Sam Vanhoutte

Azure MVP
< Belgium >

7 things I learned by coaching and investing in cloud startups

SamVanhoutte

I ♥ cloudburst

Cloudburst 2014
IoT on Microsoft platform

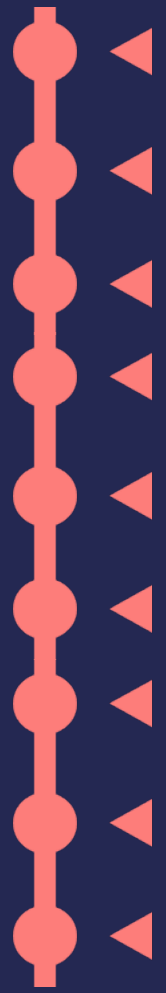
Cloudburst 2013
Cloud integration patterns
with Windows Azure

Cloudburst 2017
Living on the Edge
Azure IoT Edge

readme.md

- Belgium
- Microsoft Azure MVP (9 years)
- Advisor and investor in cloud-startups
- Former CTO at Codit (17 years)
- Focus on Data & A.I., IoT and Serverless
- Passion for cycling & traveling

How did I end up here?



2001: Started at a dev shop – learned BizTalk

2002: Joined an industry company

2004: Joined "1 person company" Codit

2007: Invested and joined as shareholder at Codit

2013: Built iPaaS at Codit, on Azure

2017: Opened 8th country of Codit

2017: Built Nebulus IoT Gateway, on Azure IoT

2018: Codit acquired by Proximus, stock listed company

2022: Left Codit, and started working with startups

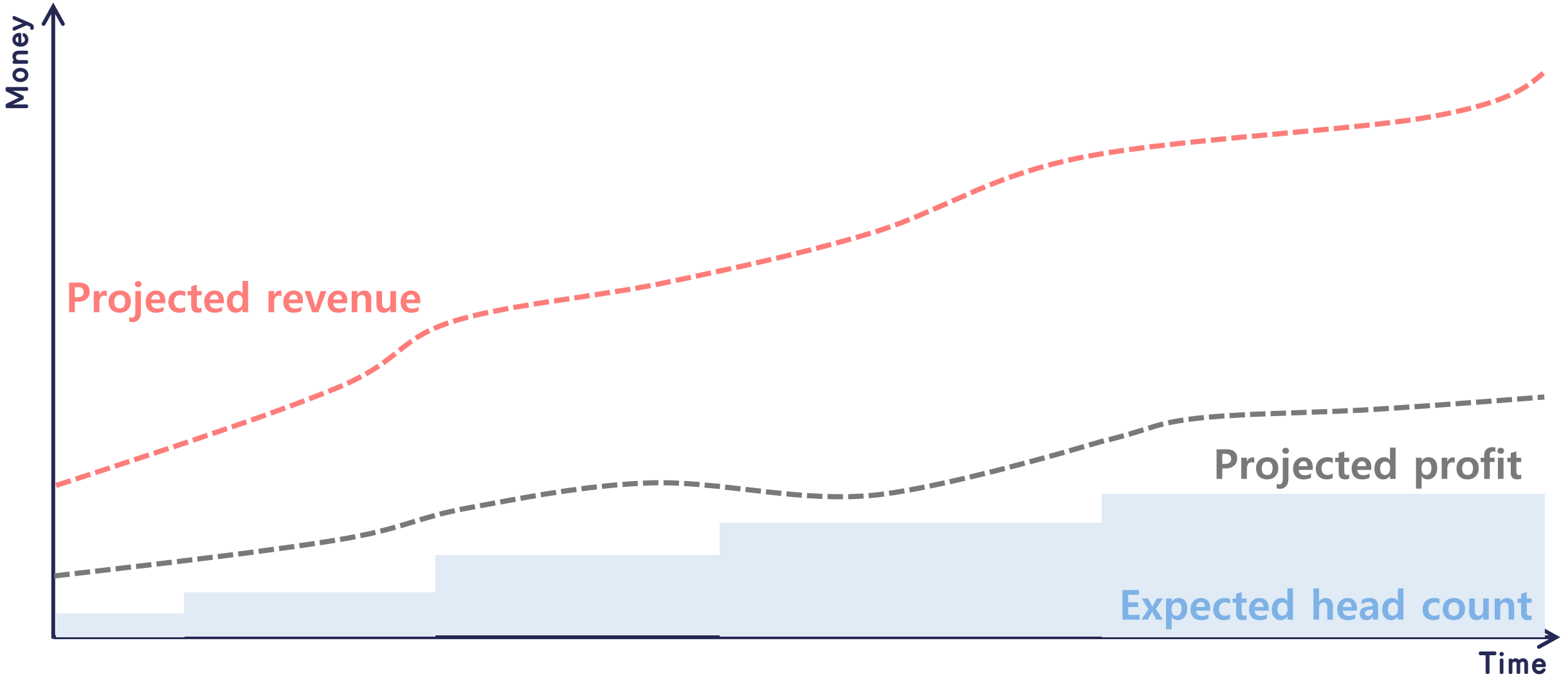
Working for a startup

And how it's different from the enterprise / consultancy world

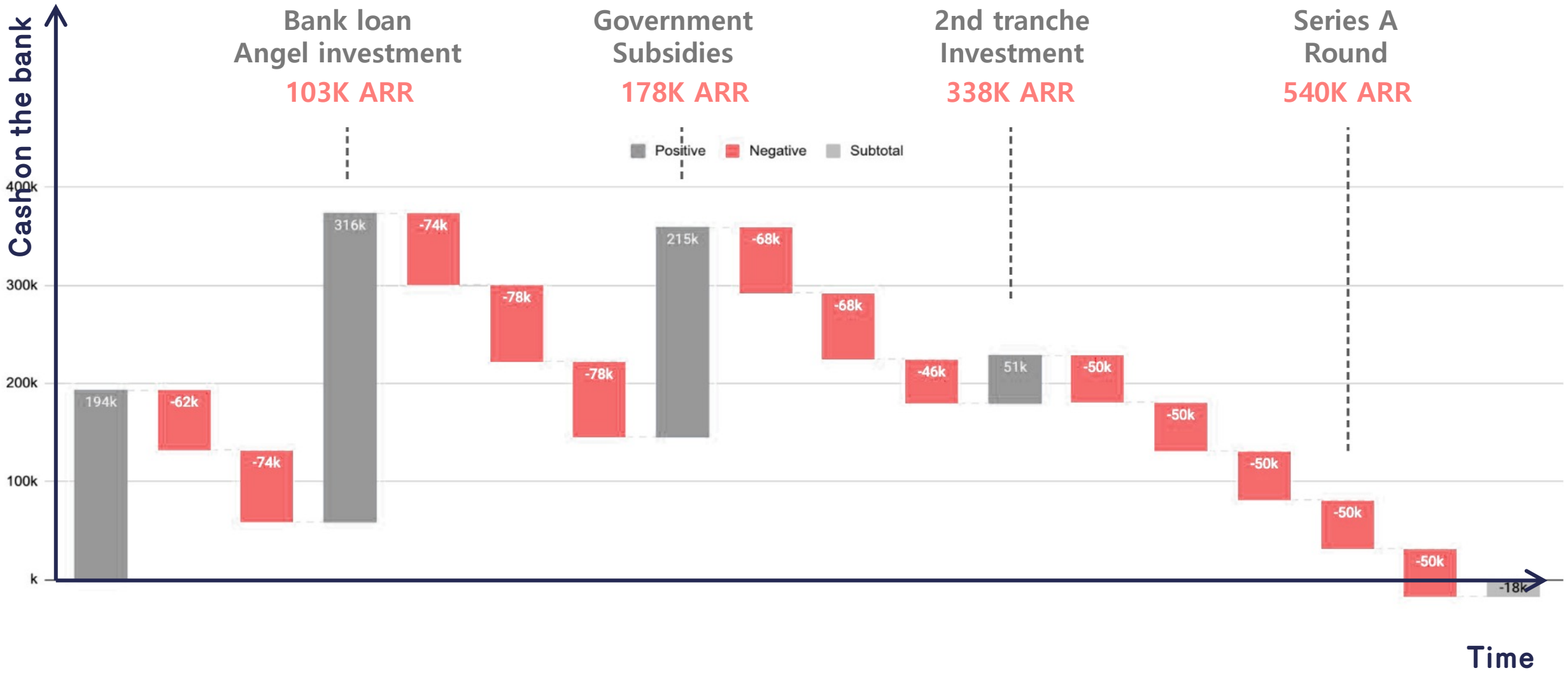
Advantages of working for different companies

