



GOTO Copenhagen 2018
Conference Nov. 19 - 21



Cloud native is about
culture, **not** containers

Holly Cummins
IBM Cloud Garage

by the way ...

the IBM **Cloud** Garage is hiring ...

by the way ...

the IBM **Cloud** Garage is hiring ...

... in Copenhagen :)



what is cloud native?

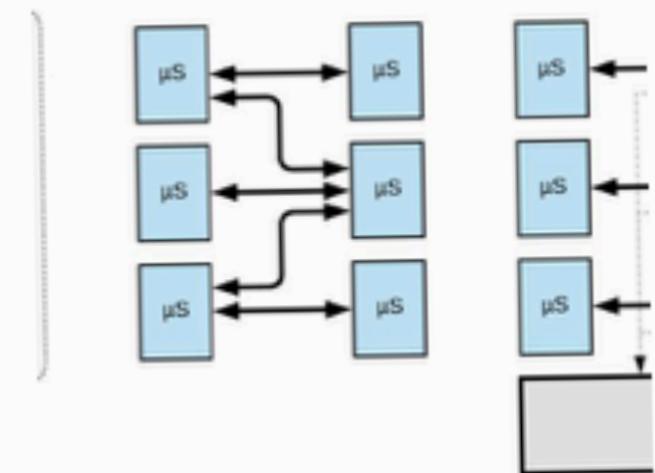


Daniel Bryant
@danielbryantuk

Following

I've gotta hand it to [@bibryam](#), he's got a great way of framing things... :-)

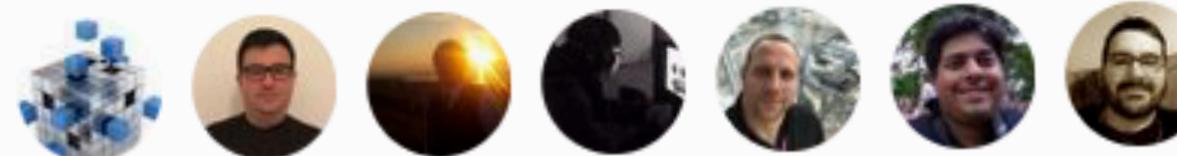
Microservices
(smart endpoints, dumb pipes)



Bilgin Ibryam @bibryam
ESB -> Microservices -> CloudNative

8:42 AM - 7 Aug 2018

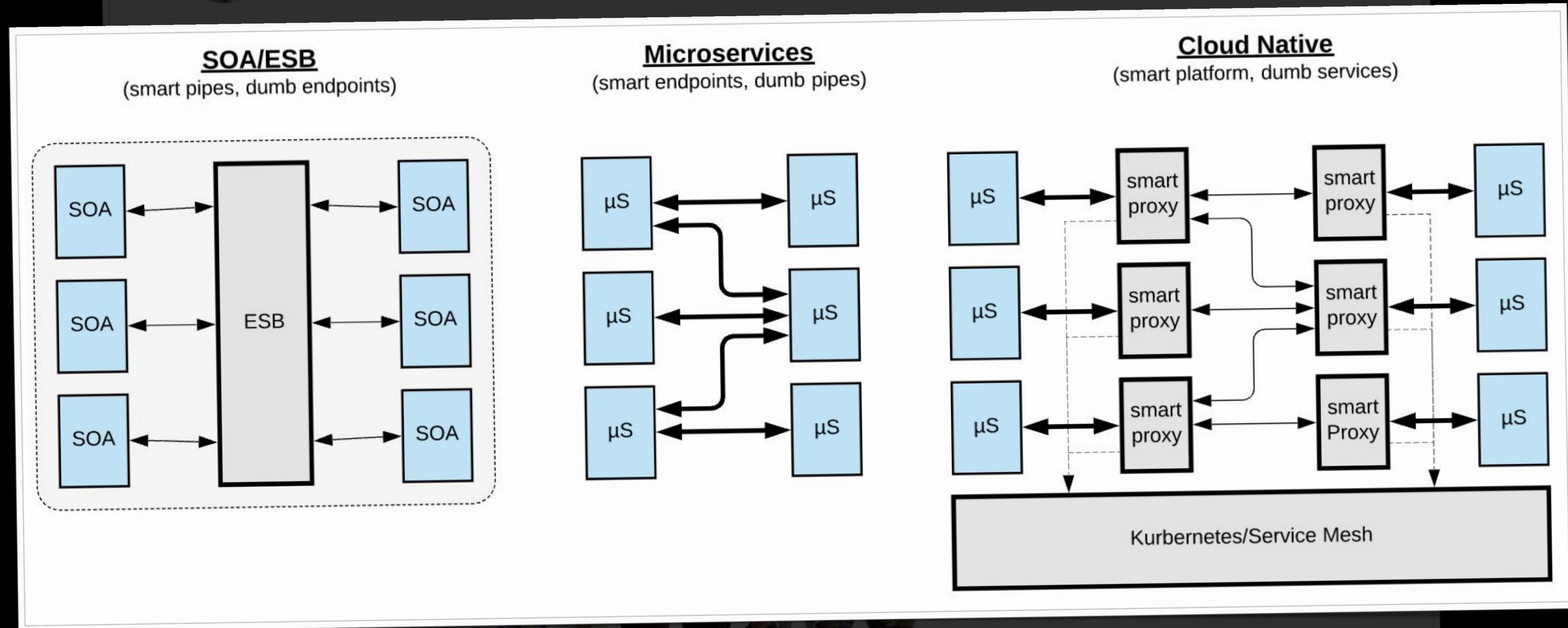
2 Retweets 7 Likes





Daniel Bryant
@danielbryantuk

Following



(a great
article, btw)

www.infoq.com/articles/microservices-post-kuberr

InfoQ

En | 中文 | 日本 | Fr | Br

1,201,360 Aug unique visitors

Development | Architecture & Design | AI, ML & Data Engineering | Culture & Methods | DevOps | Videos with Transcripts

