



# Distributed Systems

Jon Åslund

@jooon

# storage

## consistent hashing

# playlist

# playlist

## cassandra



Pitch  ck

playlist

structured key-value  
store

# playlist

## high throughput

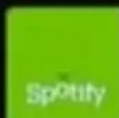
# playlist

choose redundancy and  
consistency level

playlist

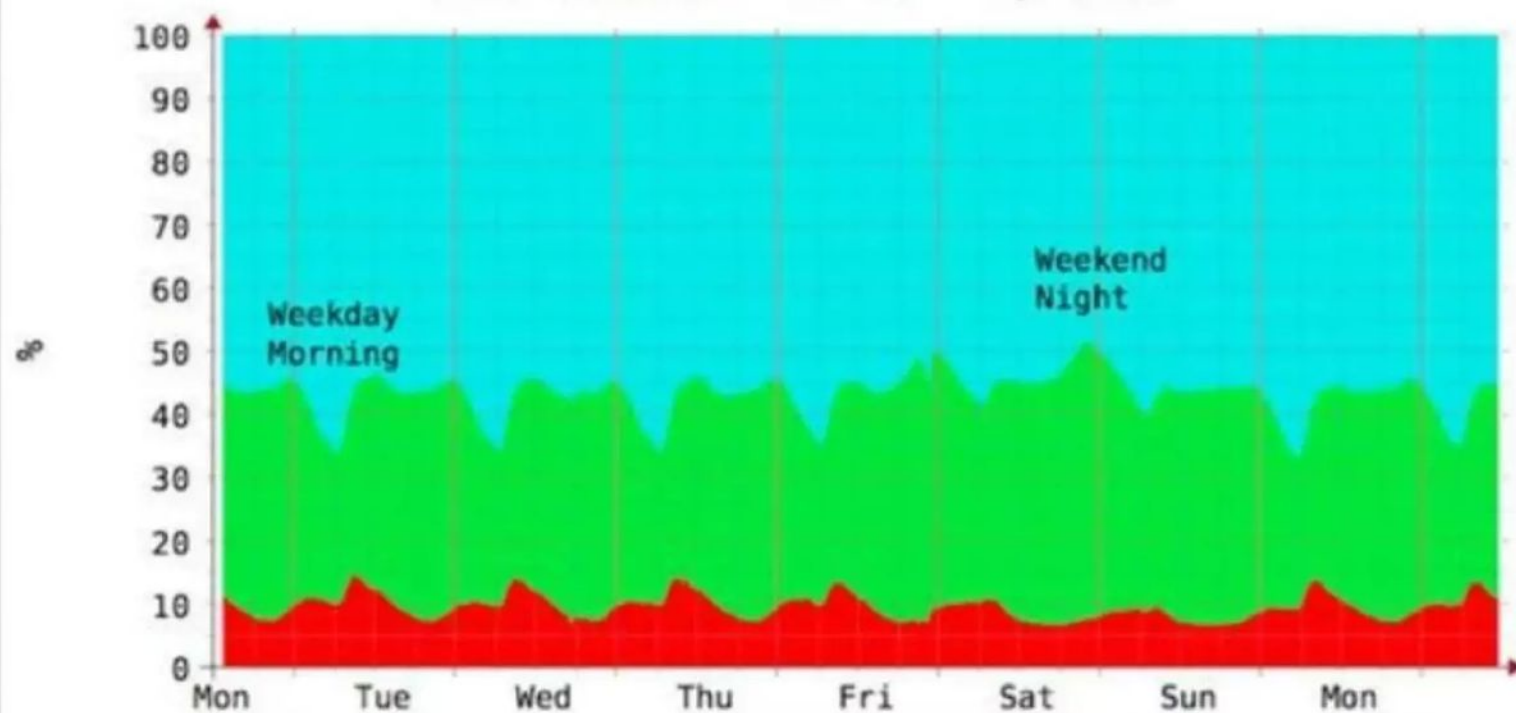
datacentre aware

p2p



Pitch

Data source - ratio - by week



■ Server  
■ P2P  
■ Cache

|       |       |       |
|-------|-------|-------|
| Cur:  | Min:  | Avg:  |
| 10.86 | 6.76  | 9.62  |
| 33.90 | 23.78 | 33.86 |
| 55.24 | 48.47 | 56.53 |

thank you

# Master Thesis?

Yes, please!