Tech firm	Startup
Access to scale & resources	Big gear in a small machine
Learn organizational mechanics	Passion triggers motivation
Job mobility inside the corp	Learn from entrepreneurs
Growth and mentorship	Move fast. Get it out, then right
Safety	Potentially large financial upside
Strict & clear roles	Jacks of all trades

Management charts at tech firms



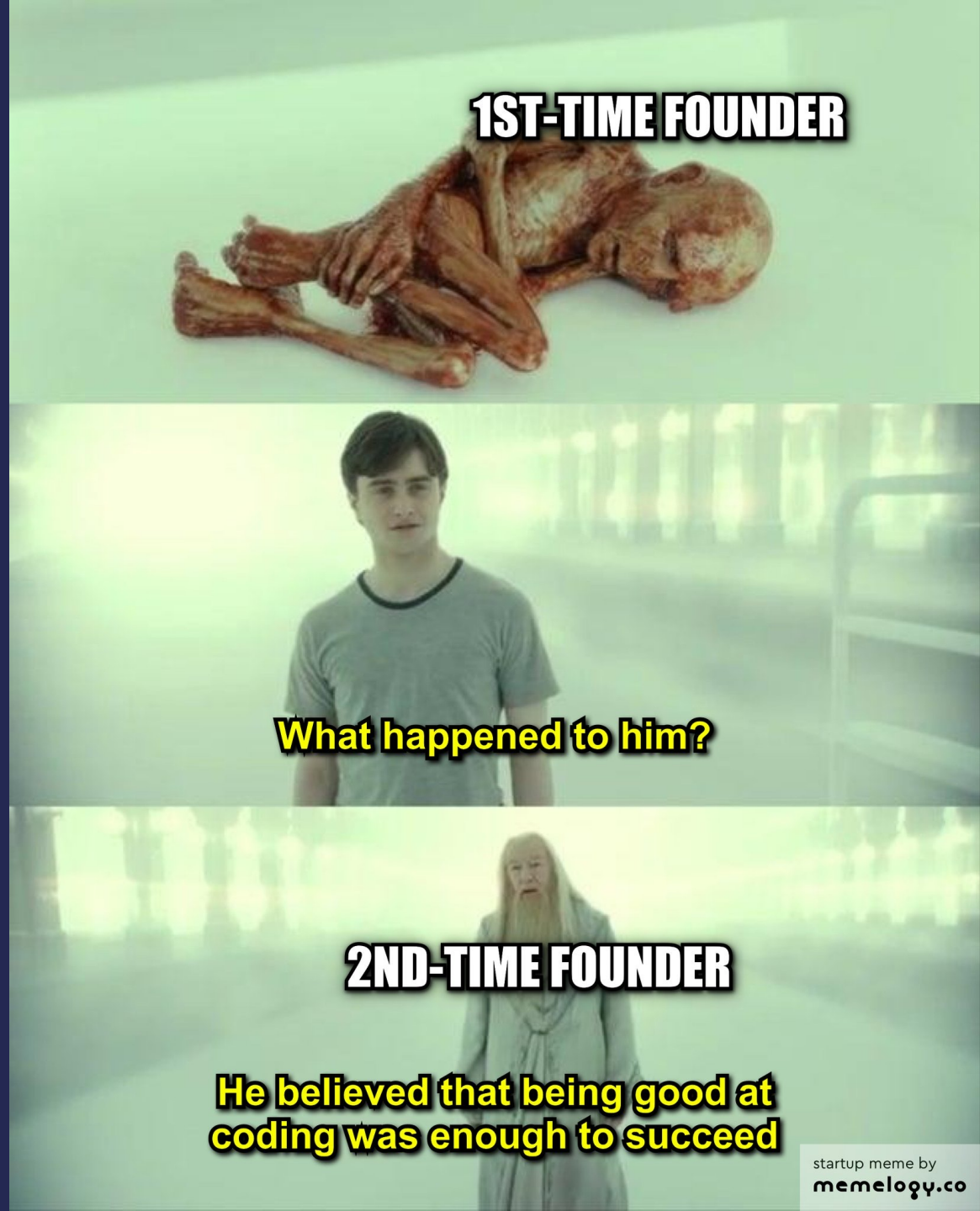
Management charts at young startups



Top reasons why start-ups fail

credits: https://www.cbinsights.com/research/startup-failure-reasons-top/

We're all naïve at the beginning



credits: twitter.com/dagorenouf

We don’t understand what’s ahead of us

« Building a product is the easy part. Hard work only starts after launch. »