Streaming | Machine Learning | Reactive | Microservices | Containers | .NET | All topics

You are here: InfoQ Homepage ▶ Articles ▶ Microservices in a Post-Kubernetes Era

Microservices in a Post-Kubernetes Era

Like / Posted by Bilgin Ibryam, reviewed by Daniel Bryant on Sep 01, 2018. Estimated reading time: 8 minutes / 1 Discuss

Share + | Twitter | LinkedIn | Facebook | Email

Reading List | Read later

Key Takeaways

- The microservice architecture is still the most popular architectural style for distributed systems. But Kubernetes and the cloud native movement has redefined certain aspects of application design and development at scale.
- On a cloud native platform, observability of services is not enough. A more fundamental prerequisite is to make microservices automatable, by implementing health checks, reacting to signals, declaring resource consumption, etc.
- In the post-Kubernetes era, using libraries to implement operational networking

RELATED CONTENT

eBay Replatforming to Kubernetes, Envoy and Kafka: Intending to Open Source Hardware and Software Sep 16, 2018

Networking Your Microservices Applications Sep 21, 2018

Designing Events-First Microservices Sep 13, 2018



#IBMCLOUDGarage

(a great
article, btw)

www.infoq.com/articles/microservices-post-kuberr

InfoQ

Development Architecture & Design AI, ML & Data Engineering Culture & Methods DevOps Videos with Transcripts

Streaming Machine Learning Reactive Microservices Containers .NET All topics

The Architects' Newsletter

Microservices

: 8 minutes / 1 Discuss

Share + | Twitter | LinkedIn | Facebook | Email

Reading List Read later

Key Takeaways

- The microservice architecture is still the most popular architectural style for distributed systems. But Kubernetes and the cloud native movement has redefined certain aspects of application design and development at scale.
- On a cloud native platform, observability of services is not enough. A more fundamental prerequisite is to make microservices automatable, by implementing health checks, reacting to signals, declaring resource consumption, etc.
- In the post-Kubernetes era, using libraries to implement operational networking

RELATED CONTENT

eBay Replatforming to Kubernetes, Envoy and Kafka: Intending to Open Source Hardware and Software Sep 16, 2018

Networking Your Microservices Applications Sep 21, 2018

Designing Events-First Microservices Sep 13, 2018

Software Development Conference San Francisco Nov 5-9, 2018 London Mar 4 – 8, 2019



The screenshot shows the homepage of the Cloud Native Computing Foundation (CNCF) website. The header features the CNCF logo and navigation links for About, Projects, Certification, People, Community, and Newsroom. A prominent pink "JOIN NOW" button is located in the top right corner. Below the header, there are four main sections: "contributors to CNCF projects", "for free Kubernetes EdX course", "Kubernetes Distributions and Platforms", and "Meetup members". The central content area features a large blue banner with the heading "What is CNCF?". The text explains that CNCF is an open source software foundation dedicated to making cloud native computing universal and sustainable. It describes how cloud native computing uses an open source software stack to deploy applications as microservices, package them into containers, and dynamically orchestrate them to optimize resource utilization. The text concludes by stating that cloud native technologies enable software developers to build great products faster. A smaller "JOIN" button is located at the bottom of the central content area. To the right of the central content, there is a vertical sidebar with social media icons for Twitter, LinkedIn, YouTube, RSS, GitHub, and Slack.

www.cncf.io

CLOUD NATIVE COMPUTING FOUNDATION

About Projects Certification People Community Newsroom

JOIN NOW

for free Kubernetes EdX course

Kubernetes Distributions and Platforms

Meetup members

What is CNCF?

CNCF is an open source software foundation dedicated to making cloud native computing universal and sustainable. Cloud native computing uses an open source software stack to deploy applications as microservices, packaging each part into its own container, and dynamically orchestrating those containers to optimize resource utilization. Cloud native technologies enable software developers to build great products faster.

JOIN

The screenshot shows the homepage of the Cloud Native Computing Foundation (CNCF) at www.cncf.io. The header features the CNCF logo and navigation links for About, Projects, Certification, People, Community, and Newsroom. A pink "JOIN NOW" button and a search icon are also present. Below the header, there are four main sections: "contributors to CNCF projects", "for free Kubernetes EdX course", "Kubernetes Distributions and Platforms", and "Meetup members". The central content area has a white background with a large, bold, dark blue text overlay that reads "What is CNCF? microservices containers dynamically orchestrated". To the right of this text is a vertical column of social media icons for Twitter, LinkedIn, YouTube, RSS, GitHub, and Slack. At the bottom of the page is a pink "JOIN" button.

What is CNCF?

microservices containers dynamically orchestrated



“the cloud native
computing
foundation is wrong
...
about cloud native.”





“the cloud native
computing
foundation is wrong
...
about cloud native.”



