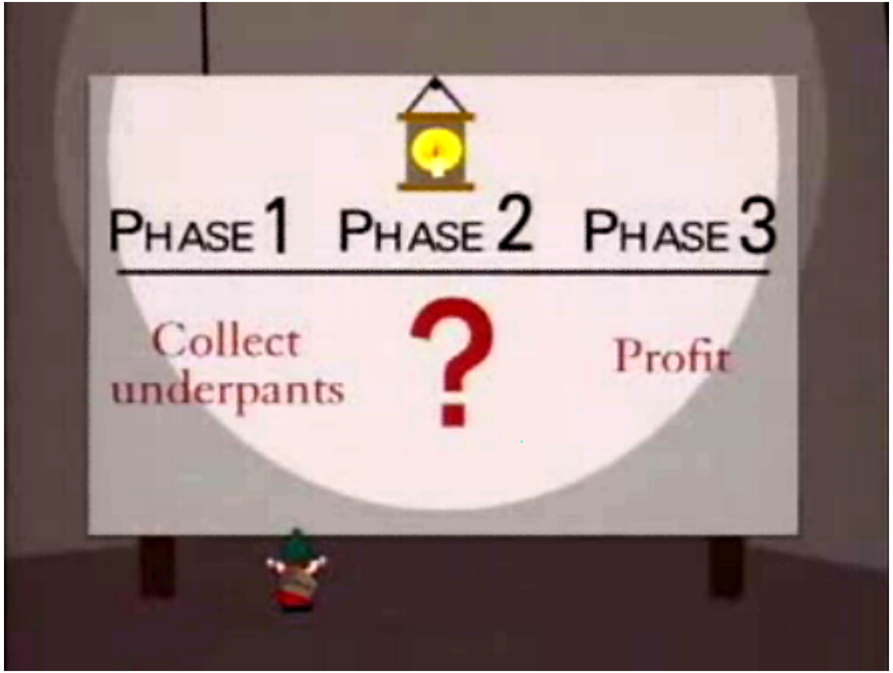
Organising for Data Success

Lars Albertsson
Data Architect, Schibsted Media Group

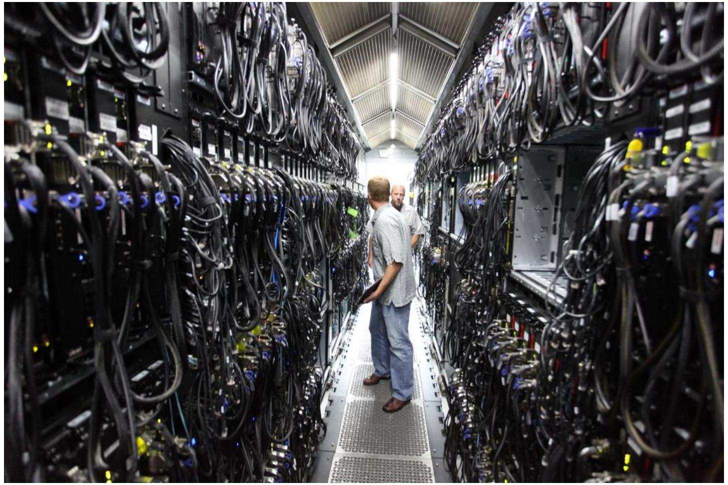
Bio

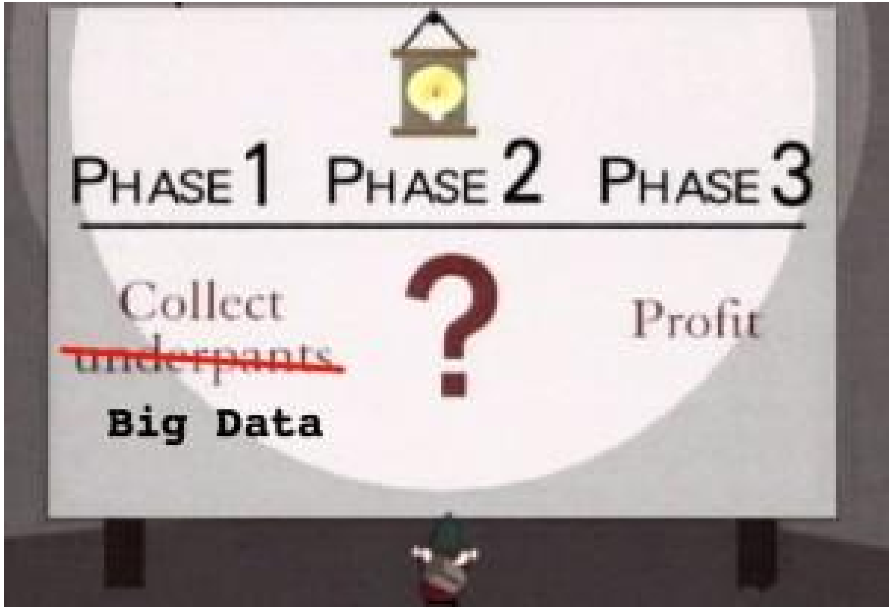
- SICS - test and debug technology for distributed systems
- Sun - high-end server verification
- Google - Hangouts, engineering productivity
- Recorded Future - data ingestion, data quality
- Cinnober - stock exchange engines
- Spotify - data processing, music data modelling
- Schibsted Media Group - data architect