credits: twitter.com/dagorenouf

And we get confused by all the advice we get

credits: twitter.com/dagorenouf

So we look for shortcuts to reach our goals

credits: twitter.com/dagorenouf

But as we start experiencing failure, and get our hearts broken

Our obsession keeps us going

credits: twitter.com/dagorenouf

memelogy.co

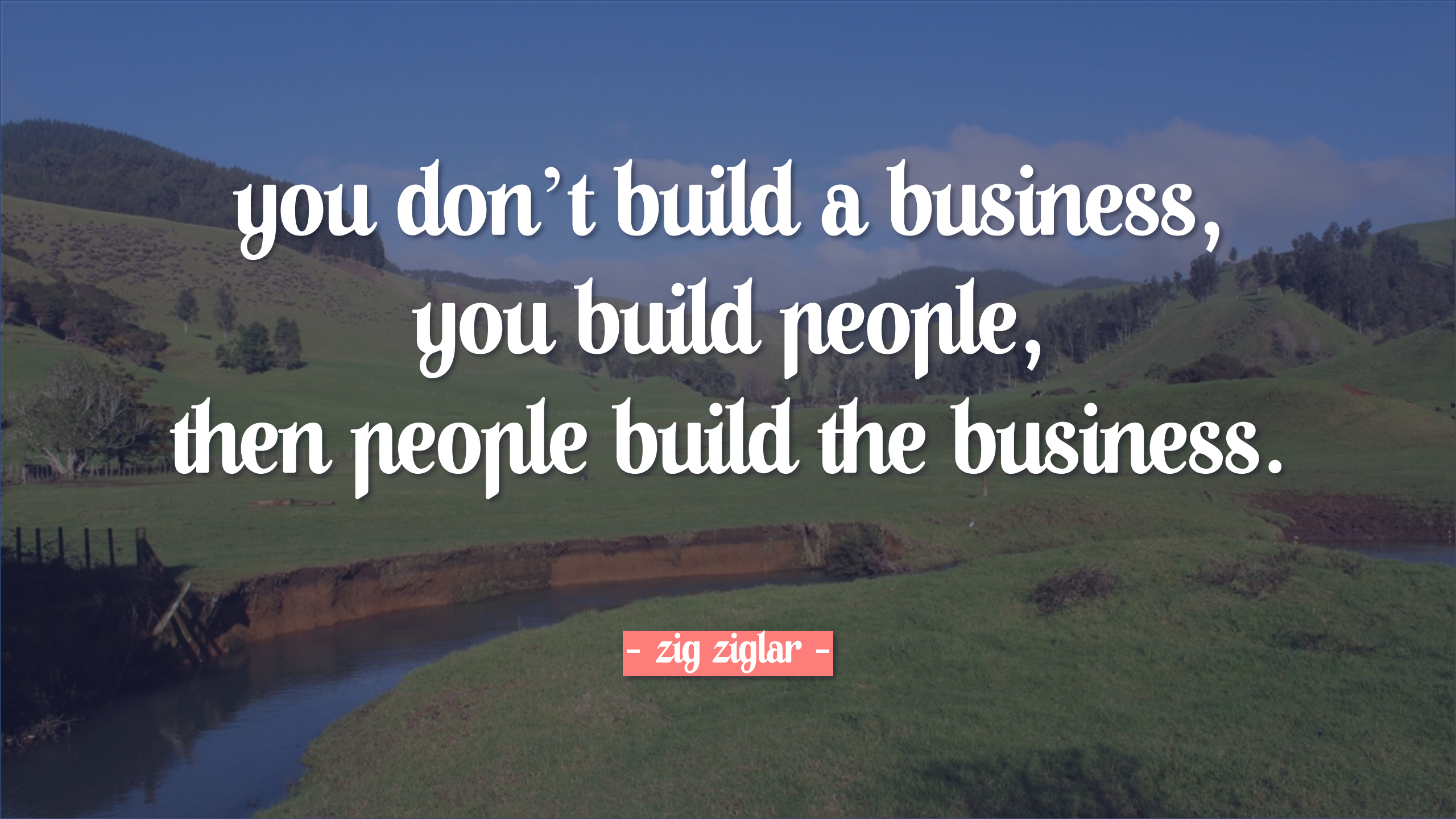
Until one day we get our first win

credits: twitter.com/dagorenouf

And celebrate with the friends we made along the way

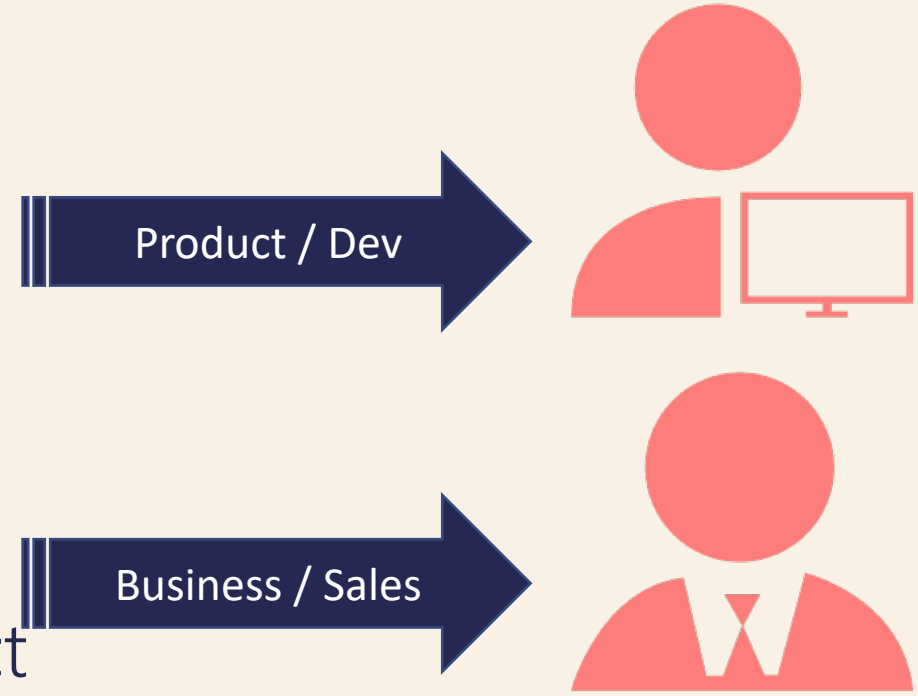
credits: twitter.com/dagorenouf

The team is more important than the tech


you don't build a business,
you build people,
then people build the business.
- zig ziglar -

Required roles

- Chief Executive Officer: the Dreamer
- Chief Product Officer: the Visionary
- Chief Technology Officer: the doer
- Chief Sales Officer: the hustler
- Chief Marketing Officer: the architect



redits https://brainhub.eu/library/5-crucial-roles-in-every-tech-startup

The dev team

- Get full-time dedicated developers
  - Experienced lead dev
  - Ambitious junior people,
    willing to make an impact and eager to learn
  - Make them co-own the company (shares/stock options…)
- Okay to hire freelancers for specific niche experience
  - Pen-tests , app dev , UX design…

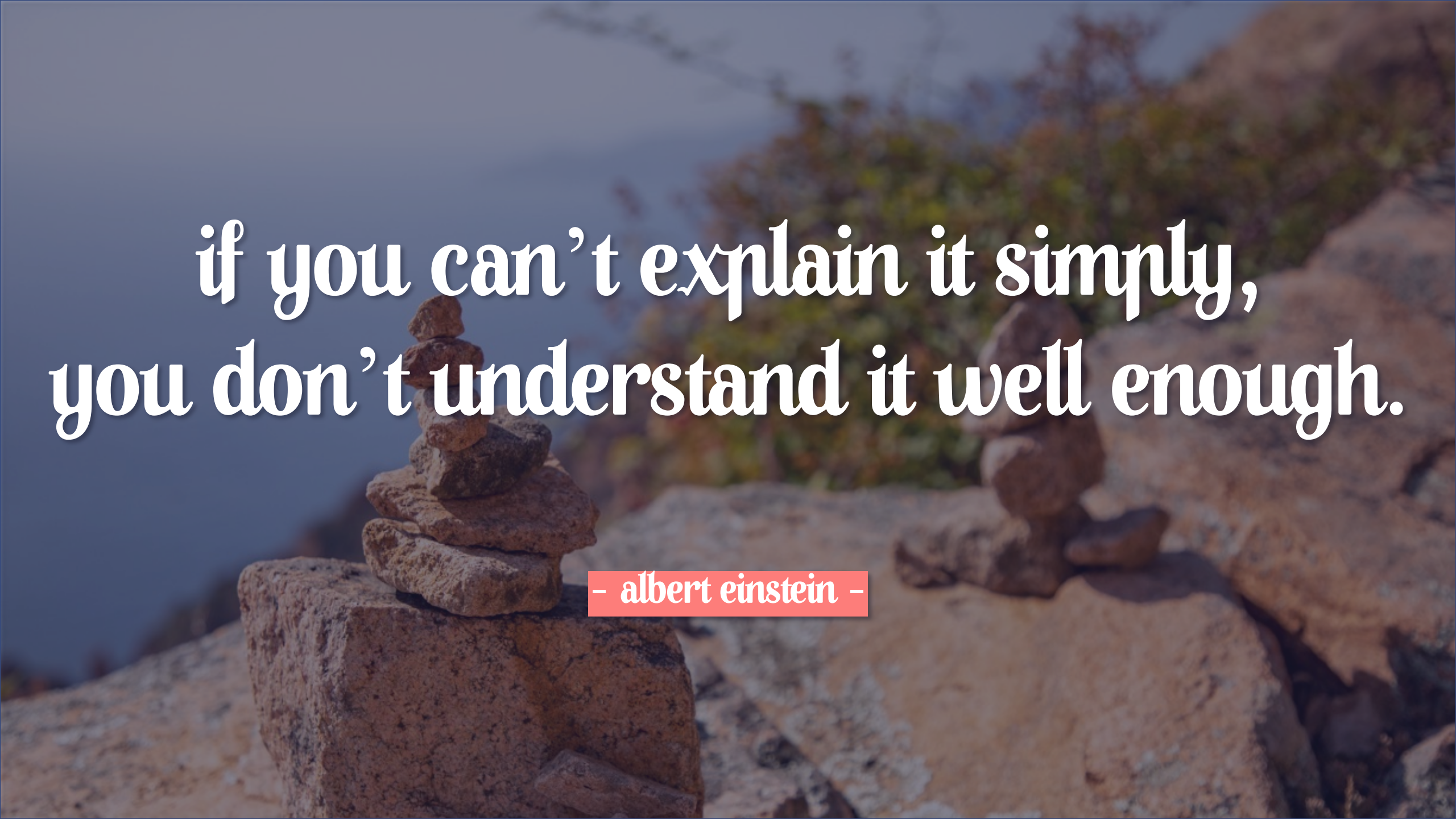
Appreciate advisory

- Every startup : Board of advisors / Board of directors
  - Focus on sales, market fit, strategy, company valuation
- My recommendation : Set up a technical board of advisors too
  - Community / tech experts
  - Learn from others