Holly



Kasper Nissen @GOTOCPH
@phennex

Follow



If you are attending #gotocph and want to discuss @CloudNativeFdn tech, please feel free to PM me or find me at the venue the next couple of days. #cloudnative

2:34 PM - 18 Nov 2018 from Copenhagen, Denmark

4 Retweets 4 Likes



4



4



4



The screenshot shows the homepage of the Cloud Native Computing Foundation (CNCF) website. The header features the CNCF logo and navigation links for About, Projects, Certification, People, Community, and Newsroom. A prominent pink "JOIN NOW" button is located in the top right corner. Below the header, there are four main sections: "contributors to CNCF projects", "for free Kubernetes EdX course", "Kubernetes Distributions and Platforms", and "Meetup members". The central content area features a large blue banner with the heading "What is CNCF?". The text explains that CNCF is an open source software foundation dedicated to making cloud native computing universal and sustainable. It describes how cloud native computing uses an open source software stack to deploy applications as microservices, package them into containers, and dynamically orchestrate them to optimize resource utilization. The text concludes by stating that cloud native technologies enable software developers to build great products faster. A smaller "JOIN" button is located at the bottom of the central content area. To the right of the central content, there is a vertical sidebar with social media icons for Twitter, LinkedIn, YouTube, RSS, GitHub, and Slack.

www.cncf.io

CLOUD NATIVE COMPUTING FOUNDATION

About Projects Certification People Community Newsroom

JOIN NOW

for free Kubernetes EdX course

Kubernetes Distributions and Platforms

Meetup members

What is CNCF?

CNCF is an open source software foundation dedicated to making cloud native computing universal and sustainable. Cloud native computing uses an open source software stack to deploy applications as microservices, packaging each part into its own container, and dynamically orchestrating those containers to optimize resource utilization. Cloud native technologies enable software developers to build great products faster.

JOIN

The screenshot shows the homepage of the Cloud Native Computing Foundation (CNCF) website. The header features the CNCF logo and navigation links for About, Projects, Certification, People, Community, and Newsroom. A "JOIN NOW" button and a search icon are also present. Below the header, there are four main sections: "contributors to CNCF projects", "for free Kubernetes EdX course", "Kubernetes Distributions and Platforms", and "Meetup members". The central focus is a large, bold text "What is CNCF? build great products faster". To the right of this text is a vertical column of social media icons for Twitter, LinkedIn, YouTube, RSS, GitHub, and Slack. The URL "www.cncf.io" is visible in the browser's address bar.

www.cncf.io

CLOUD NATIVE COMPUTING FOUNDATION

About Projects Certification People Community Newsroom

JOIN NOW

JOIN NOW

contributors to CNCF projects

for free Kubernetes EdX course

Kubernetes Distributions and Platforms

Meetup members

What is CNCF?

build great products faster

CNCF is an open source software foundation dedicated to making cloud native computing universal and sustainable. Cloud native computing uses an open source software stack to deploy applications as microservices, packaging each part into its own container, and dynamically orchestrating those containers to optimize resource utilization. Cloud native technologies enable software developers to build great products faster.

JOIN

why?

what **problem** are
we trying to solve?



#IBMCloudGarage

@holly_cummins

CV-driven development.



“my CV looks dull”
is not a good reason
to go cloud native

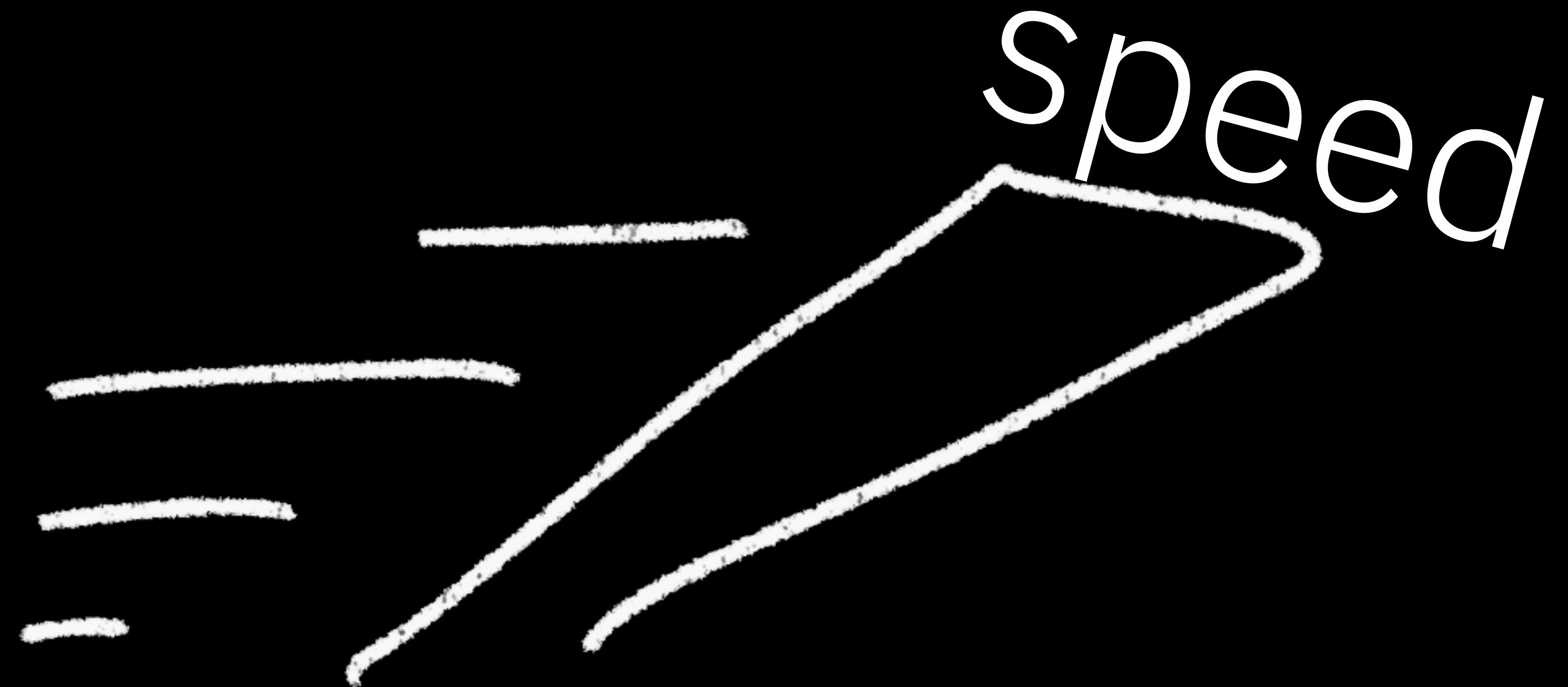
why cloud?