or come and work for us

Oh, and  
we have  
cake as  
well! :D

[spotify.com/jobs](https://spotify.com/jobs)

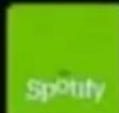


# Who am I?

# Jon Åslund



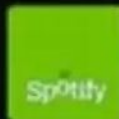
# Computer Science





# Shakespeare Programming Language

Joined Spotify  
Autumn 2006



# April 2006

## the idea



# Spotify™



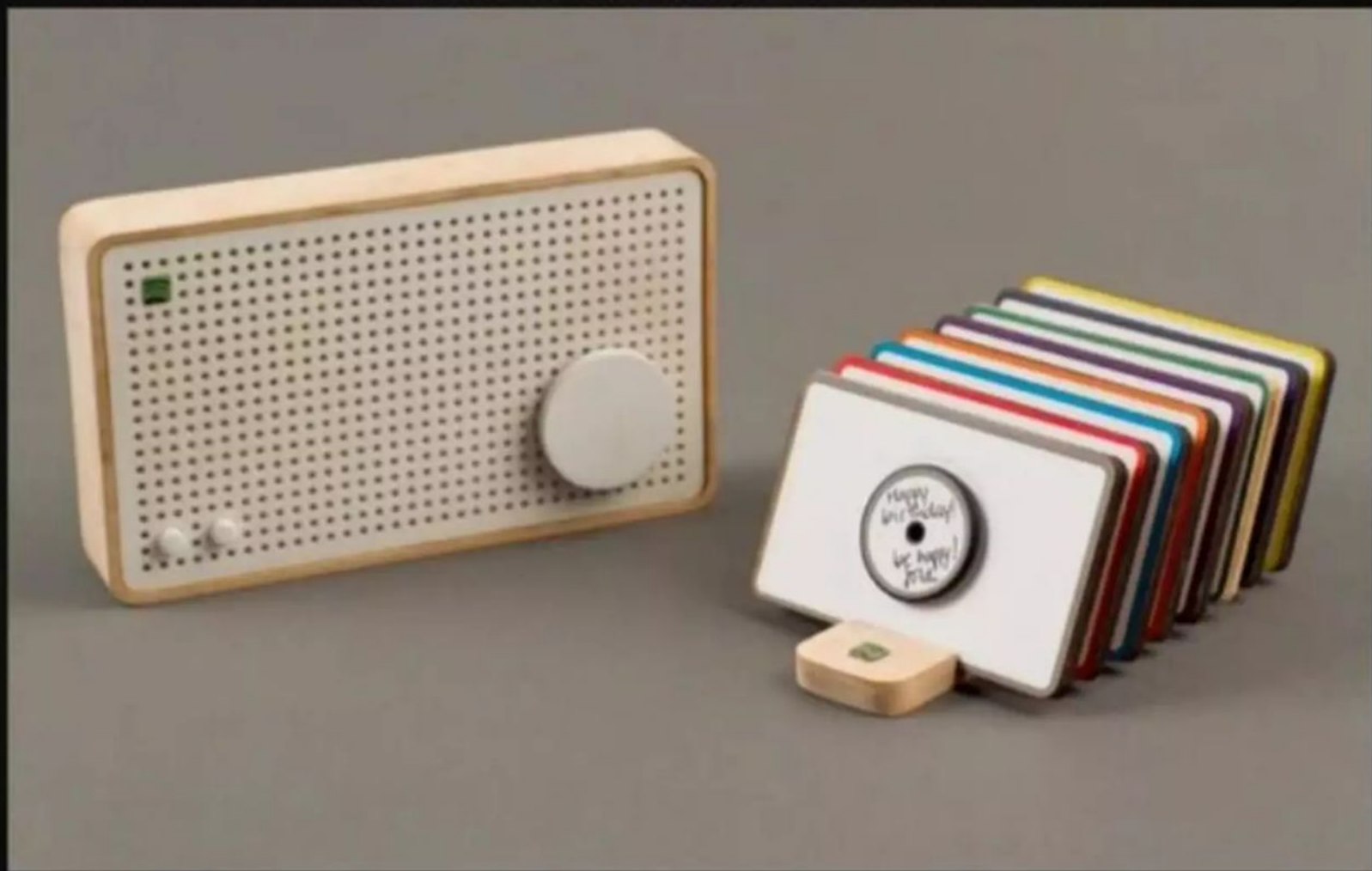
*“music itself is going to become  
like running water or electricity”*

David Bowie, 2002









# February 2012

**10+ million users**  
(active within previous 30 days)

15+ million tracks

500+ million playlists

1000+ servers  
in 3 countries

Spotify is a distributed  
system

Fast  
<200ms

# Always working



How do you  
accomplish that?