Path to profit



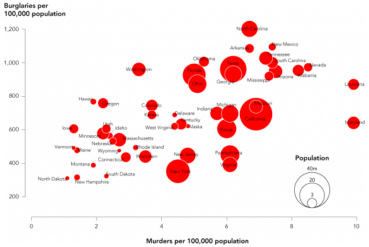
Big data path to profit



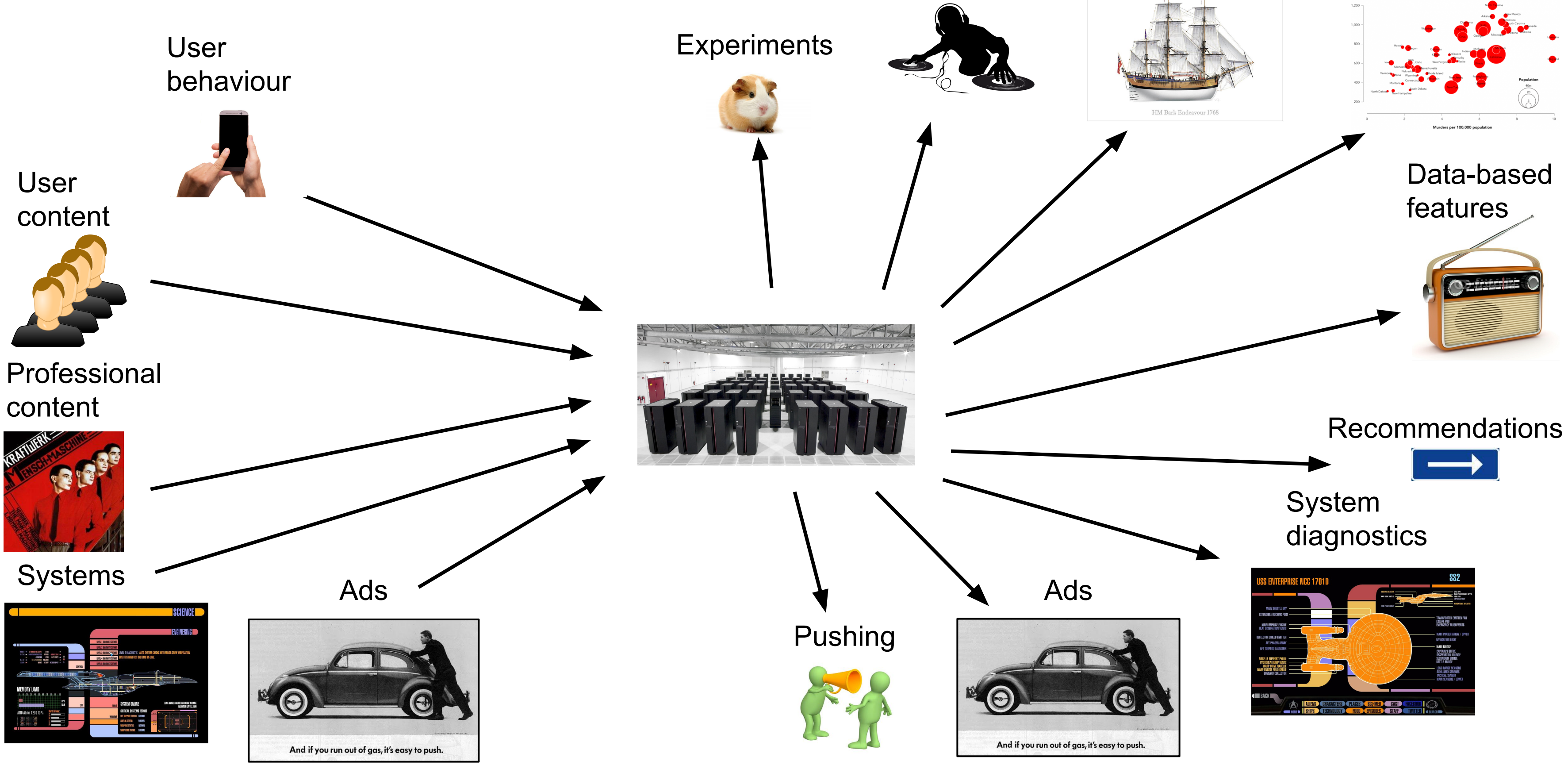


You start out simple

User behaviour



Things get complex



User behaviour

User content

Professional content

Systems

Ads

Experiments

Curated content

Exploration

Business intelligence

Data-based features

Recommendations

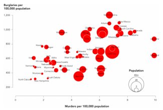
System diagnostics

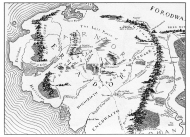
Ads

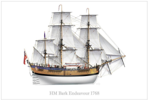
Pushing

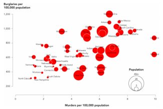
KRAFTWERK
DIE MENSCH-MASCHINE
SCIENCE
ENGINEERING
MEMORY LOAD
And if you run out of gas, it's easy to push.
HM Bark Endeavour 1768
Burglaries per 100,000 population
Murders per 100,000 population
Population
USS ENTERPRISE NCC 1701D
And if you run out of gas, it's easy to push.