cost





elasticity



exotic capabilities



why cloud native?



#IBMCLOUDGarage

@holly_cummins

12

factors

12
factors

how to write a
cloud application
so you don't get
electrocuted

#IBMCLOUDGarage

@holly_cummins

#IBMCLOUDGarage

@holly_cummins

cloud native is not a
synonym for
‘microservices’

#IBMCLOUDGarage

@holly_cummins

if ‘cloud native’ has to be a
synonym for anything, it would be
‘idempotent’

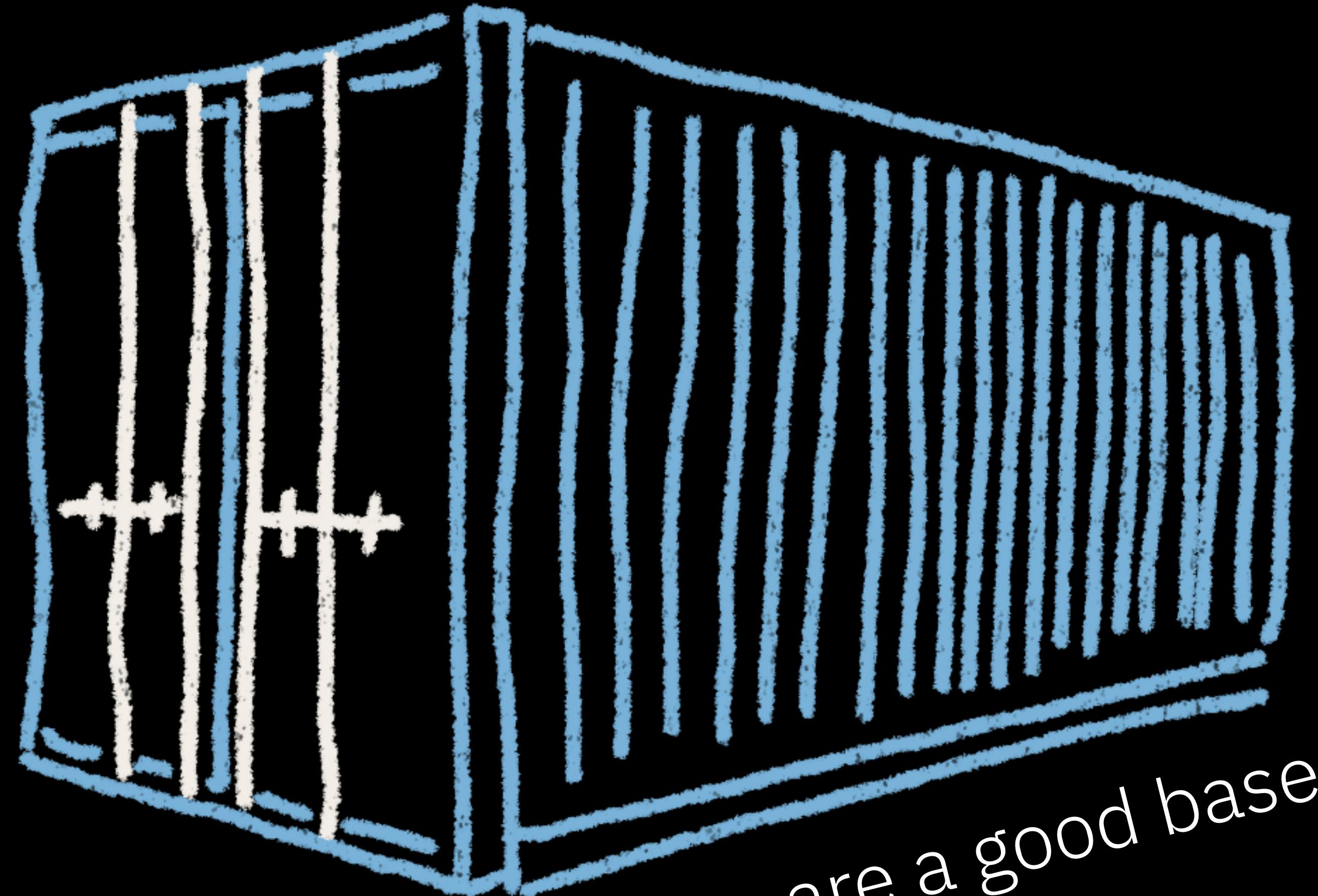
if ‘cloud native’ has to be a
synonym for anything, it would be
‘idempotent’