No sales,
without marketing

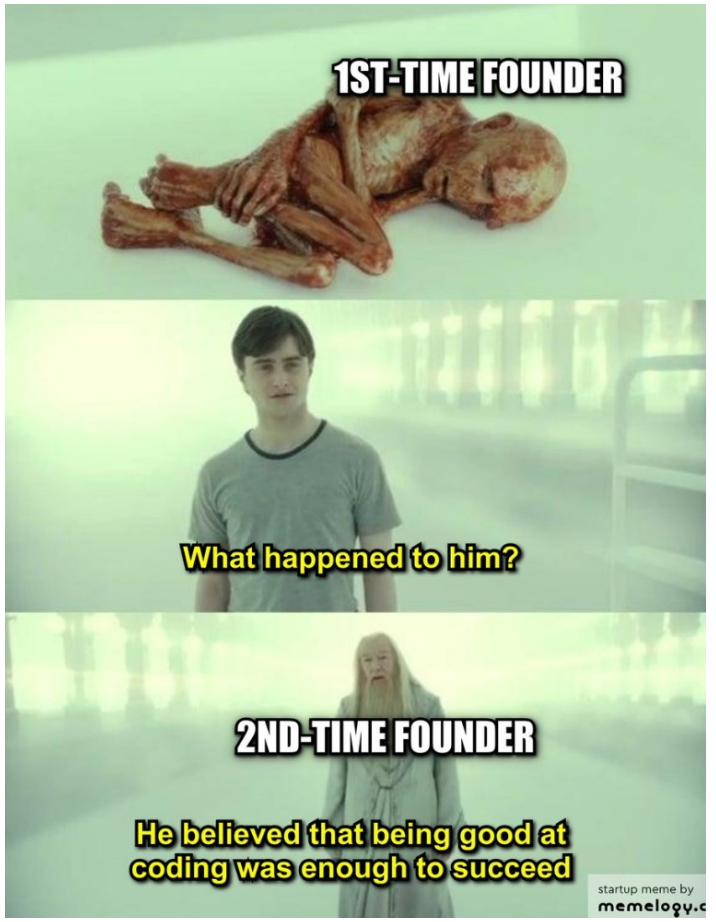
The most successful company is the company with the best sales,

not always with the best product.


if you can't explain it simply,
you don't understand it well enough.
- albert einstein -

Marketing: keep iterating your story

- Too many people are stuck in their dev environment
- Get out and get customers, the sooner, the better



Product-market fit

A bottle of water can be $0.50 at a supermarket.

$2 at the gym. $3 at the movies

And $5,000 when I'm selling it to someone who's on fire

- Same water. Only thing that changed its value was the need and the place. So the next time you feel your worth is nothing, maybe you're solving the right need.

Valuation of cloud startups

- **ARR:** Annual Recurring Revenue
  - Subscriptions & licenses
  - User base

< goals >

Increase

Strategic Revenue

Don't run

Out of Money

Growing sales

- More new customers
  - Sales cycles can be long, initially
  - Pick your focus customers (industry specific, area specific…)
- Grow existing customers

  - Search for premium features
  - Tiered licenses: value based pricing

Leverage technical debt, but repay in time

The cost of refactoring


make it work,
make it right,
make it fast
- kent beck -

The concept of technical debt

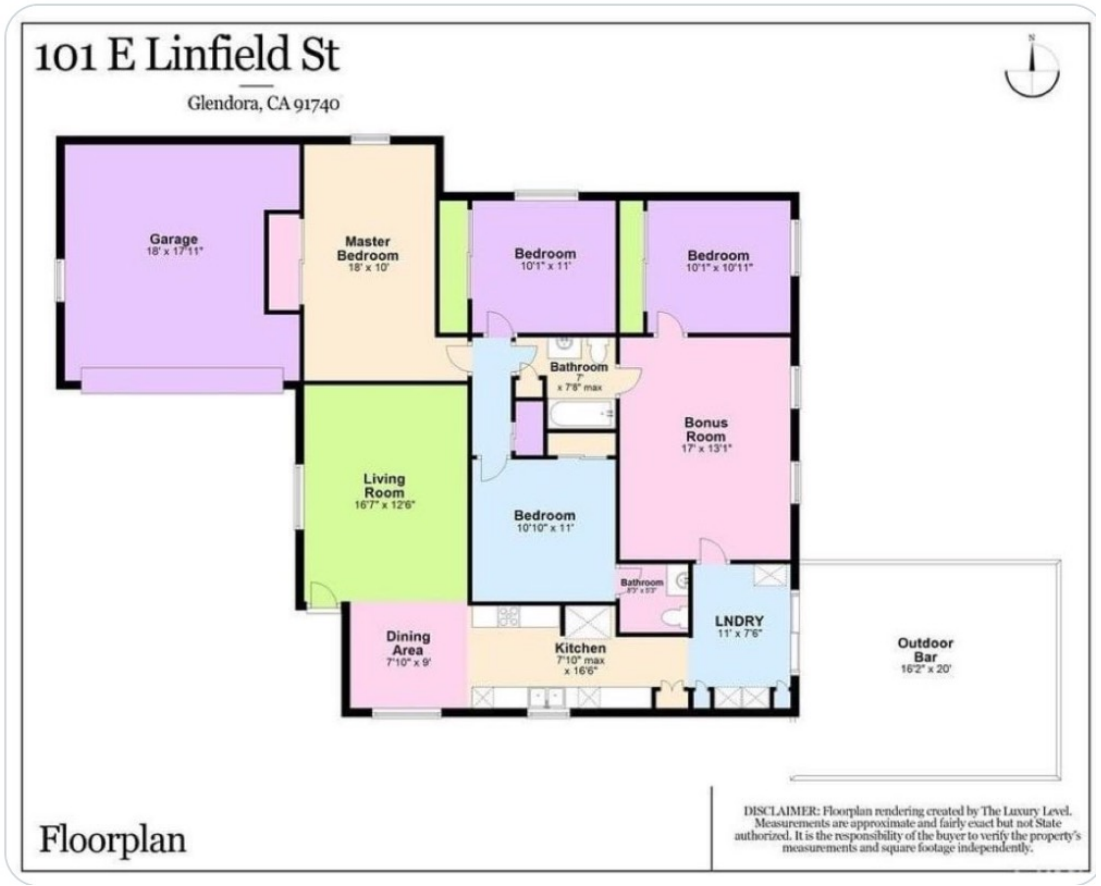
- Much like financial debt
  - Quickly have something now (purchase) than to save funds to purchase all at once
- Prioritize speed over well-designed solutions.
  - Quick fixes & patches
- Dev teams have to make decisions, but make debt responsibly
  - Debt has to be repaid
- If not repaid
  - Will slow down the process of delivery
  - Increase operational costs and maintenance
- Budget for refactoring & convince stakeholders

The concept of technical debt

- Much like financial debt
  - Quickly have something now (purchase) than to save f
- Prioritize speed over well-designed solutions.
  - Quick fixes & patches
- Dev teams have to make decisions, but make de
  - Debt has to be repaid
- If not repaid
  - Will slow down the process of delivery
  - Increase operational costs and maintenance
- Budget for refactoring & convince stakeholders

