Doing things the hard way

@ChrisSinjo
Hi
An SRE
“Obvious” mistakes and why we make them
Conference talks favour certain structures.
Conference talks favour self-contained narratives.
“How we fixed the unfixable”

–Fixing Things Ltd
“How we scaled our system 100x”

–ScaleCorp
These are great stories to tell!
But there's more...
Mistakes
The ones that were “obvious”
The mistakes you never thought you'd make
Except you did
And I hope I can convince you
This is normal
The reasons are often reasonable
Talking openly is important
Context & biases
Size:
25 → 215 total
(8 → 60 eng)
GO CARDLESS
Hindsight
Structure:
3 examples
foreach(example):
foreach(example):

Define it
foreach(example):

Define it
What it looks like
foreach(example):

Define it
What it looks like
Problems caused
foreach(example):

Define it
What it looks like
Problems caused
Fixes
Common themes
Common themes

Q&A
So let’s get to it
Failure mode 1

Early Infra/Product Divide
You’re a young company
You’ve built a product
Your userbase is growing
You’ve also built this other thing
Your product needs it to work
It caught you by surprise.
You have an infra!
It takes up dev time
You weren’t ready for this
“Can’t someone make this go away?”
“We need to hire a DevOps”
It sounds silly
But it literally happens
“We have all this rubbish that’s distracting our devs.”

–The least appealing job description ever
The phrasing was clunky
But the framing is common
Convenience
An understandable lever to pull
But...
Problems
Now you have organisational problems
Disconnect devs from production
A new bottleneck
Too much infra

Too soon
Solutions
Assuming you can’t un-split
Make infra contributions easy
Make it obvious
what needs changing
Make experimentation easy
Set aside time to coach
Breaking my own rules
Some up-front advice
First infra hire: dev background
Embed them in the existing team
Don’t give them sole ownership of the pager
Failure mode 2

Distracted by hard problems
We hear it so often
“Join us and solve hard problems”

— Every job ad
We assume hard problems are most important.
They frequently aren't
Outcome: we neglect the basics
When I say "basics"...
Observability
Metrics

Monitoring

(Structured)

Events/Logs
Metrics

Monitoring

(Structured)

Events/Logs

→

SLOs
Metrics

Monitoring

(Structured)

Events/Logs

Goals
Metrics

Monitoring

(Structured)

Events/Logs

Uptime
Metrics

Monitoring

(Structured)

Events/Logs

Error rate
Metrics
Monitoring (Structured) Events/Logs ➔ Latency
Easy to defer
It feels mundane
It feels mundane
- As a project
“So how does this improve the service?”
“So how does this improve the service?”

“We can measure it better.”
“So how does this improve the service?”

“We can measure it better.”

“How does that improve it?”
“So how does this improve the service?”

“We can measure it better.”

“How does that improve it?”
- Faster debugging
- Faster debugging
- Shorter outages
- Faster debugging
- Shorter outages
- Better project choice
It feels mundane
- As a project
It feels mundane

- As a project
- As ongoing work
Observability is ongoing work
Problems
Previously...

Zero-downtime Postgres upgrades

Restarting databases without the apps noticing

@ChrisSinjo

https://www.youtube.com/watch?v=SAkNBiZzEX8
Was 10-15s of downtime okay?
Back to basics?
- Faster debugging
- Shorter outages
- Better project choice
- Slower debugging
- Longer outages
- Worse project choice
Lack of confidence
Post-mortem meta-analysis
“It wasn’t clear where the problem was.”

–Post-mortems 1, 2, 3
“We couldn’t break the errors down by user.”

–Post-mortems 2, 3, 4
“It was a false alarm. Again.”

–Post-mortems 3, 4, 5
You can do better at the basics
A cultural shift
Definition of done
Done when it’s shipped

↓

Done when it’s measured
A huge shift
Cultural change takes time
Start somewhere
There are other basics
- Post-mortem analysis
- Post-mortem analysis
- Tracking toil
- Post-mortem analysis
- Tracking toil
- Tracking pages per shift
Failure mode 3

The everything project
Story-based
Kinda painful to tell
The most immediate impact
You have an infra!
You’re not happy with it :(

You’re not happy with it :(
It evolved haphazardly
You know where the problems are
You want to fix them
Reshaping the core
Previously...