which definitely needs a synonym



containers are a good base

it's not a
competition
to see how
many you
can have



containers are a good base

you do not need intra-app http
communication to be cloud native



#IBMCLOUDGarage

@holly_cummins

complexity adds expense

unnecessary complexity
adds unnecessary expense

#IBMCLOUDGarage

@holly_cummins

space pencil

space pencil

\$128.89

space pencil

\$128.89

space pen

space pencil

\$128.89

space pen

\$2.39

space pencil

\$128.89

+medical bills

space pen

\$2.39

accidental
complexity

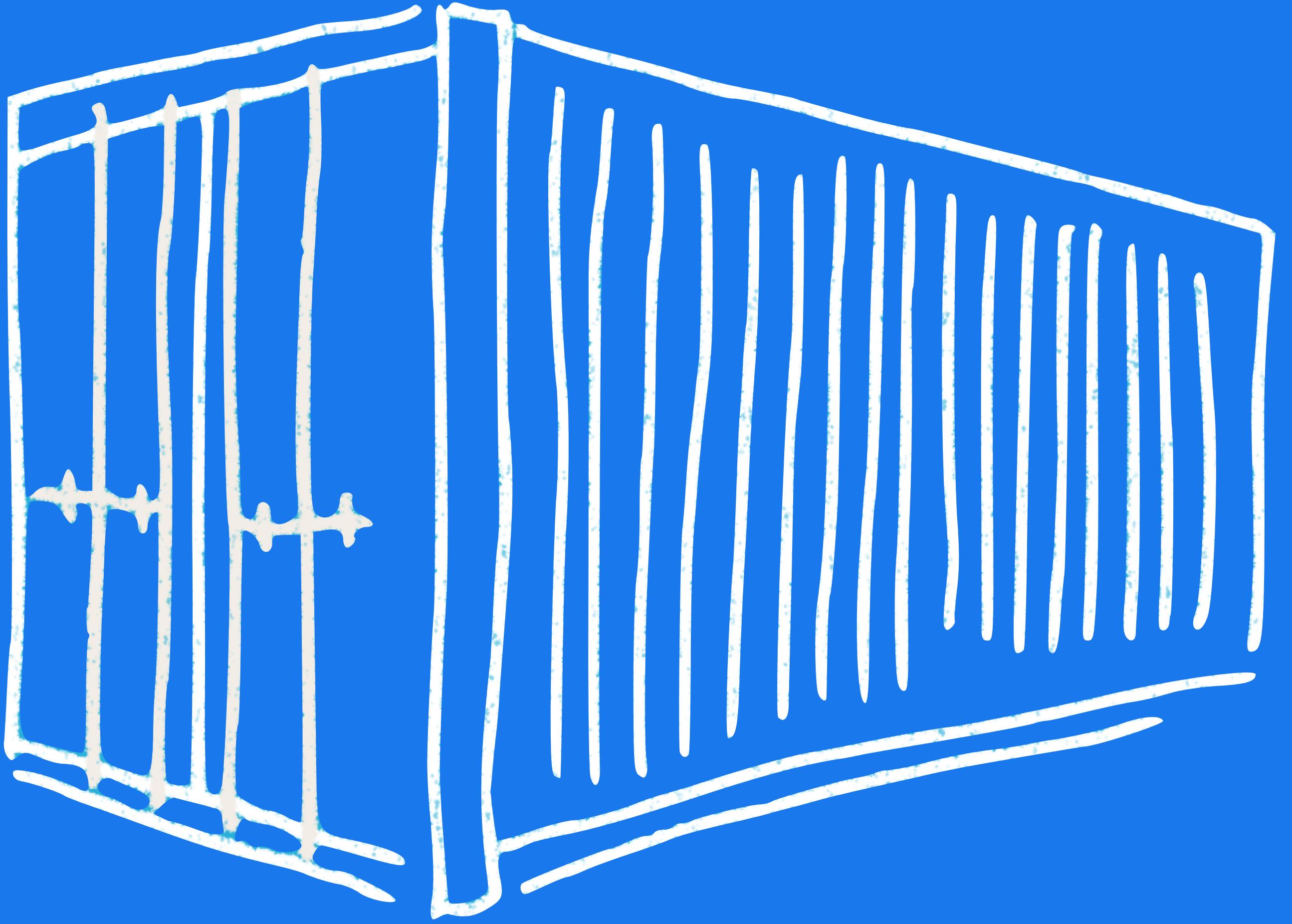
essential
complexity

accidental
complexity

essential
complexity



“this will impress my
boss” complexity



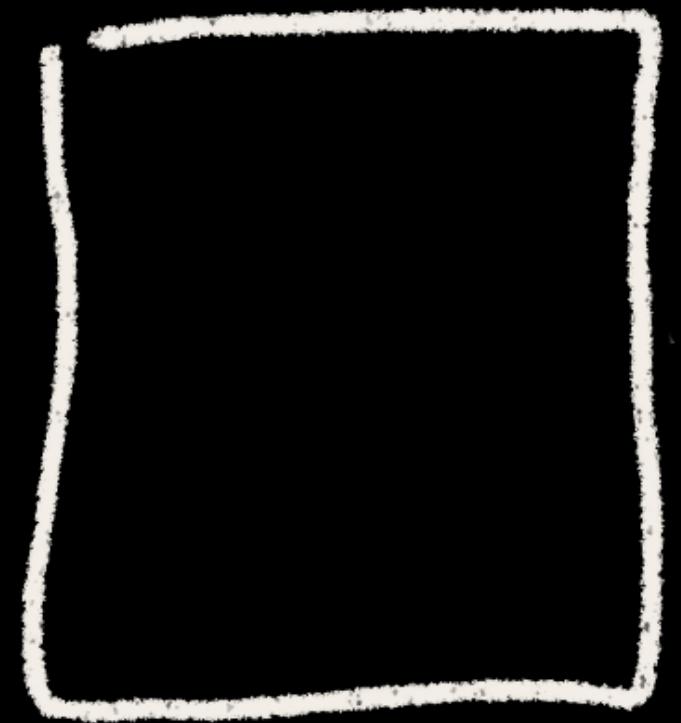
what is a container?