Cheat  
as much as possible

Fast  
vs  
Feels fast

Always working  
VS  
Pretend there is  
nothing wrong

# Build to fail

# Redundancy

(talk about waffles)



# Error recovery

Don't forget to test  
your recovery code

# Fire drill: Cut the atlantic cable

# Netflix Chaos Monkey

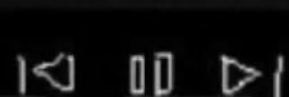
“On January 15, 1990,  
AT&T’s long-distance  
telephone switching system  
crashed.”

# Isolation

# Getting the right tool for the job



Spotify



PLAYING:

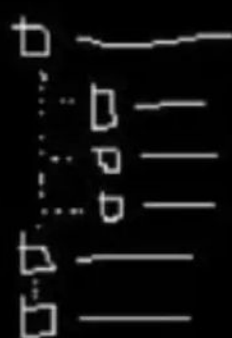
Bosse - finlåt

2.13

-X

Sök

Mitt  
BIBLIOTEK



Låtar & filer

Spellista

Sök

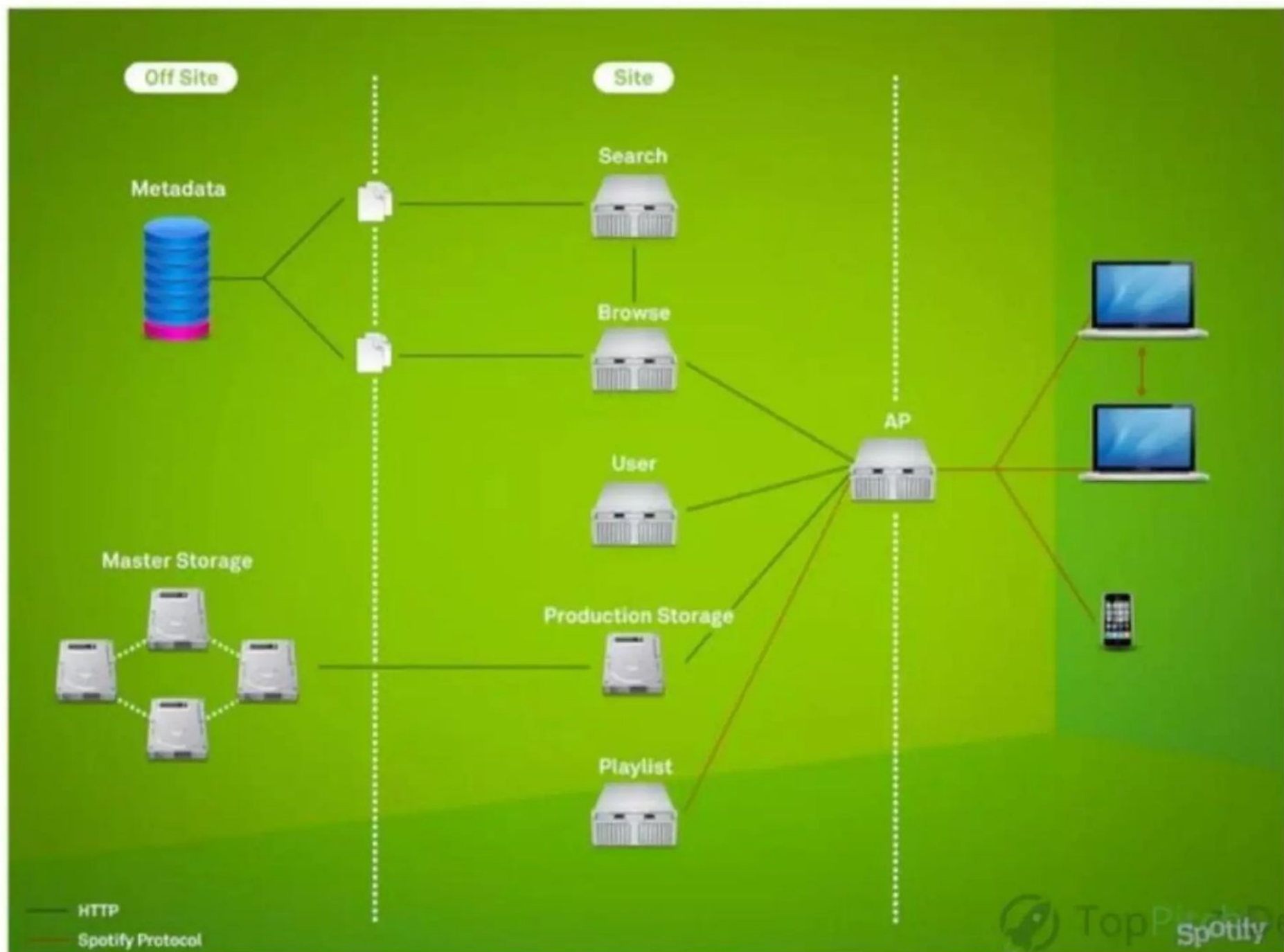
MODER-  
BIBLIO-  
TEKET

MER  
