FROM CRUD TO EVENT SOURCING

Why CRUD is the wrong approach for microservices

James Roper

@jroper

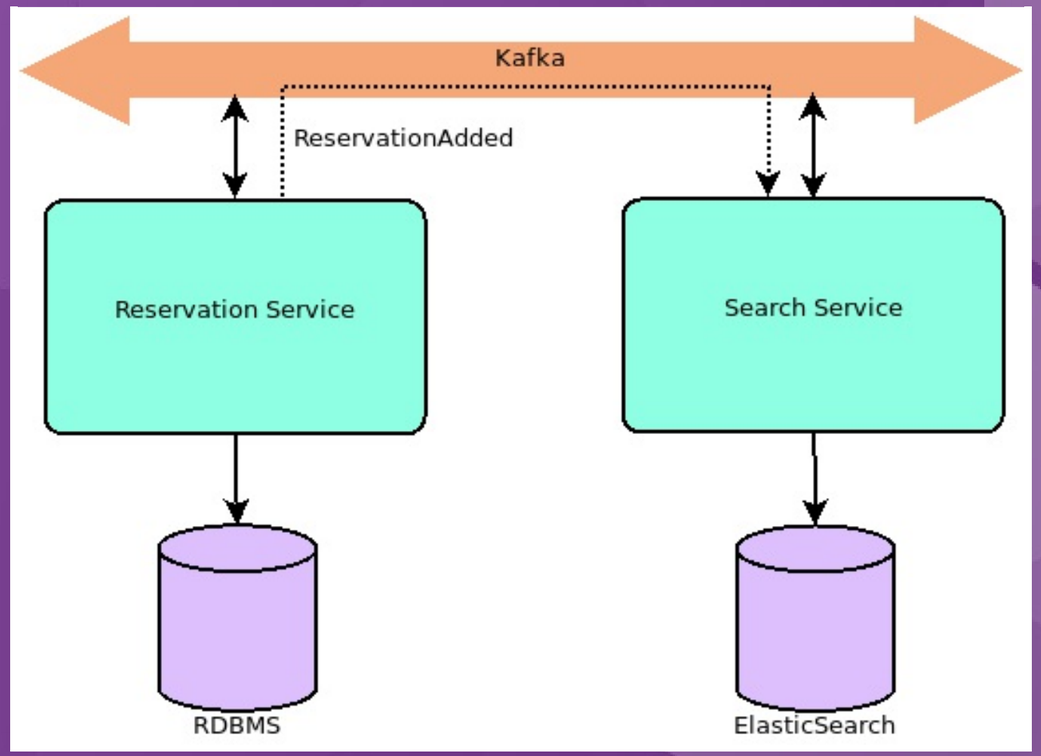
AGENDA

- Failure test CRUD solutions
- Learn about Event Sourcing
- See why Event Sourcing is resilient
- Live coding!

HOLIDAY RENTALS

- Airbnb like application
- Asynchronous microservice architecture
- Kafka for messaging
- Reservation service uses CRUD persistence