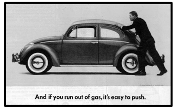
Presentation objectives

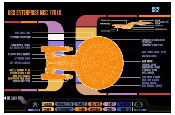










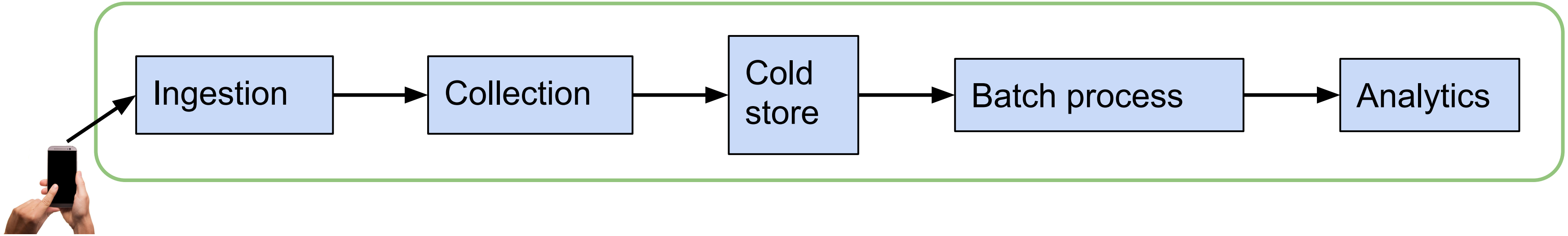


Conway’s law

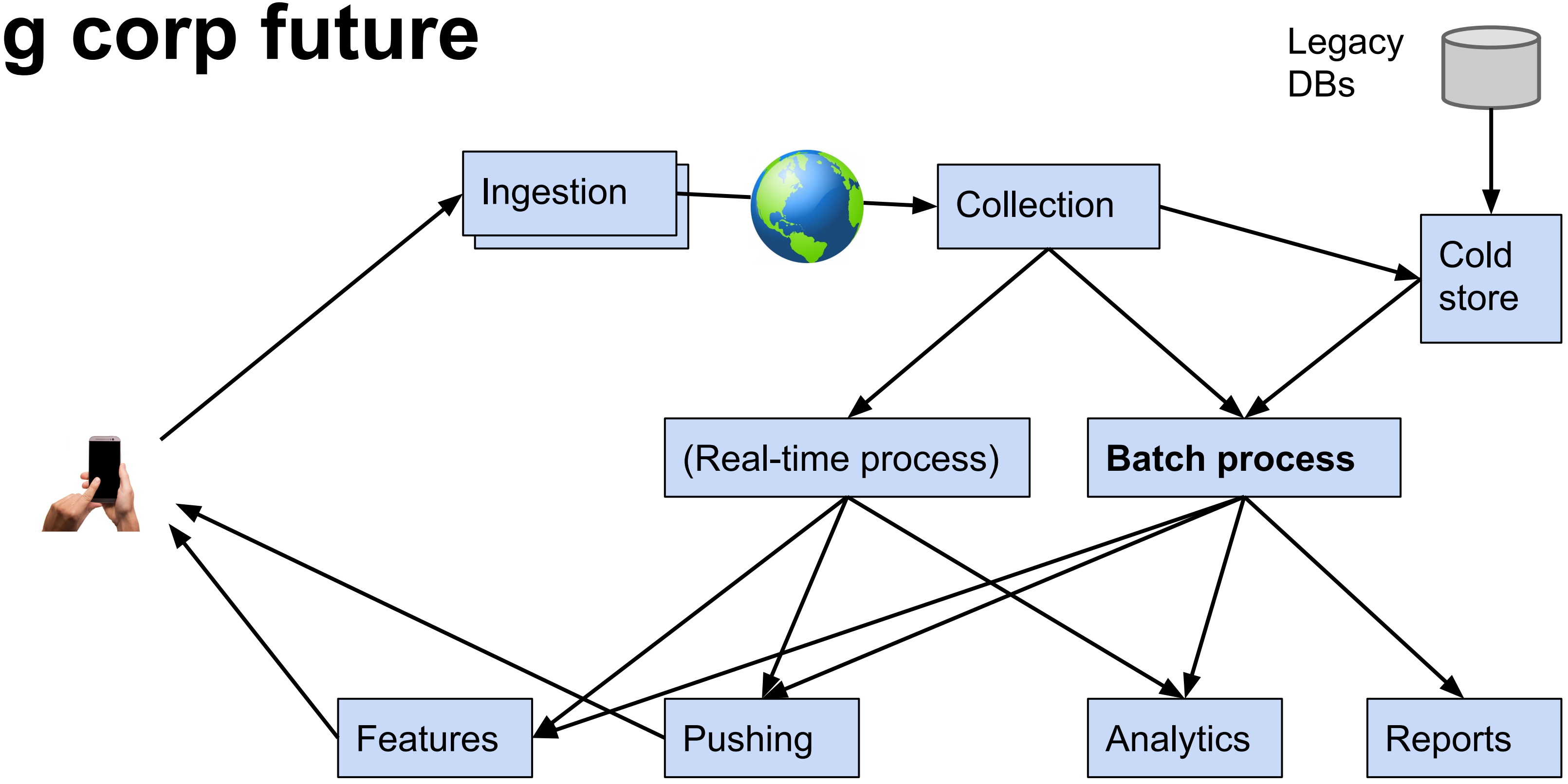
“Organizations which design systems ... are constrained to produce designs which are copies of the communication structures of these organizations.”

Better organise to match desired design, then.

