

# Thou Shalt Not Randomly Partition

Accidental Complexity Illustrated

Swiss Streaming Meetup | Oct' 21 | Julien Perrochet @ OSAG

# Quick BIO

- Software Engineer
- EPFL Communication Systems '13
- Functional Programming & Build Systems (Scala/Bazel)
  - Go, Python, Java, Typescript
- Principal Software Engineer @ Open-Systems AG<sup>[1]</sup>

[1]: <https://www.open-systems.com/>

# Agenda

- Context
  - Story of a migration
  - Particular Industry
- A Short Adventure
- Lessons
- Q&A

# Disclaimer

- Not a fancy/advanced user
- Basic use cases

# Context

- From Enterprise Messaging to Kafka For Data Transport
  - Partition, Broker, Replica: new concepts
- Cloud + Edge
  - We Ship Our Own Hardware
  - Thousands of Devices
- (Semi-)Embedded code
- Multiple Languages
  - Perl & Go on the producer side
  - Go, Scala on the consumer/transformer side (+ some kafka streams)

Your Own Adventure

# Your Own Adventure

“Messages I get are out of order?!”

# What do you do?

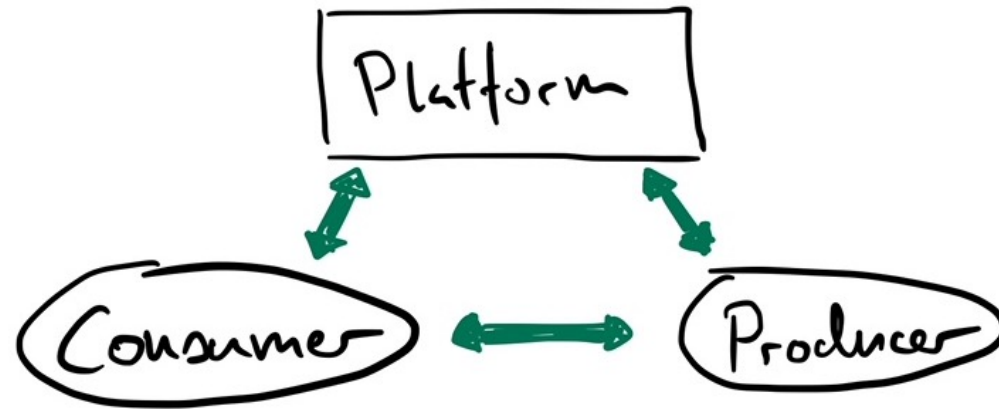
- Is it expected?
- If not: where is the issue?



# What do you do?



Ideally, talk to producers!



# Who's the Producer?

- Is it you/your team? Problem Solved!
- A Sister Team? Probably Fine.
- Random Team in the same department? Getting hairy.
- Does anyone know, actually?

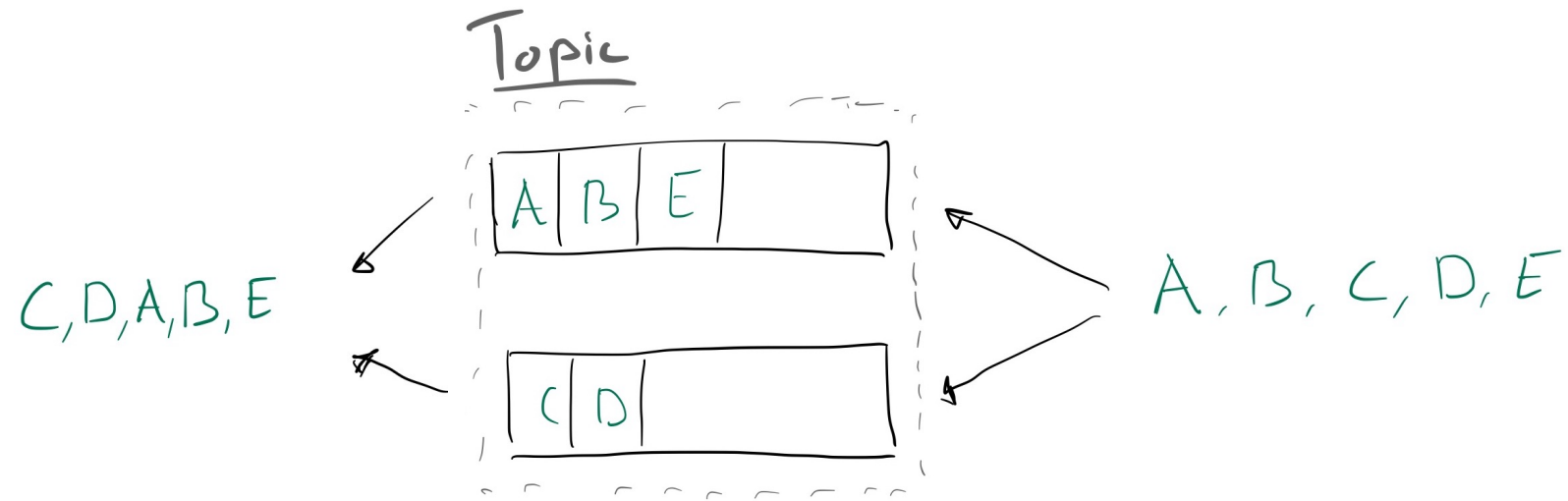
Harder with tight deadlines, understaffing, turnover, ...