ARCHITECTURE

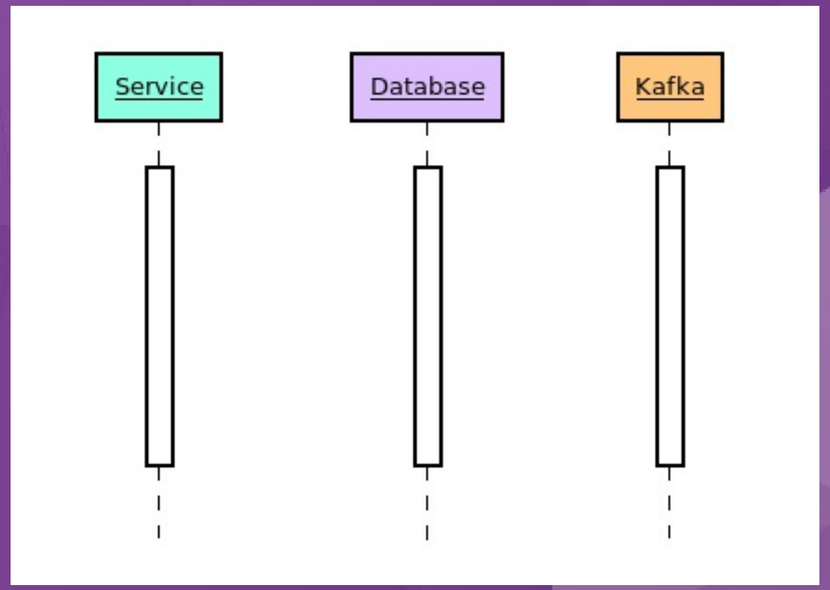


SERVICE IMPLEMENTATION

- Basic approach:
  - Update database

    and
  - Publish to Kafka

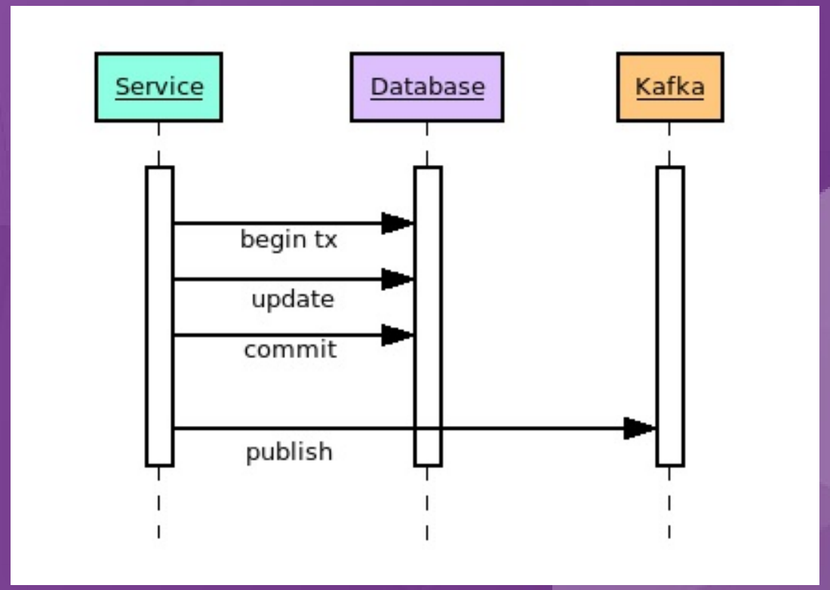
NAIVE APPROACH

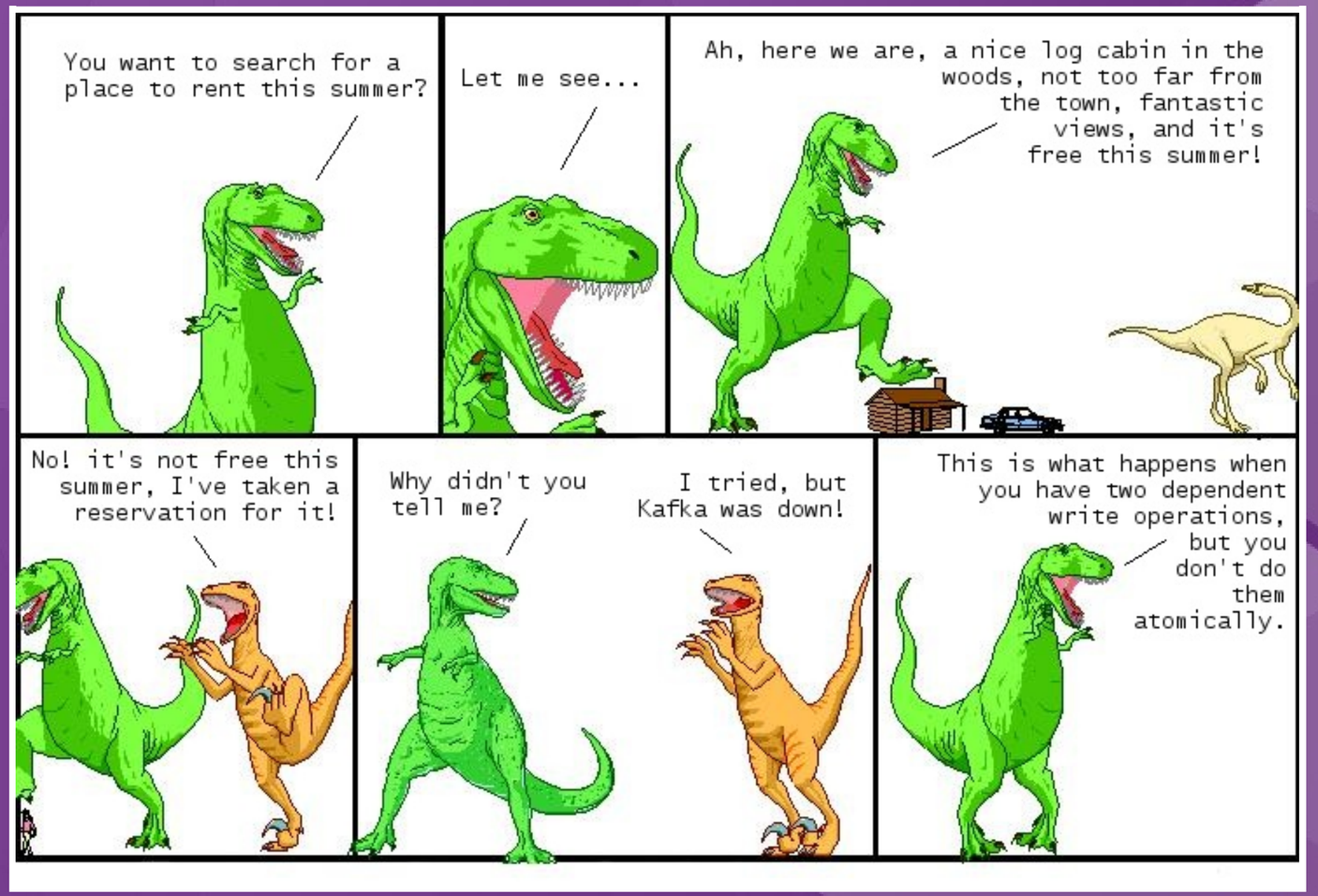