Startup mode



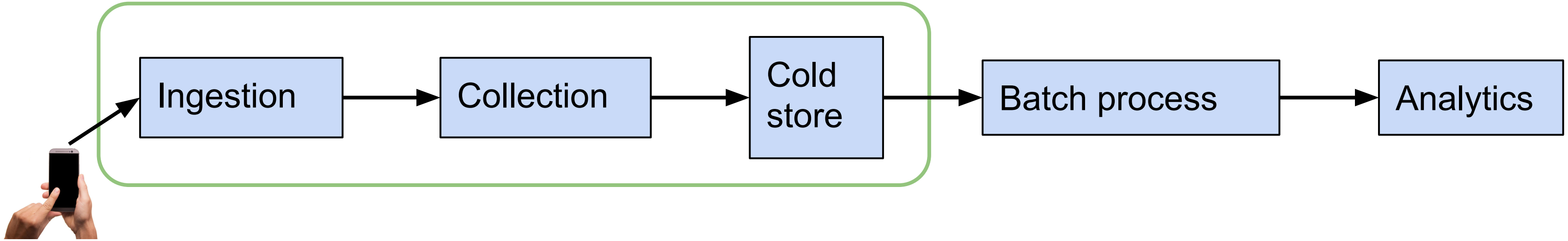
Big corp future



User visible - robustness

User hidden - agility

Data is gold

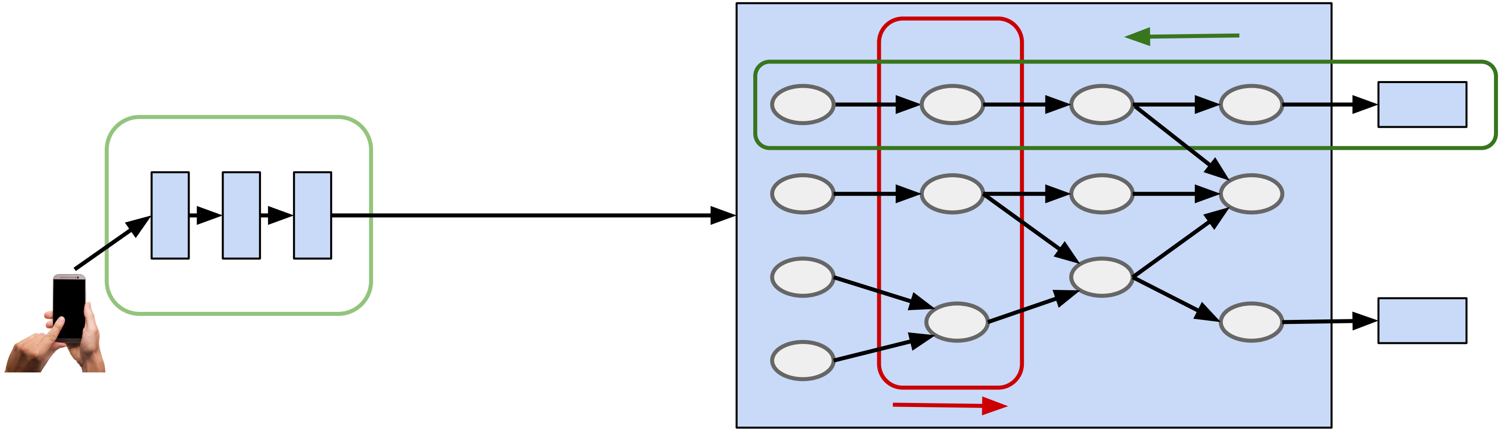


Don’t drop it - make it one team’s focus

Reliable path source -> cold store

- Minimal complexity
- Human & machine fault tolerance

Data pipelines

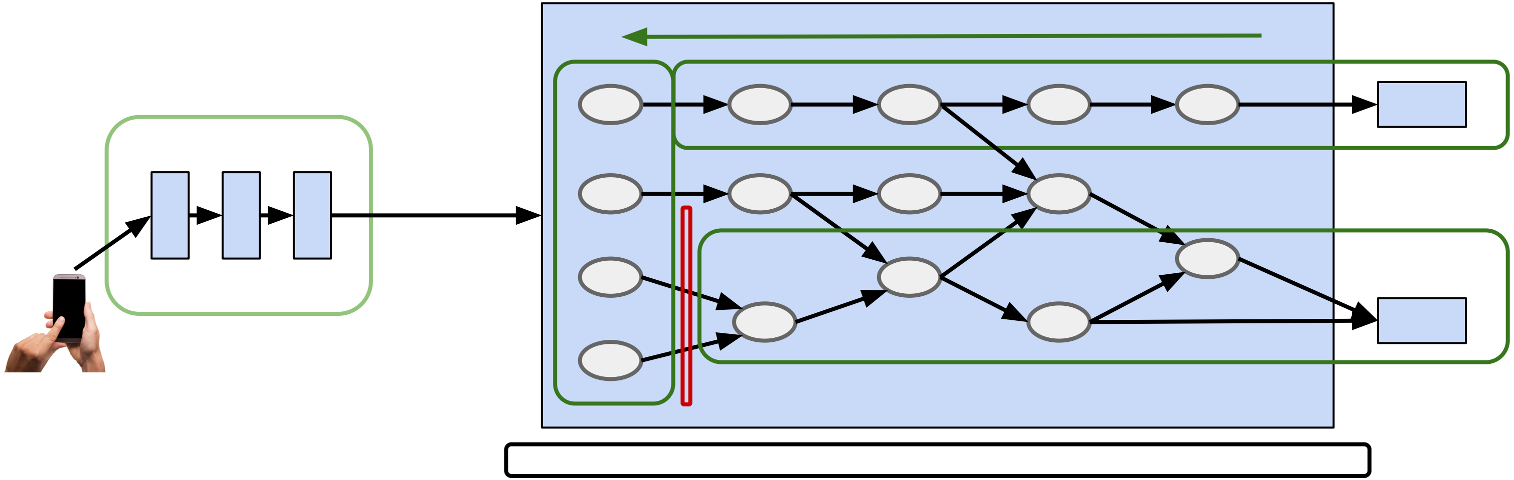


Form teams that are driven by business cases & need

Forward-oriented -> filters implicitly applied

Beware of: duplication, tech chaos/autonomy, privacy loss

Data platform, pipeline chains



Common data infrastructure

Productivity, privacy, end-to-end agility, complexity

Beware: producer-consumer disconnect

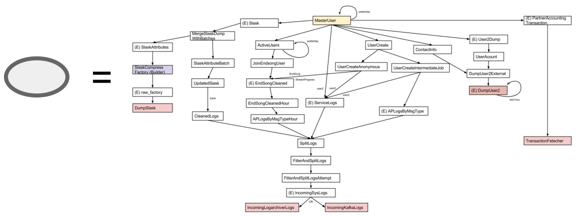
Example case: Spotify

~50M active users, 5-10 TB/day, 20PB

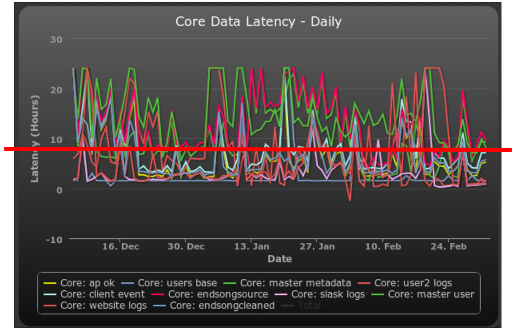
100-200 people touch data daily

Autonomous team and tech culture

Stabilising data platform



Morning coffee



- + Business-driven pipes, enabled teams
- - Productivity, end-to-end agility, privacy, stability, duplication, security

Example case: Schibsted Prod & Tech

10-200M users, 5+TB/day, 0-1PB

Blocket, Aftonbladet, Leboncoin, Finn, VG, ...

Grew 1-100 people in 1 year, 20 touch data

Big corp culture, governance

Fast-forwarded to platform stage, reverted to autonomy

+ Privacy, security, modern high-level components

- Productivity, stability, forward-driven, dependent teams

Survival utilities, technology