REKLAM



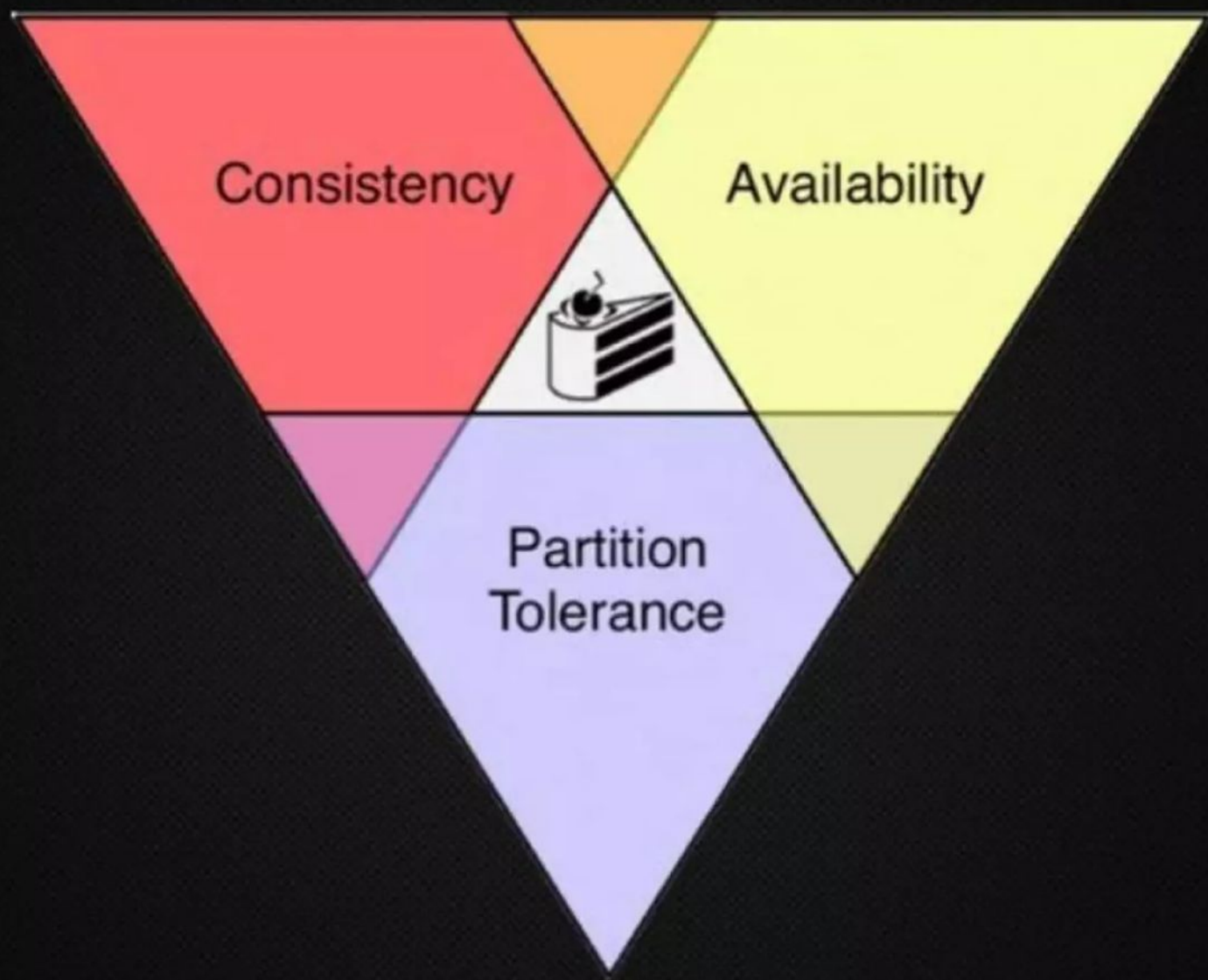
REKLAM/spoktar

Pitch Back



# Consistency Availability Partition tolerance

<http://ksat.me/a-plain-english-introduction-to-cap-theorem/>

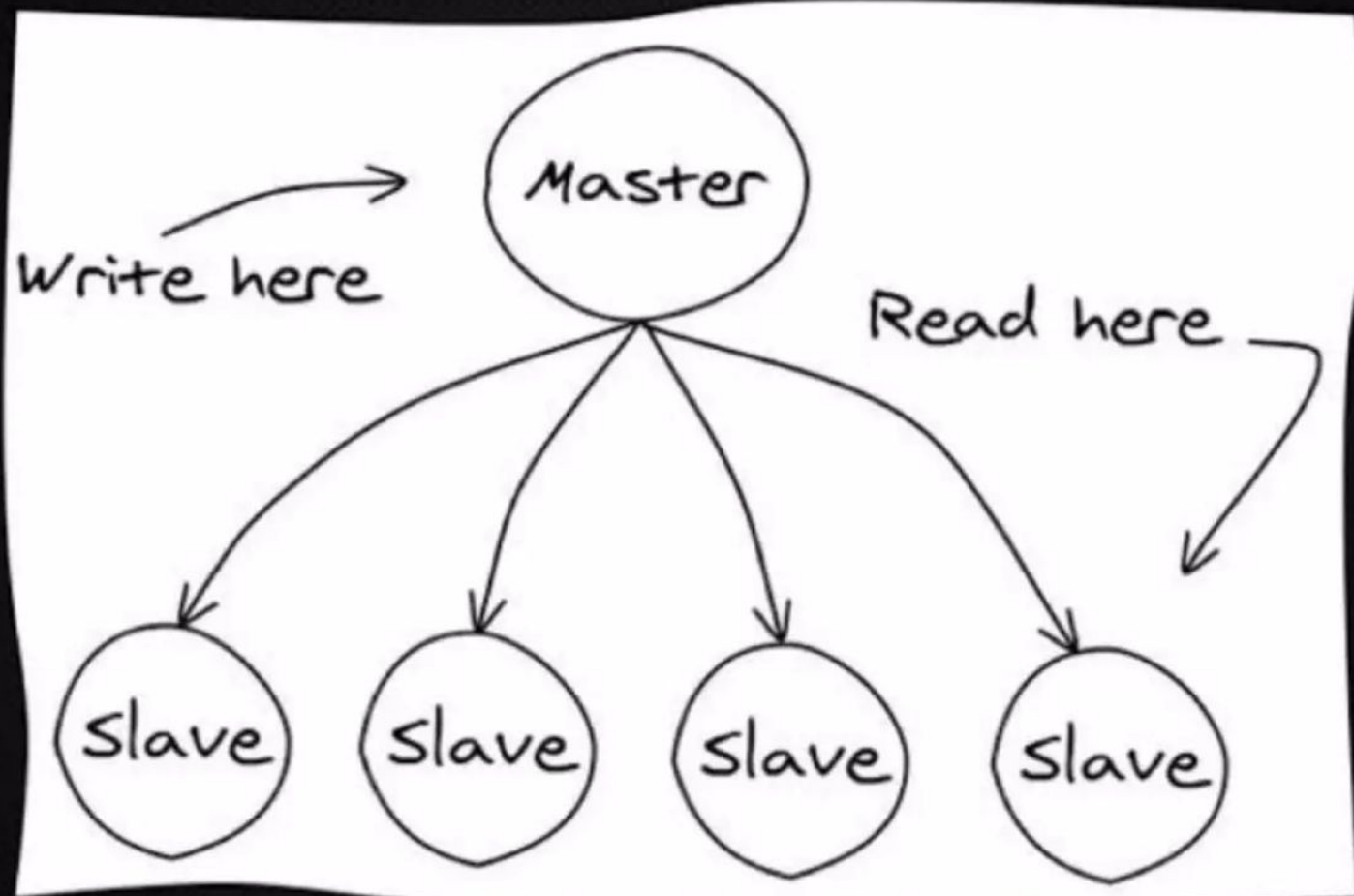


user

user

PostgreSQL

# user



search

search

# Lucene index in memory

search

replaced daily

