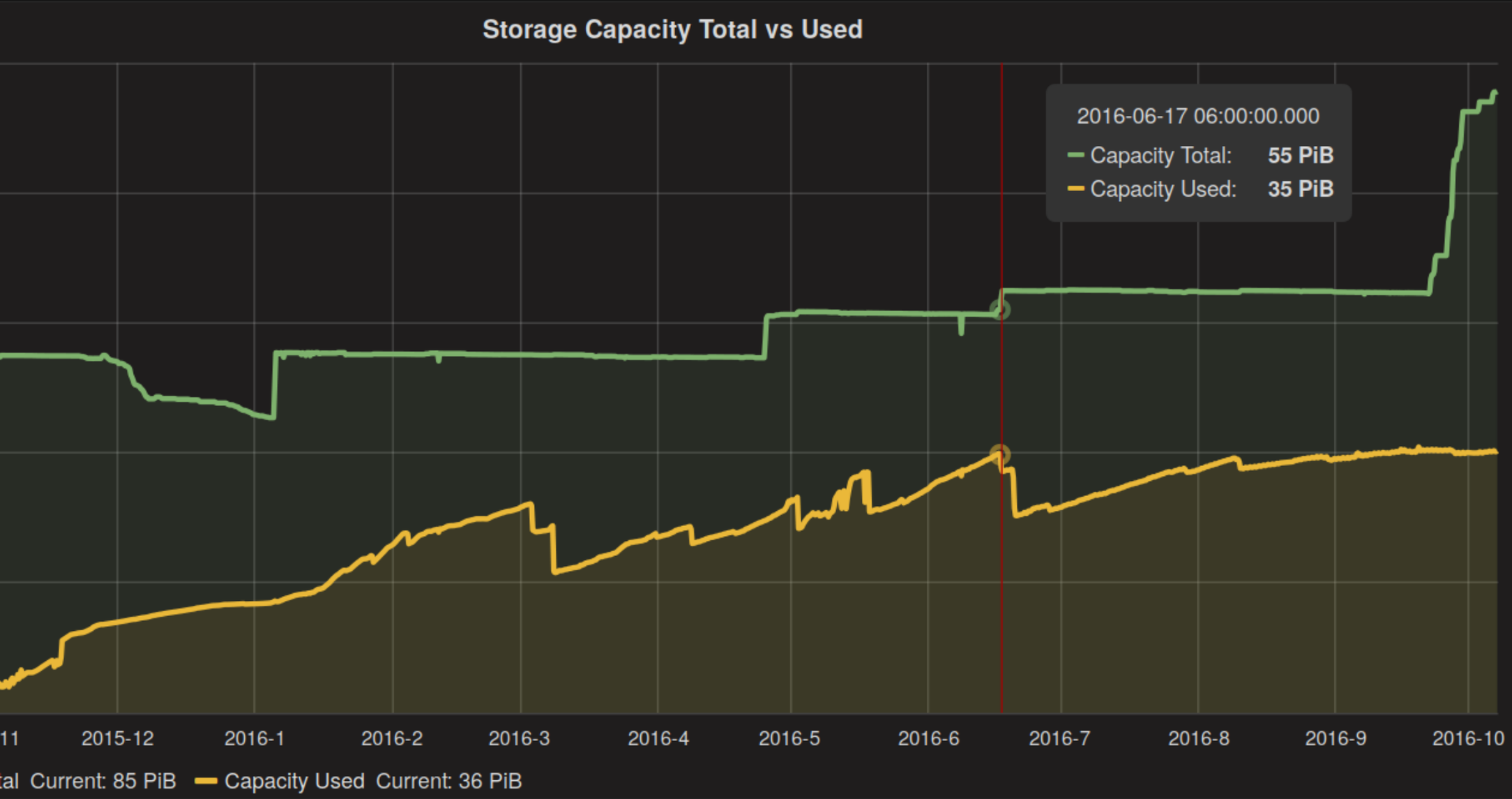
- CDH4 → CD
  reliabilit

The users

transfer everything → halt

MASSIVE MIGRATION FAILURE

- as they moved to the new cluster users found
  - and thus used more resources
- no more compute resources on new cluster before all jobs moved
- old cluster still running jobs
- some jobs moved back
- unable to buy more machines fast enough