A million containers isn’t cool

You know what’s cool? A hundred containers.

@ChrisSinjo

https://www.usenix.org/conference/srecon17americas/program/presentation/sinjakli
The precursor
Goal: Better deployment
Containers
Containers

Orchestrator (Mesos)
Containers

Orchestrator (Mesos)

Load balancing
Containers

Orchestrator (Mesos)

Load balancing

Staging-per-developer
Containers
Orchestrator (Mesos)
Load balancing
Staging-per-developer
Self-serve developer UI
We were working on an Everything Project
Problems
Everything is seen in terms of the New World.
So nothing happens back in The Old World
It feels efficient
But it's not
Loss of impact
Loss of confidence
Loss of

team morale
Solutions
STOP
ARRÊT
Look for the smallest version
Look for the valuable part
For us:
deployment
Containers
Orchestrator (Mesos)
Load balancing
Staging-per-developer
Self-serve developer UI
Containers
From idea to reality: containers in production at GoCardless

As developers, we work on features that our users interact with every day. When you’re working on the infrastructure that underpins those features, success is silent to the outside world, and failure looks like this:

https://gocardless.com/blog/from-idea-to-reality-containers-in-production-at-gocardless/
Efficiency cannot come at the cost of everything else.
No stopping the world
Long-term goals
Long-term goals
Short-term reality
Things You Should Never Do, Part I

Netscape 6.0 is finally going into its first public beta. There never was a version 5.0. The last major release, version 4.0, was released almost three years ago. Three years is an awfully long time in the Internet world. During this time, Netscape sat by, helplessly, as their market share plummeted.

It’s a bit smarmy of me to criticize them for waiting so long between releases. They didn’t do it on purpose, now, did they?

Well, yes. They did. They did it by making the single worst strategic mistake that any software company can make:

They decided to rewrite the code from scratch.
Mistakes

Early Infra/Product Divide

Distracted by hard problems

The everything project
I’ve presented 3 “obvious” mistakes
Not first
Not last
Each has an internal logic
Conference talks favour self-contained narratives.
Even when talking about mistakes
Technical mistakes are self-contained.
Us vs Them
And I hope I have convinced you.
You won’t avoid every mistake
We certainly didn’t
It's never perfect
It’s perfectly fine to correct course.
Thank you 🍀❤️
Schemes supported by GoCardless

GoCardless provides a single point of access to Europe’s largest Direct Debit schemes. Learn more about each scheme below, or get in touch to speak to our payments expert.

**UK**
Bacs

**Eurozone**
SEPA

**Australia**
BECS

Currency: GBP

Automated payments are at the very centre of the UK’s financial system, providing an essential service for both consumers and organisations. Bacs is the company which runs Direct Debit in the UK. 74% of all recurring payments are made by Direct Debit.

Currency: EUR

SEPA Direct Debit is a Europe-wide Direct Debit system that allows merchants to collect Euro-denominated payments from accounts in the 19 SEPA countries and associated territories.

Key markets:
- Germany
- Austria
- France
- Spain
- and The Netherlands

Currency: AUD

The Bulk Electronic Clearing System (BECS) is managed and regulated by BECS. Direct Debit is a popular payment method in Australia, making up around 1/3 of all non-cash payments.

https://gocardless.com/schemes
We’re hiring
Image credits

• XOXO Festival Day 2 - CC-BY - https://www.flickr.com/photos/textfiles/15237123601/

• USS Barry conducts a practice pipe-patching drills during MultiSail 17 - CC-BY - https://www.flickr.com/photos/usnavy/32480491984/

• Train lever - CC-BY - https://www.flickr.com/photos/darkbuffet/2309897403/

• Calendar - CC-BY - https://www.flickr.com/photos/dafnecholet/5374200948/
Image credits


• Stop sign - CC-BY - https://www.flickr.com/photos/wolfsavard/4812833180/

• Rope - CC-BY - https://www.flickr.com/photos/49140926@N07/6798304070/
Questions?

@ChrisSinjo
@GoCardlessEng