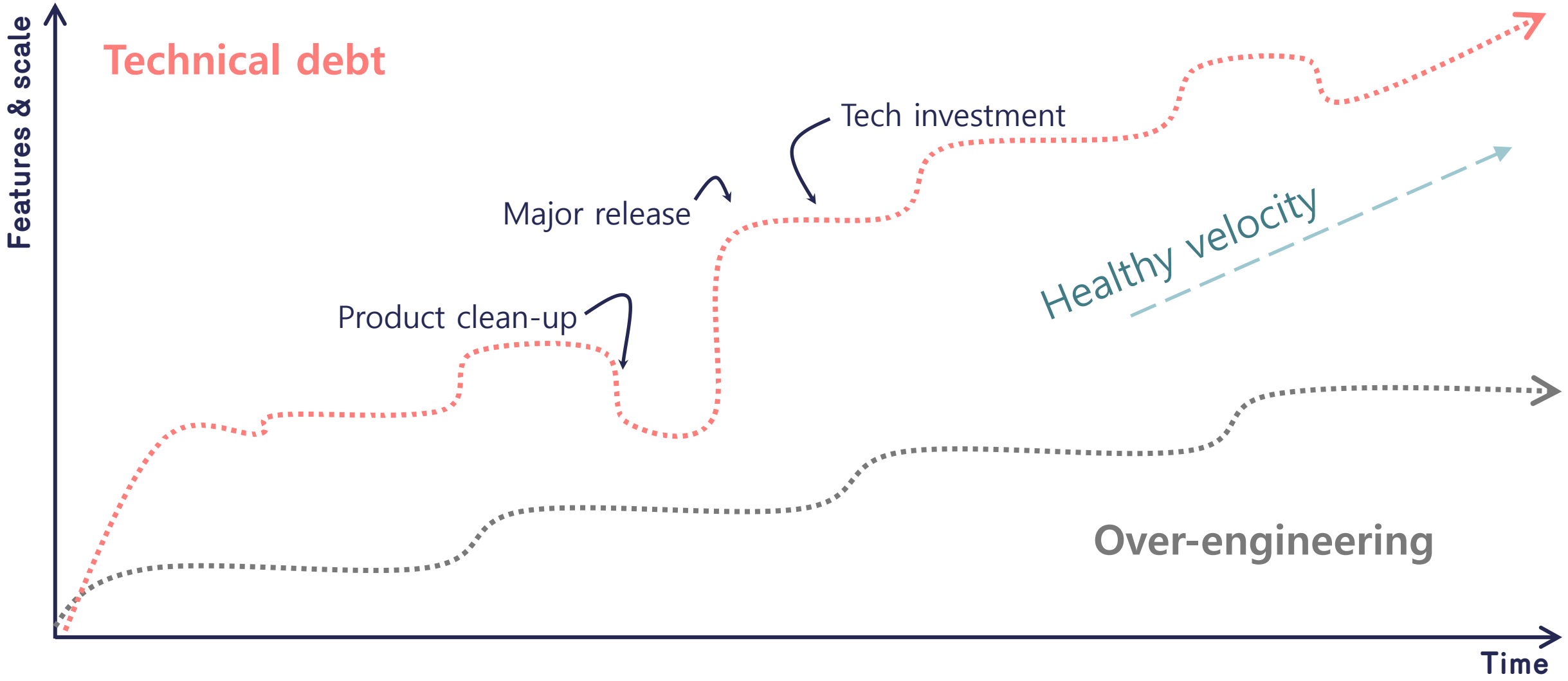
DM of Engineering @dmofengineering · Sep 17

Next time a CEO asks me to explain technical debt, this is going on the slide...



credits: twitter.com/dmofengineering

The healthy use of technical debt



Yacht problems

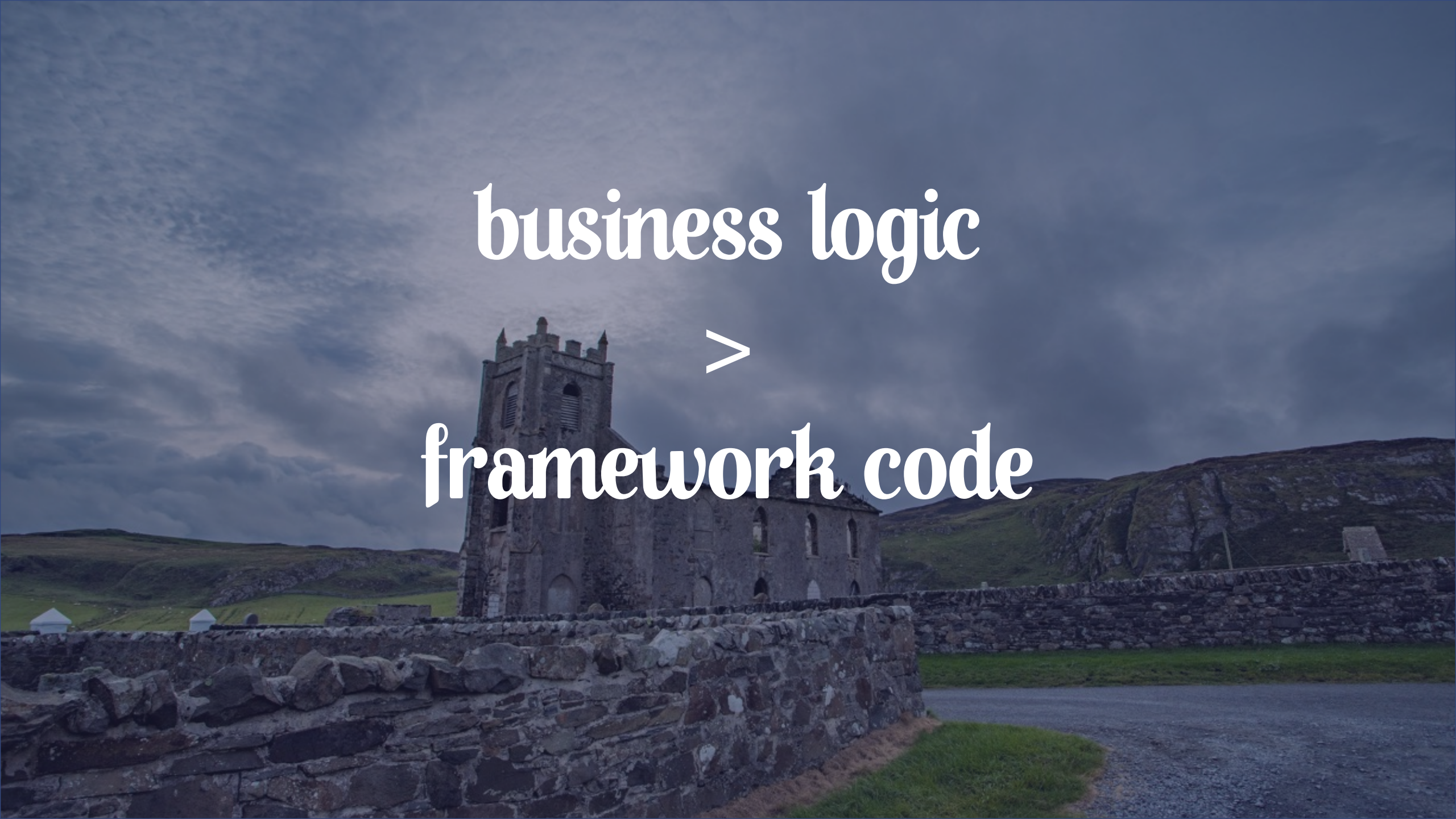
- Problems you're probably worrying about too early. As in,

  "if we run into that problem, we can solve it from the deck of our yacht, because having the problem means this product is already wildly successful."

- a.k.a. Maserati problems, or boat-naming

  "Which color would we pick for our Maserati, when we are successful?"

credits: twitter.com/lauraklein


business logic
>
framework code

Business logic > Framework code

- Every hour you save by not having to write ‘framework code’, can be invested in making a difference for your product.
- Leverage (and support!) open source
  - It is a good thing if you use something that is “not invented here”
- Cloud brings a lot of value out of the box
  - Cost of “build your own” vs Cloud services

Automate everything!