Heed ecosystem direction

Follow leaders

- Twitter, LinkedIn, Facebook, AirBnB, Netflix

Technology has no overlap with yesterday’s

Keep up

Survival utilities, ingestion

Data owners should export data

- Difficult, needs attention
- Pull database/API from Hadoop/Spark = DDoS

Quickly hand off incoming data to reliable storage

Measure loss and latency

Survival utilities, workflow

- Productive workflow from day one
  - Upstream easily breaks downstream
  - No off-the-shelf tools
- Privacy strategy from day one
  - Data spreads like weed
- Expect machine and human error
  - Capability to rebuild from cold store

Parting words

1. Keep things simple

2. Don’t drop data

3. Focus on productive developer workflows

4. Choose right components
   Open source is safer
   Avoid rolling your own

Bonus slides

Personae - important characteristics

Architect

- Technology updated
- Holistic: productivity, privacy
- Identify and facilitate governance

Backend developer

- Simplicity oriented
- Engineering practices obsessed
- Adapt to data world

Data scientist

- Capable programmer
- Product oriented

Product owner

- Trace business value to upstream design
- Find most ROI through difficult questions

Manager

- Explain what and why
- Facilitate process to determine how
- Enable, enable, enable

Devops

- Always increase automation
- Enable, don’t control

Cloud or not?

+ Operations
+ Security
+ Responsive scaling
- Development workflows
- Privacy
- Vendor lock-in