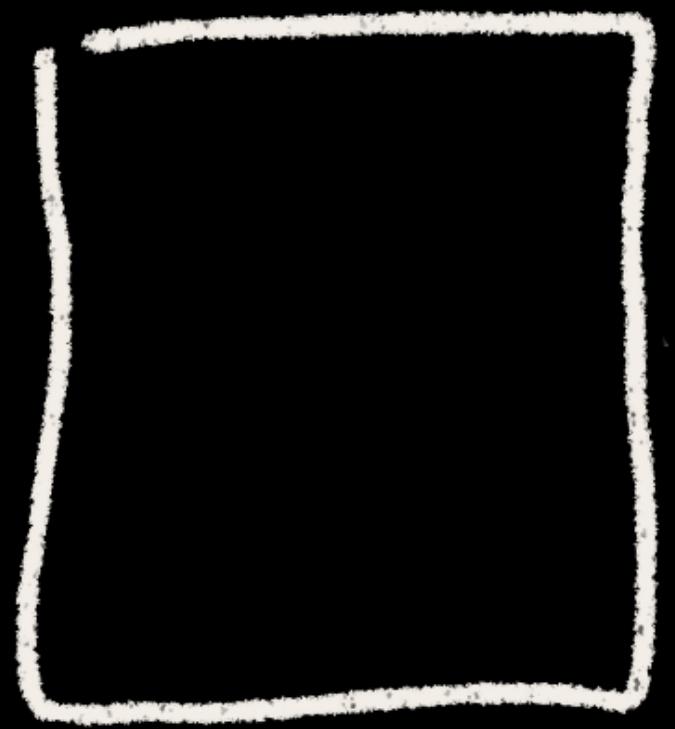
#IBMCLOUDGarage

@holly_cummins

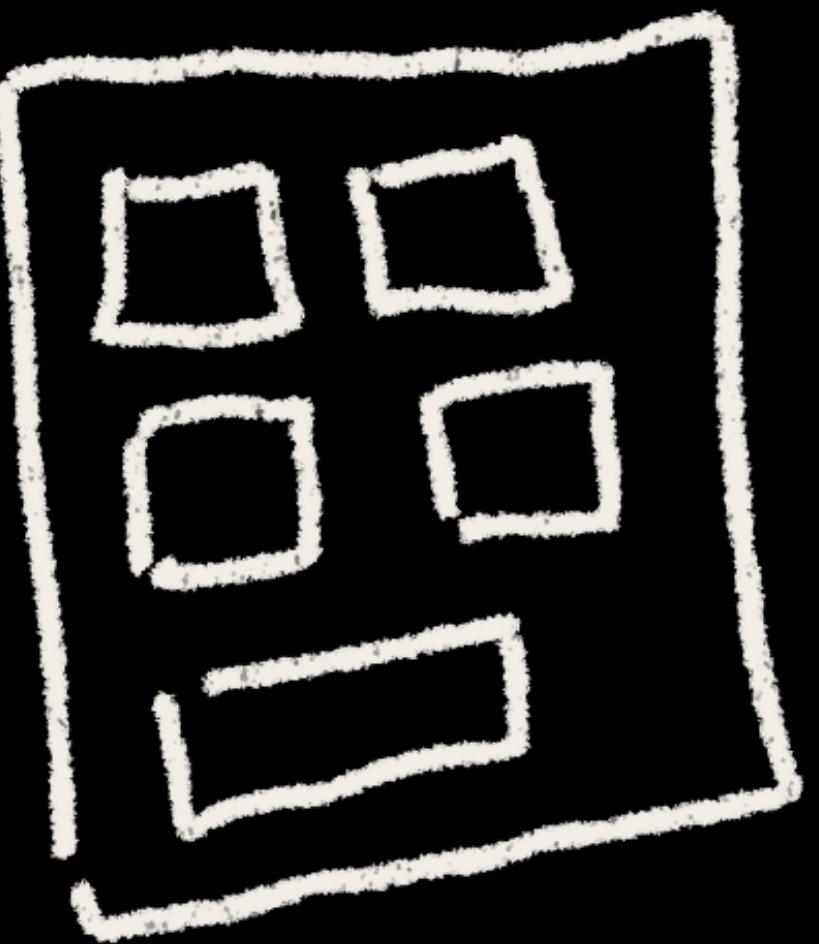
bare metal

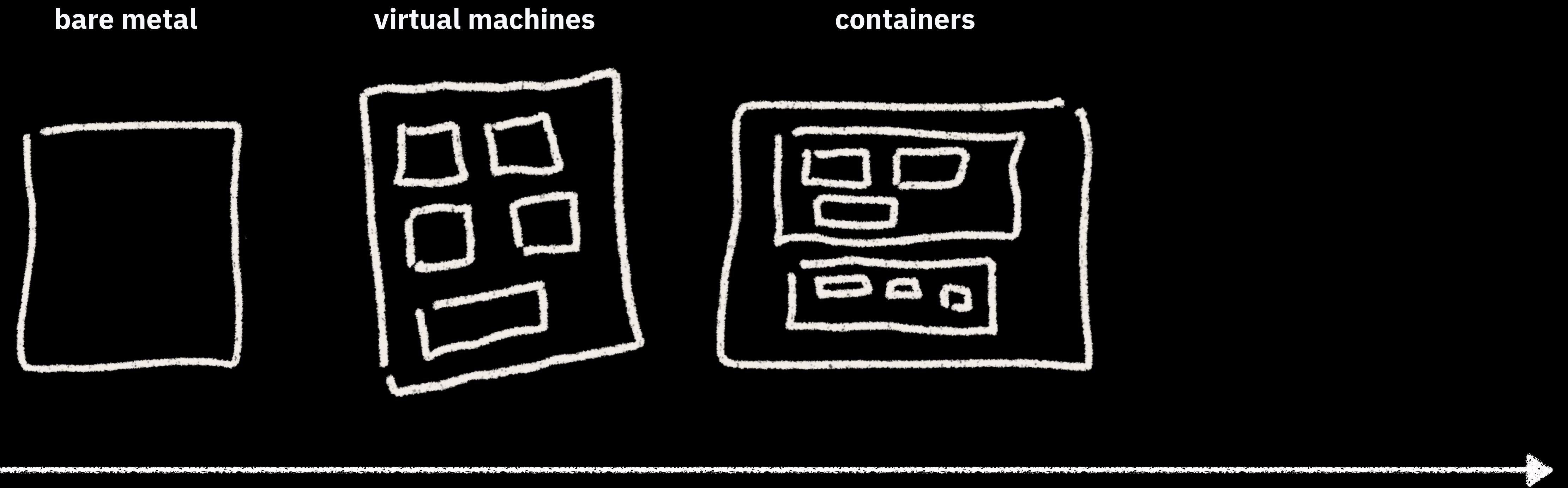


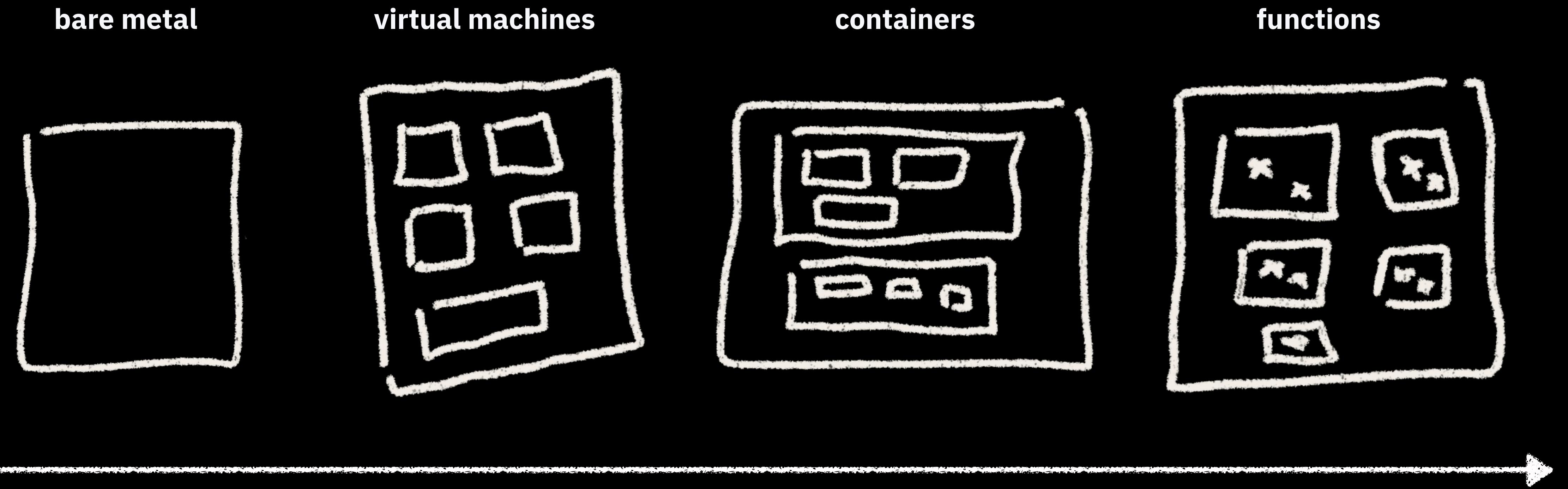
bare metal



virtual machines







funtainers

function-tainers

funtainers



Thermos® FUNtainer®

funtainers

all the fun of containers, but without having to worry about kubernetes

2010 the dawn of cloud native

#IBMCLOUDGarage

The screenshot shows a web browser displaying a blog post on pzf.fremantle.org/2010/05/cloud-native.html. The title of the post is "Cloud Native". The date is listed as MAY 28. The post discusses the concept of "Cloud Native" applications and middleware, comparing them to traditional infrastructure and modern cloud environments like Amazon EC2 and Eucalyptus. It also mentions the analogy of a 6-lane freeway versus a dirt track. The post concludes with a list of attributes for "Cloud Native" software.

Paul Fremantle's Blog SOA, Cloud, Web Services, Synapse, Tin Whistles, Black...

Classic Flipcard Magazine Mosaic Sidebar Snapshot Timeslide

MAY 28

Cloud Native

Together with [Sanjiva](#) and the rest of the WSO2 [architecture](#) team, I've been thinking a lot about what it means for applications and middleware to work well in a cloud environment - on top of an Infrastructure-as-a-Service such as [Amazon EC2](#), [Eucalyptus](#), or [Ubuntu Enterprise Cloud](#).