search

read only

search

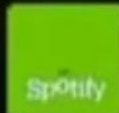
scales horizontally

search

sort of

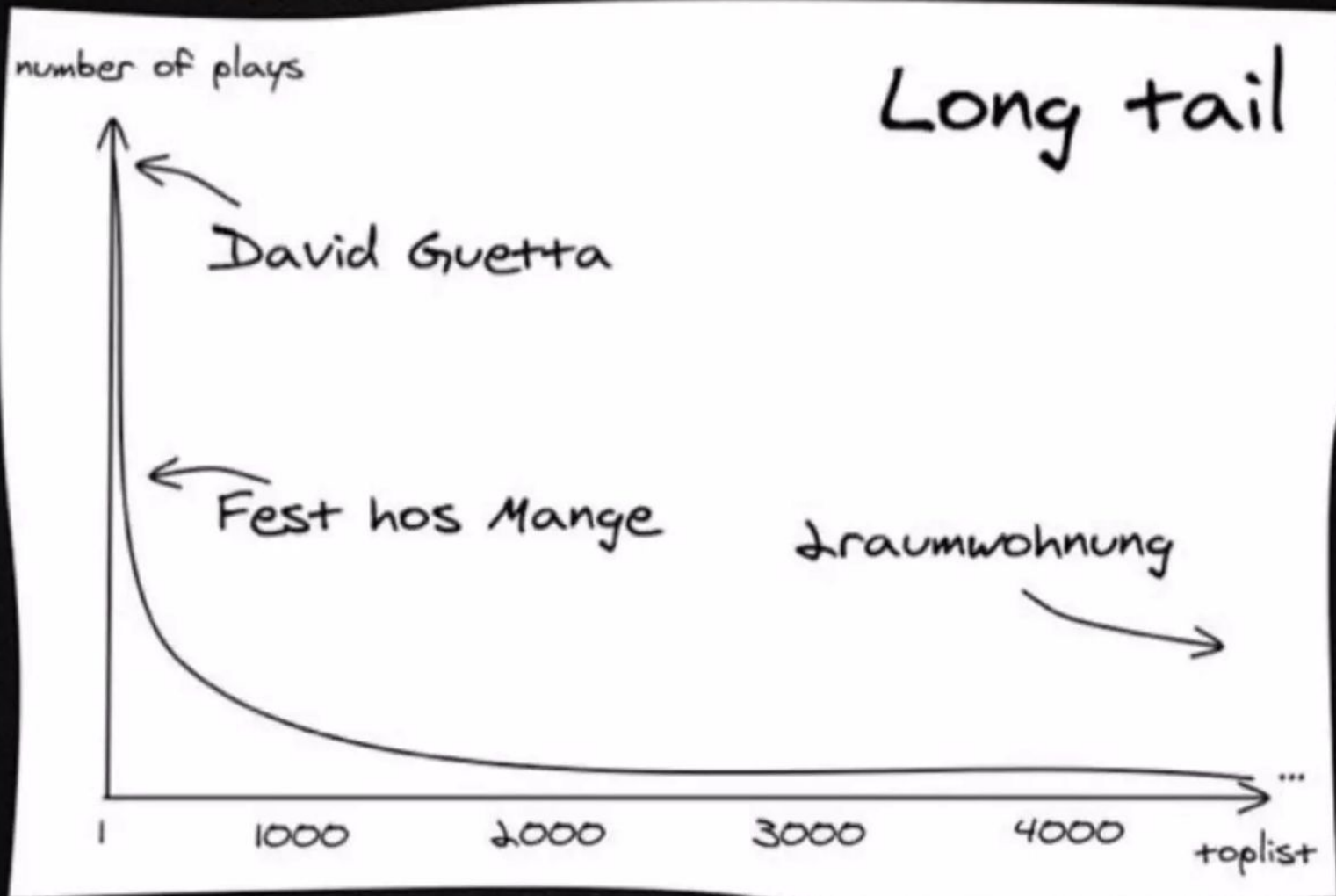
# search

How do you replace an index in memory when you are using all of it?



# storage

# storage



# storage

## lru cache in nginx

storage

better use of memory  
with sharding

## Storage