The cost is worth the value


the most powerful tool we have
as developers is automation
- scott hanselman -

Automated deployment

- Infrastructure as code – DevOps – CI/CD pipelines
- By automating and integrating all deployments
  - You gain trust in your releases
  - You link your code / project mgmt and tests to your deployments
  - You never skip testing
- It takes time to get this right,

  but it is crucial to convince your co-investors of the value and need

Advantages of CI/CD approach

- Short release cycles
- Smaller code changes
- Fault isolations
- Smaller backlog
- Increased quality, thus customer satisfaction
- Increased accountability & transparency of the team

Measure & monitor all the things


don't find customers for your product,
find the product for your customers
- seth godin -

Leverage telemetry

- Enrich your telemetry (don't just treat it as trace lines in a log file)
  - Add context & correlation in distributed applications
  - Leverage tracing/correlation ids, in your (API) responses
- Enable search for your application context data

Enable alerts & integrate those events

- Leverage your telemetry & alerting system to generate alert events
  - Handle these as events to integrate as well
  - Integrate these with your service desk

Customer Service depends on Telemetry

- Focus on customer centricity
- Prioritize appropriately
- Write documentation
- Know before they know!
- Business context of telemetry is key

User behavior analytics

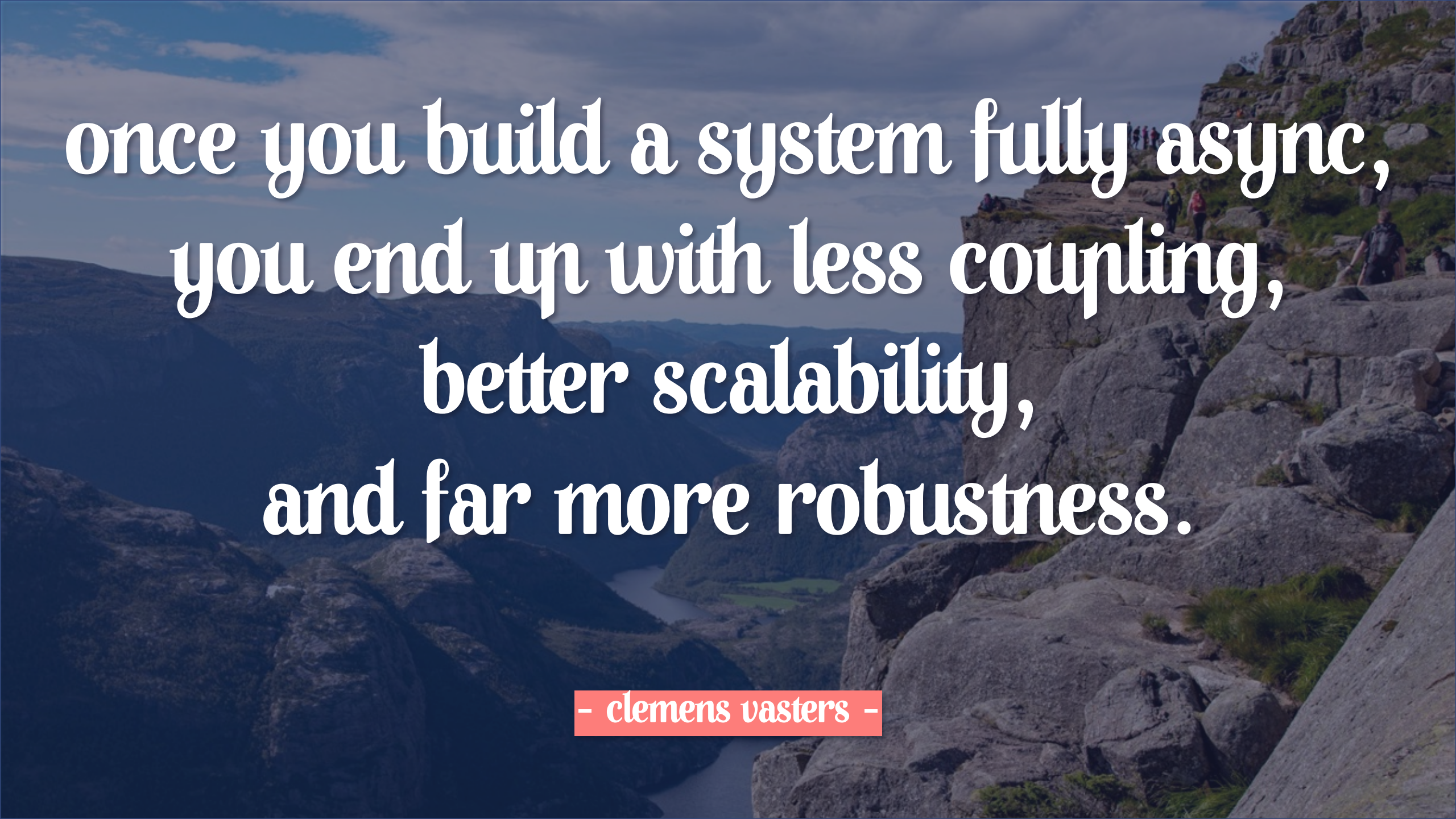
- Know if/how your users are using the product
- Monthly subscriptions / licenses:
  - smaller amounts
  - steady growth, natural ‘pay as you use’
- Yearly subscriptions
  - larger upfront cash
  - tricky moment of renewal (especially in B2B): reconsideration

You only make money after billing

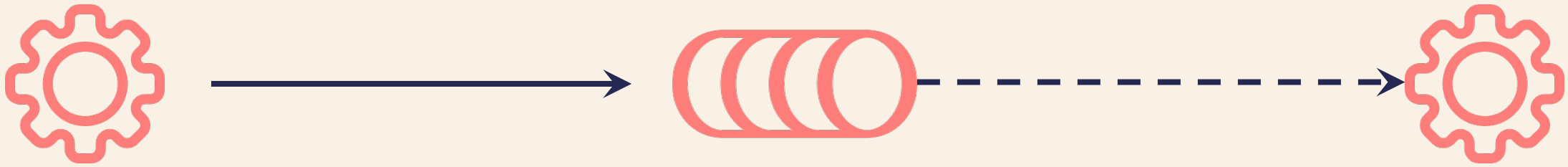
- Pricing models change frequently
  - Know what a change in that model could mean to your business
- Add all raw relevant telemetry to a separate repository
  - Run your billing engine against that
  - Isolation of those metrics

    from the operational/support metrics is key

Loose coupling
is your only option

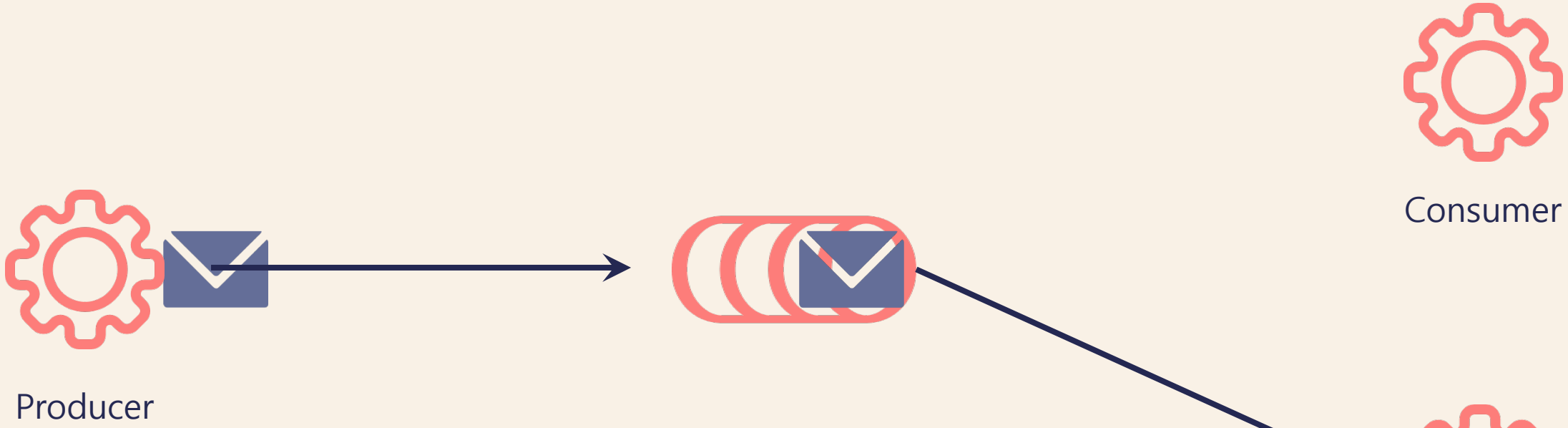

once you build a system fully async,
you end up with less coupling,
better scalability,
and far more robustness.
- clemens vasters -