Storage Capacity Total vs Used
2016-06-17 06:00:00.000
Capacity Total: 55 PiB
Capacity Used: 35 PiB
11
2015-12
2016-1
2016-2
2016-3
2016-4
2016-5
2016-6
2016-7
2016-8
2016-9
2016-10
tal Current: 85 PiB
Capacity Used Current: 36 PiB

ADDING DATANODES KILLS THE NAMENODE

- At ~ 170 million blocks
  - each block needs memory
    - GC (garbage collection) too long
    - datanode reports are delayed
    - datanodes are lost
    - block replication starts
    - datanodes return
    - extra blocks are freed
    - more GCs
  - → vicious circle

Cluster Summary

Security is *ON*
202426875 files and directories, 169674724 blocks = 372101588 total
Heap Memory used 139.59 GB is 91% of Commited Heap Memory [illegible] Max Heap Memory is [illegible]
Non Heap Memory used 125.43 MB is 98% of Commited Non Heap Memory 127.88 MB. Max Non Heap Memory is -1 B.

Configured Capacity	:	84.84 PB
DFS Used	:	35.71 PB
Non DFS Used	:	3.79 TB
DFS Remaining	:	49.13 PB
DFS Used%	:	42.09%
DFS Remaining%	:	57.91%
Block Pool Used	:	2 B
Block Pool Used%	:	0.00%
DataNodes usages	:	Min [illegible]
		0.00% 57.47% 60.20% 21.89%
Live Nodes	:	1022 (Decommissioned: 2)
Dead Nodes	:	10 (Decommissioned: 0)
Decommissioning Nodes	:	0
Number of Under-Replicated Blocks	:	4[illegible]