WHAT IF SOMETHING GOES WRONG?

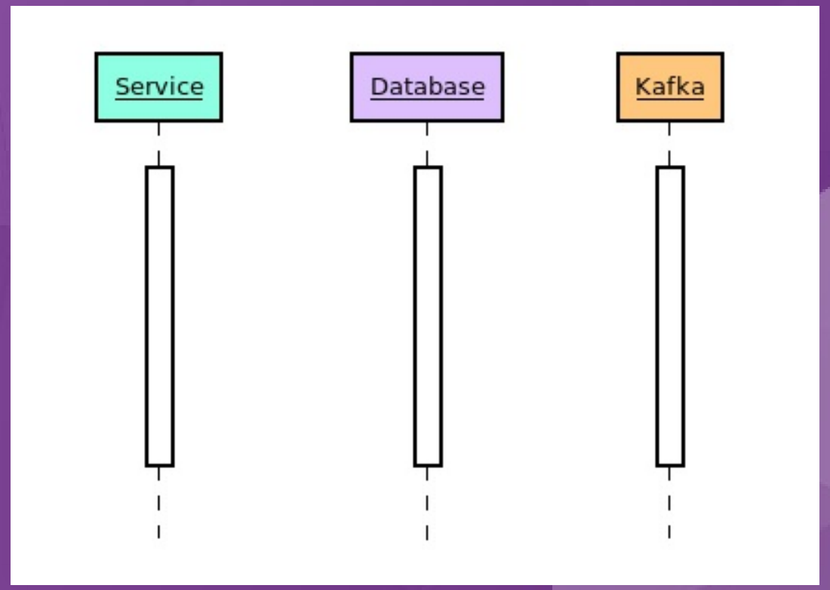
- Is this approach resilent?
- What if:
  - The service goes down?
  - The database goes down?
  - Kafka goes down?
  - The network goes down?