Why queues are important in software: offline & temporal decoupling



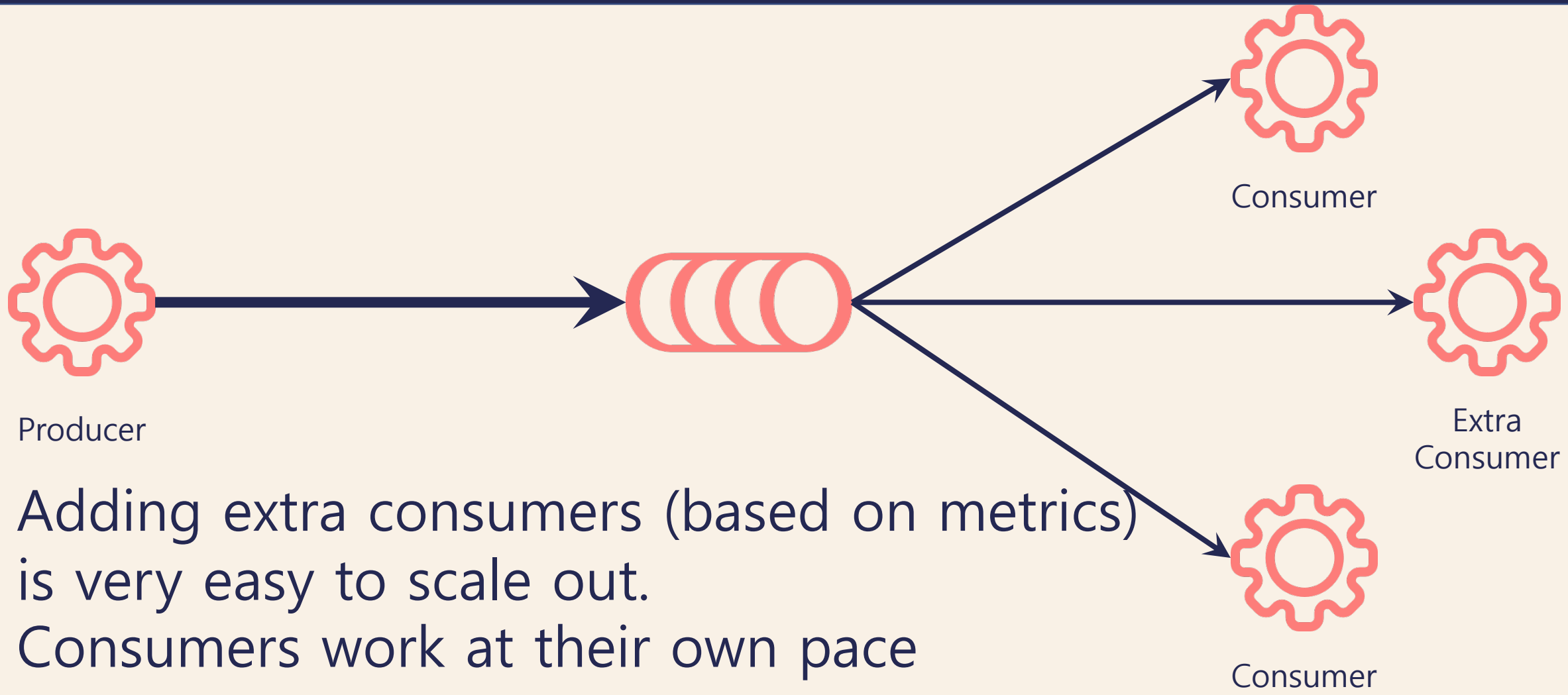
Producers are not blocked,
as messages get buffered,
when consumers are offline

Why queues are important in software: reliability



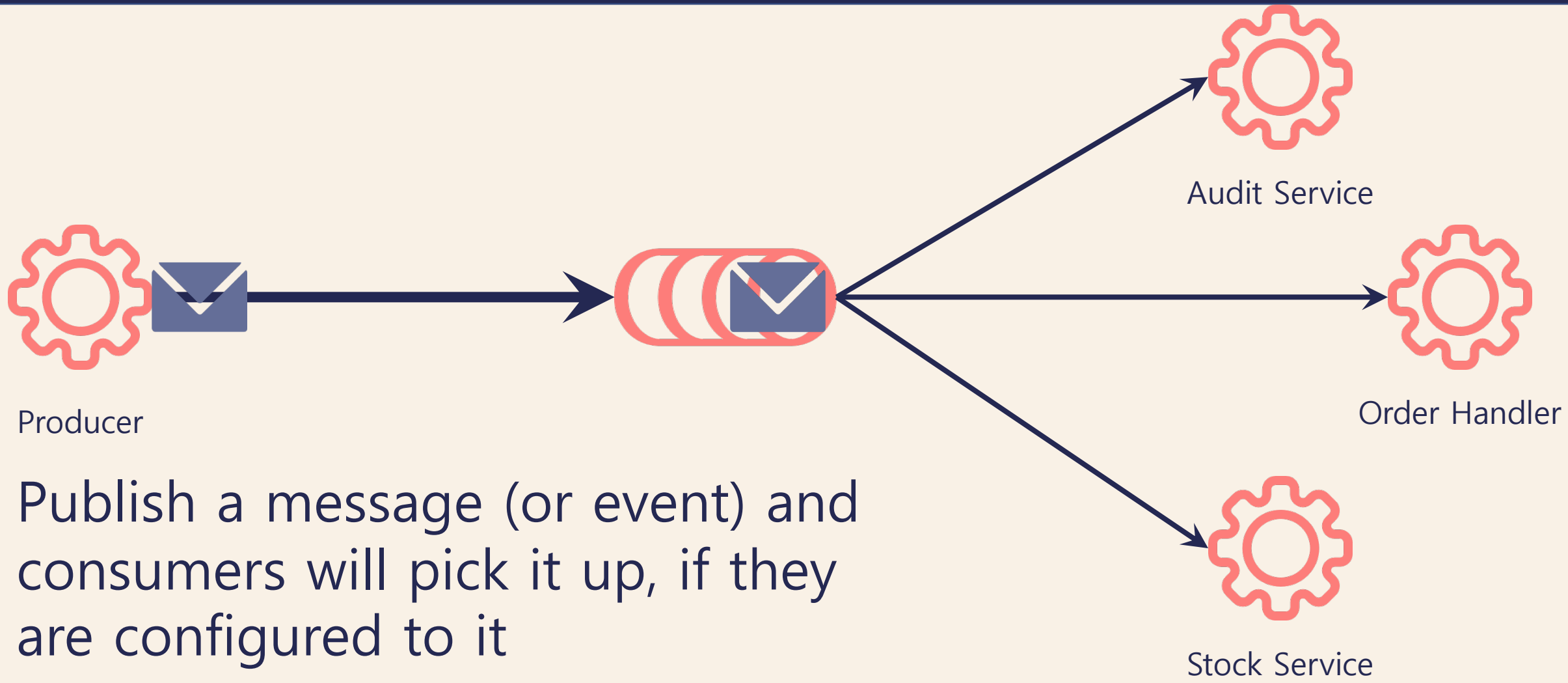
Consumers can fail, so messages become available to other consumers

Why queues are important in software:
load leveling: ready for scalability



Adding extra consumers (based on metrics) is very easy to scale out.
Consumers work at their own pace