CHOOSE GC FOR NAMENODE

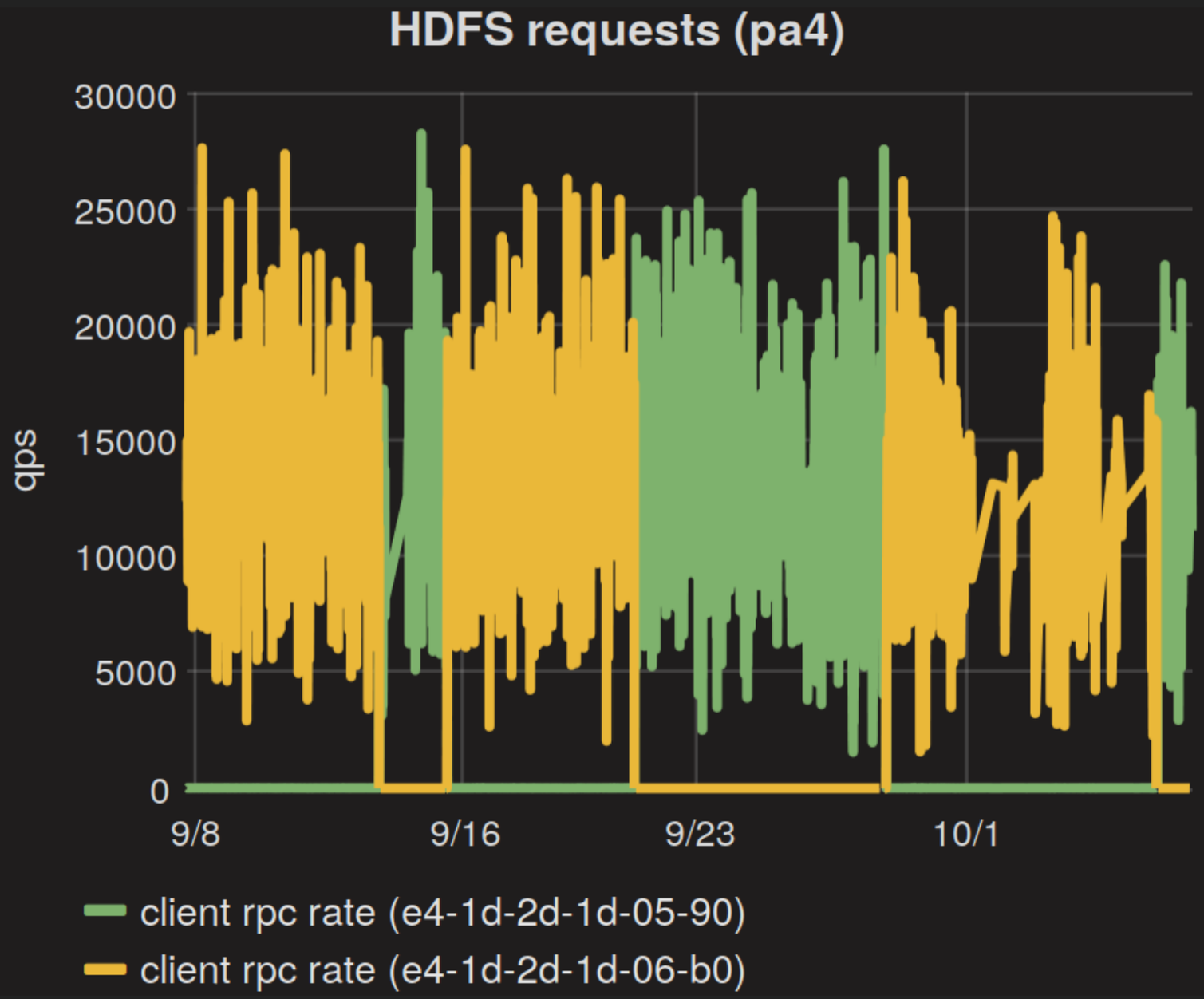
serial not efficient on multi-thread

parallel long pauses + high throughput

Concurrent Mark Sweep short pauses + lower throughput

G1 divided GC time by ~5

Azul under test


HDFS requests (pa4)
qps
30000
25000
20000
15000
10000
5000
0
9/8
9/16
9/23
10/1
client rpc rate (e4-1d-2d-1d-05-90)
client rpc rate (e4-1d-2d-1d-06-b0)

CURRENT SITUATION

Two different clusters in parallel

Where to run each job?

Which data to copy?

Open questions

Ad-hoc procedures

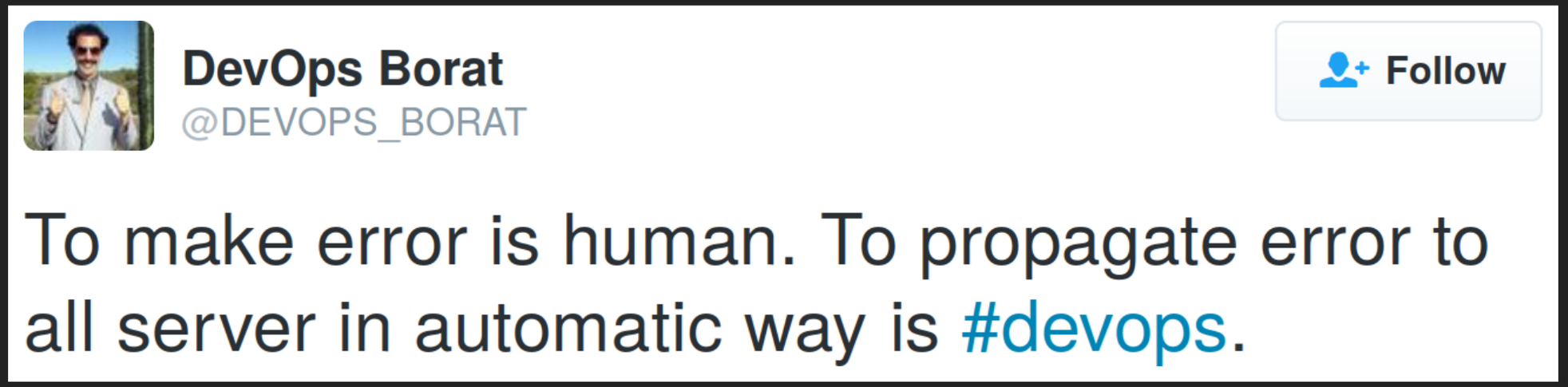
Was not meant to happen

HOW REDUNDANT ARE WE?

A single incident must not impact both clusters

But the same Chef code is used on both clusters

- Test >24 hours in pre-prod
- Rollout carefully



WHAT ABOUT OPERATOR ERROR?

A “fat-finger”

```
parallel -a prod_machines ssh {} sudo rm -r /var/chef/cache/lsi

parallel -a prod_machines ssh {} sudo rm -r / var/chef/cache/lsi
```

WE HAVE

- 5 prod clusters
- 3 pre-prod clusters
- 2 running CDH4
- 6 running CDH5
- 2395 datanodes
- 42 120 cores
- 135 PB disk space
- 692 TB RAM
- 200 000+ jobs/day

s.pook@criteo.com @StuartPook www.criteolabs.com