NAIVE APPROACH

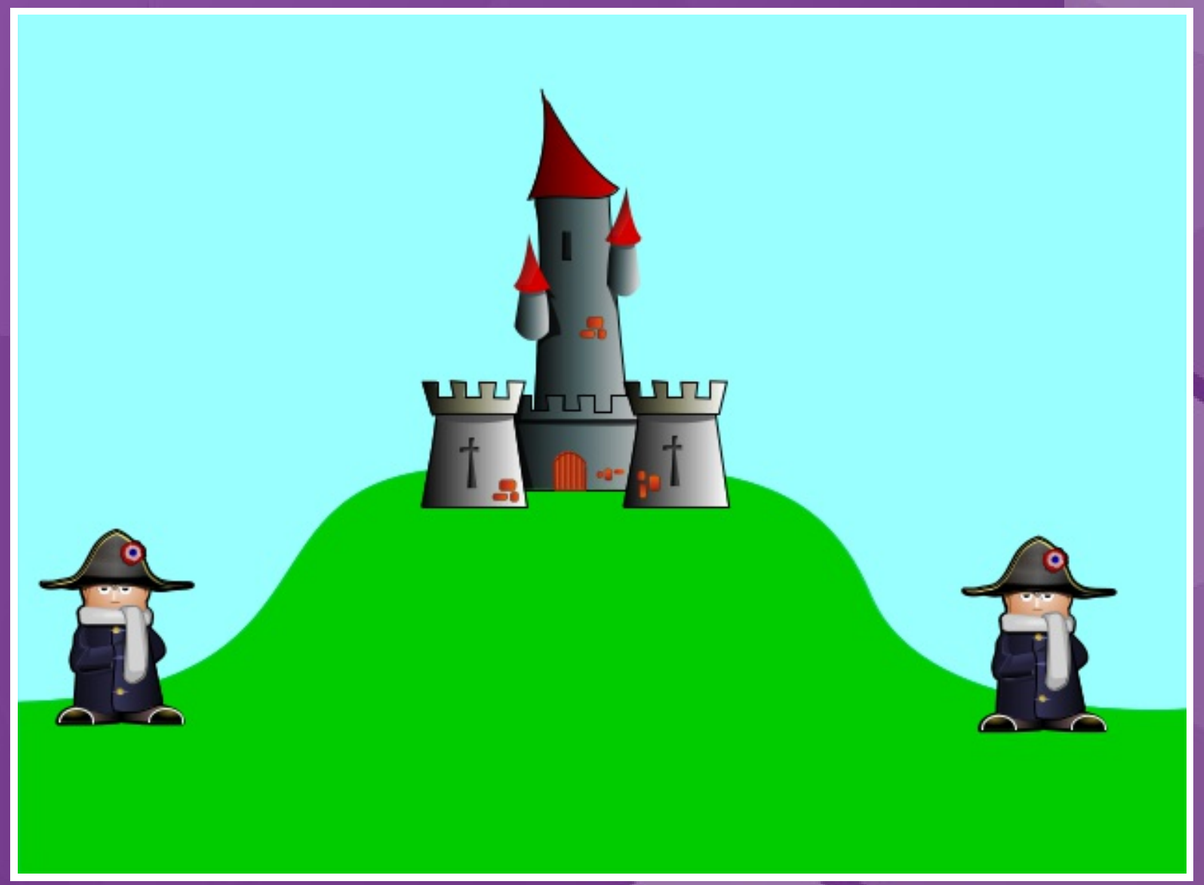



You want to search for a place to rent this summer?
Let me see...
Ah, here we are, a nice log cabin in the woods, not too far from the town, fantastic views, and it's free this summer!
No! it's not free this summer, I've taken a reservation for it!
Why didn't you tell me?
I tried, but Kafka was down!
This is what happens when you have two dependent write operations, but you don't do them atomically.

ANOTHER SOLUTION?



TWO GENERALS PROBLEM



TWO GENERALS PROBLEM

- It is proven to be unsolveable

TWO-PHASE COMMIT

- Can't we just use two-phase commit?
  - No.
  - Only reduces the window for failure to cause a problem
  - The window is largest when failure is most likely
  - Coordination comes at high cost

TOO HARD?

- Maybe it's all too hard
- Is inconsistency really that bad?