# The Plot Thickens

# The Plot Thickens

“I just write metrics with a timestamp to the topic: *everything I emit is ordered.*”

# Your Own Adventure



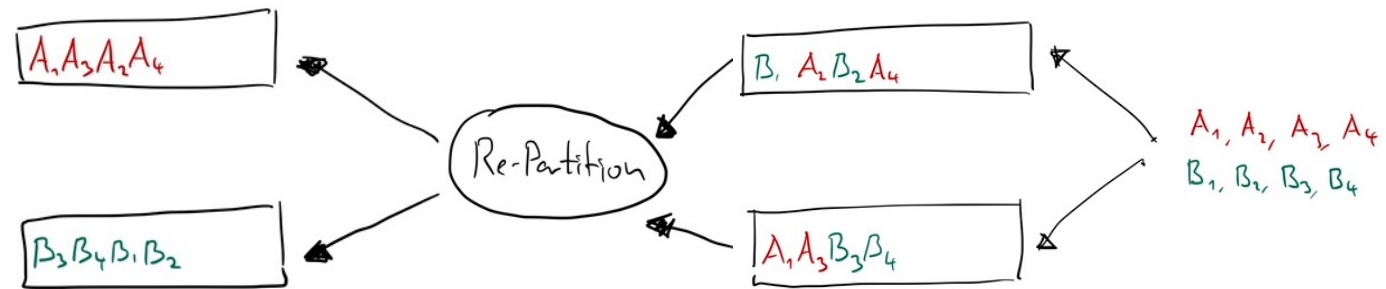
Yielding to the temptation

# Yielding to the Temptation

“We can just repartition!”



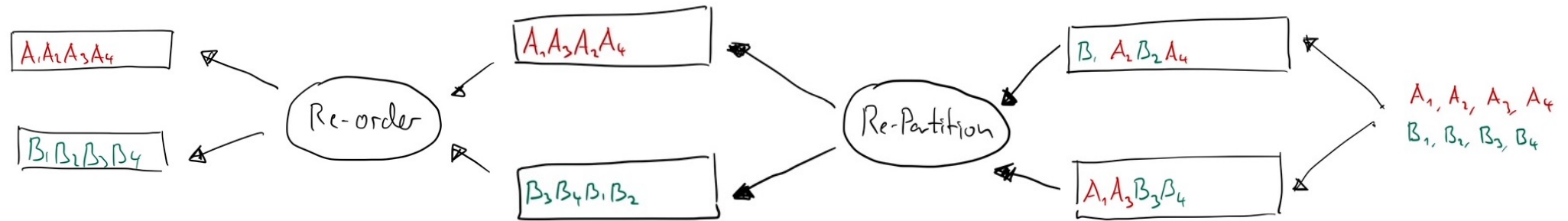
# Yielding to the Temptation



# Yielding to the Temptation

“They’re still out of order!”

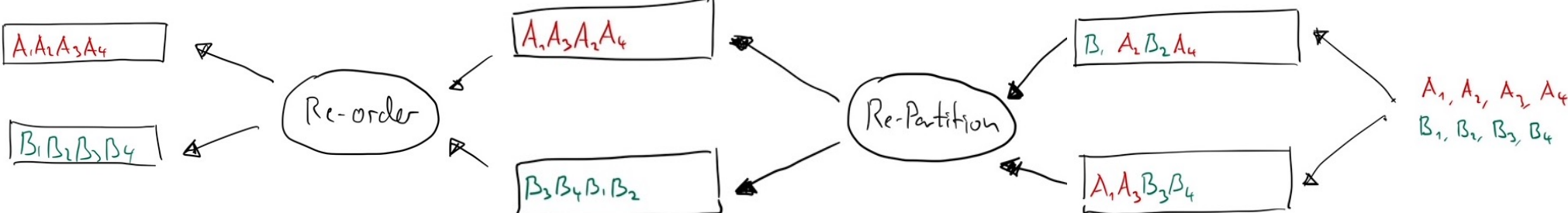
# Yielding to the Temptation



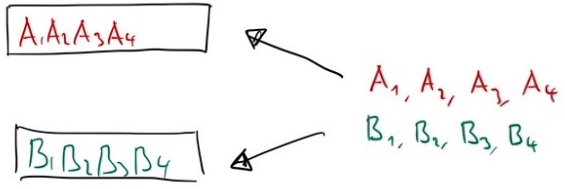
# Downsides

- Latency
- Unneeded Added Complexity
- Additional Resource Usage

# Downsides



VS



# Main Lesson

# Main Lesson

“Obvious technical bad designs may arise for non-technical reasons.”

# Lessons

- Unevenly spread or deprecated knowledge
  - Across Organisation(s)
  - Across Time
- Education, Curiosity
- Multiple Touch-points between Teams
  - Consumers & Producers contacts
  - Wider code review
- Anti-footgun libraries
- Organisational structure matters



# Executive Summary

# Executive Summary

“/dev/null beats random partitioning”

# Shameless Plug

## We're Hiring!

- (Senior) Software Engineers
  - Data Engineering
  - Build/Release/Ops Engineering
  - Data Products & Services
- (Senior) Frontend Engineers
  - With interest in the whole stack
  - With a product mindset

# Q & A

*Get in touch:*

- [jperrochet@open-systems.com](mailto:jperrochet@open-systems.com) / [julien@sqooba.io](mailto:julien@sqooba.io)
- [@0xTelegraph](#)
- [j3t.ch](#)