Why queues are important in software: publish & subscribe



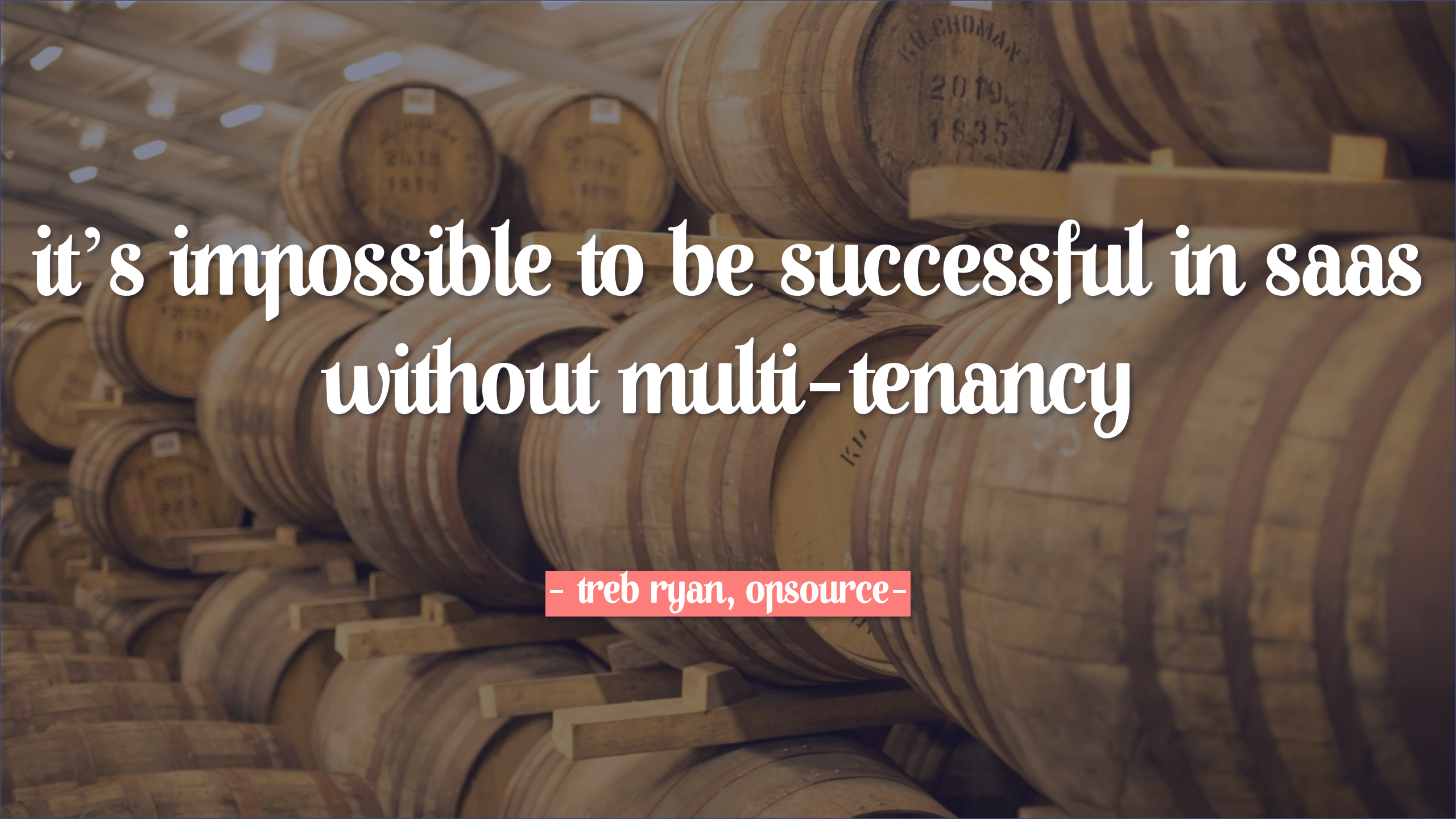
Publish a message (or event) and consumers will pick it up, if they are configured to it

Best practices

- Loose coupling can scale your components independently
- Provide extensibility & integration points with Event publication (such as Azure Event Grid)
- Decouple internal implementation from external callers
  - DNS names – API gateway layer …

Data storage

Multi-tenancy

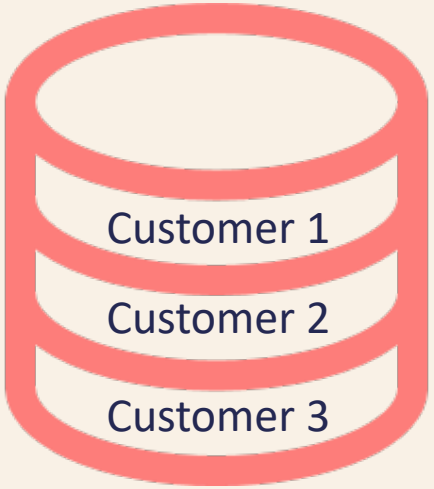


it's impossible to be successful in saas without multi-tenancy

- treb ryan, opsource -

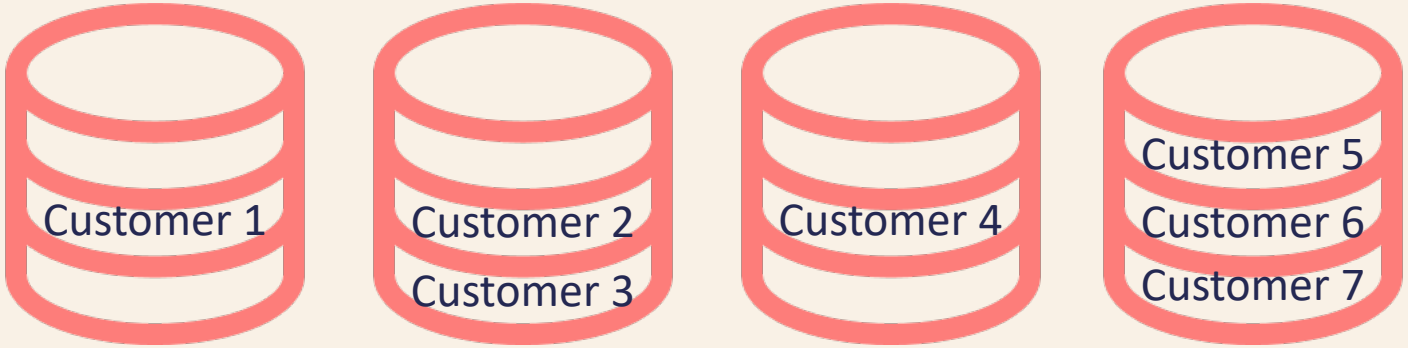
Multi-tenancy strategies

- Each tenant has its own data store
- Perfect isolation
- Can become expensive for smaller customers



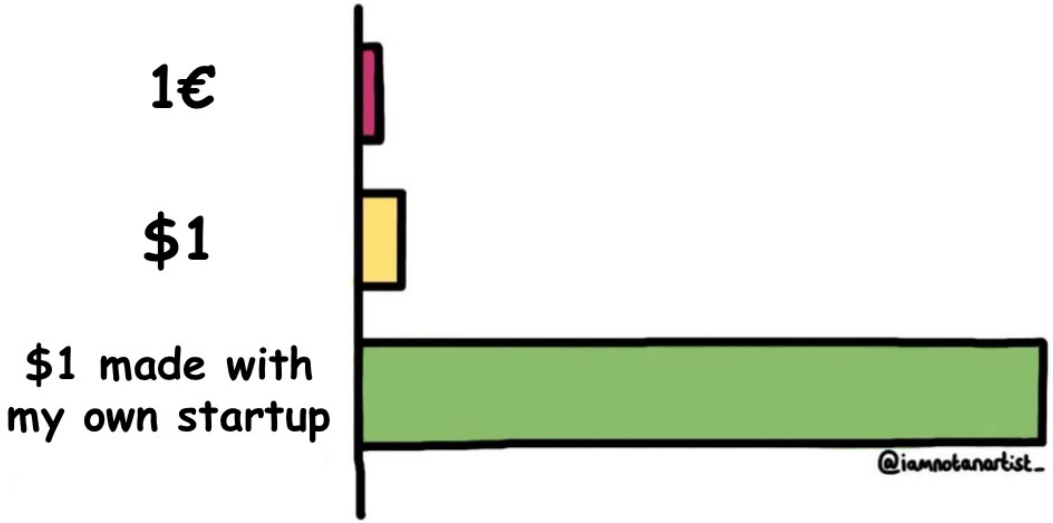
- Multiple tenants in the same store
- Less isolation, requires extra filter in queries
- Less expensive

Prepare for both scenarios



- Most flexibility to be future ready
- Dedicated storage can be premium feature
- Refactoring later can become challenging

Top currencies in the world in 2022



credits: memelogy.co

Thank you
let's connect

twitter.com/SamVanhoutte

github.com/SamVanhoutte

All samples available on:

▷ https://github.com/SamVanhoutte/azure-time-travel