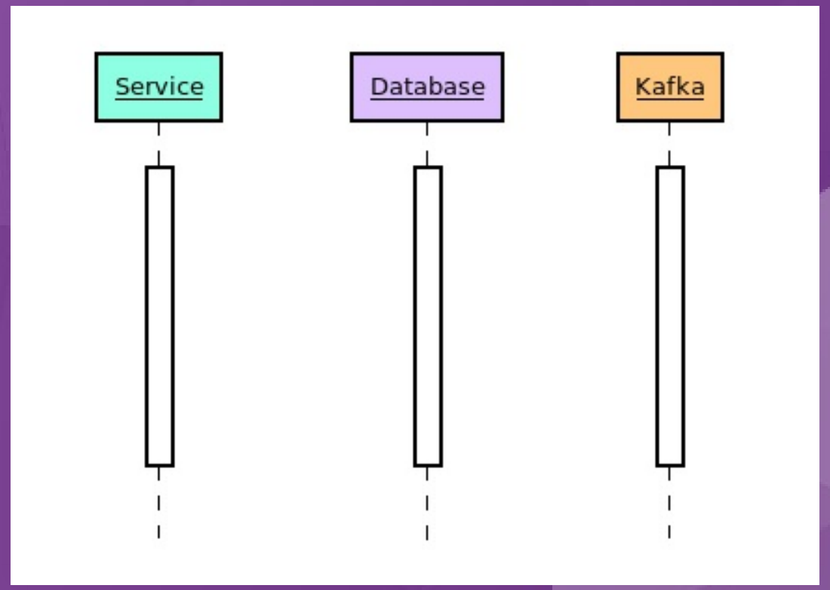
AN EASIER WAY?

- We can't solve the 2 generals problem
- What if come up with a different plan of attack?

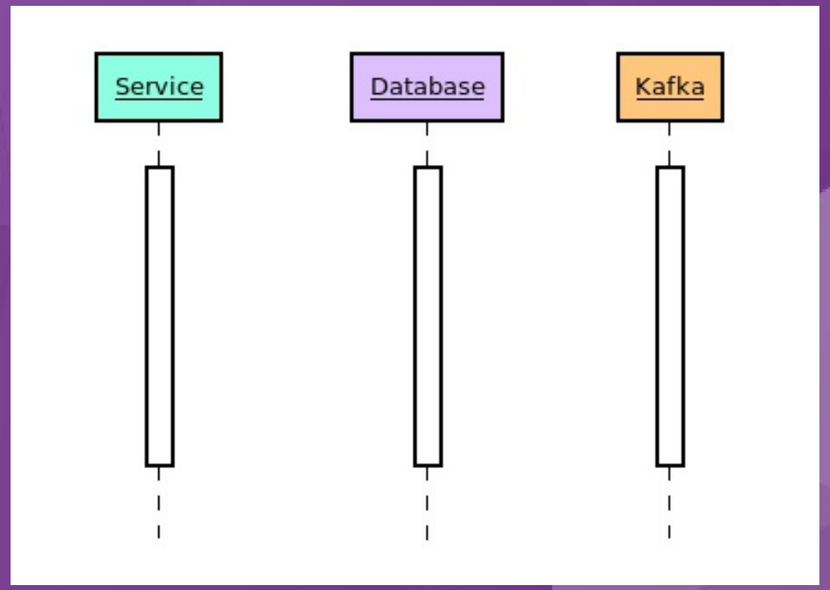
EVENT SOURCING

- Don't store the current state
- Store the events that occurred
- Compute the state from the events

EVENT SOURCING



EVENT SOURCING



EVENT SOURCING

- Advantages:
  - Fact centric approach
  - Self healing when problems occur
  - Built in audit log
  - Compute the state of the system at any time

EVENT SOURCING

- Details:
  - How is consistency addressed?
  - Is replaying events expensive?
  - How to query data?

SUMMARY

- CRUD is not an ideal fit for distributed systems
- Event sourcing is an ideal fit for distributed systems
- Event sourcing brings a host of other advantages

NEXT STEPS

https://www.lagomframework.com

https://github.com/jroper/crud-to-es