One of our team - [Lavi](#) - has a great analogy. Think of a 6-lane freeway/motorway/autobahn as the infrastructure. Before the autobahn existed there were forms of transport optimized first to dirt tracks and then to simple tarmac roads. The horse-drawn cart is optimized to a dirt track. On an autobahn it works - but it doesn't go any faster than on a dirt track. A [Ford Model T](#) can go faster, but it can't go safely at autobahn speeds: even if it could accelerate to 100mph it won't steer well enough at that speed or brake quickly enough.

Similarly, existing applications taken and run in a cloud environment may not fully utilize that environment. Even if systems can be clustered they may not be able to dynamically change the cluster size (elasticity). Its not just acceleration, but braking as well! We believe there are a set of these technical attributes that software needs to take account of to work well in a cloud environment. In other words - what do middleware and applications have to do to be *Cloud Native*.

Here are the attributes that we think are the core of "Cloud Native":

- *Distributed / dynamically wired*

In order for an application to work in a cloud environment the system must be inherently distributed and dynamically wired.

#IBMCLOUDGarage

@holly_cummins

behaves well on the cloud

behaves well on the cloud
written for the cloud

this is all how we **run** our
application, not what's **in** it

speed

speed

what's the point of getting the
same old **stuff** to market faster?

what's the point of being able to
respond to the market, if you **don't?**

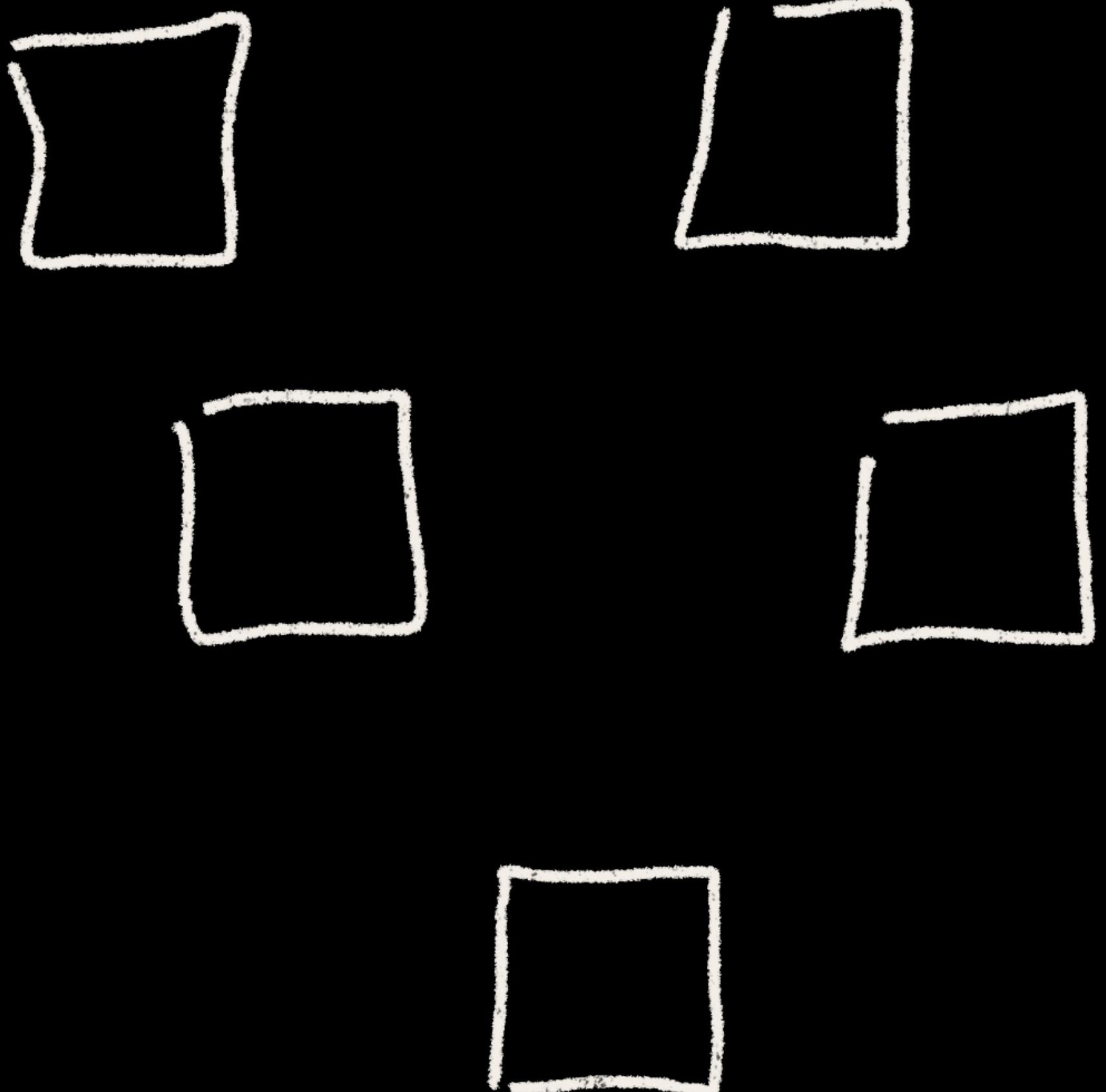
what's the point of
architecture that can go
faster, if you don't go faster?

how to fail at cloud native

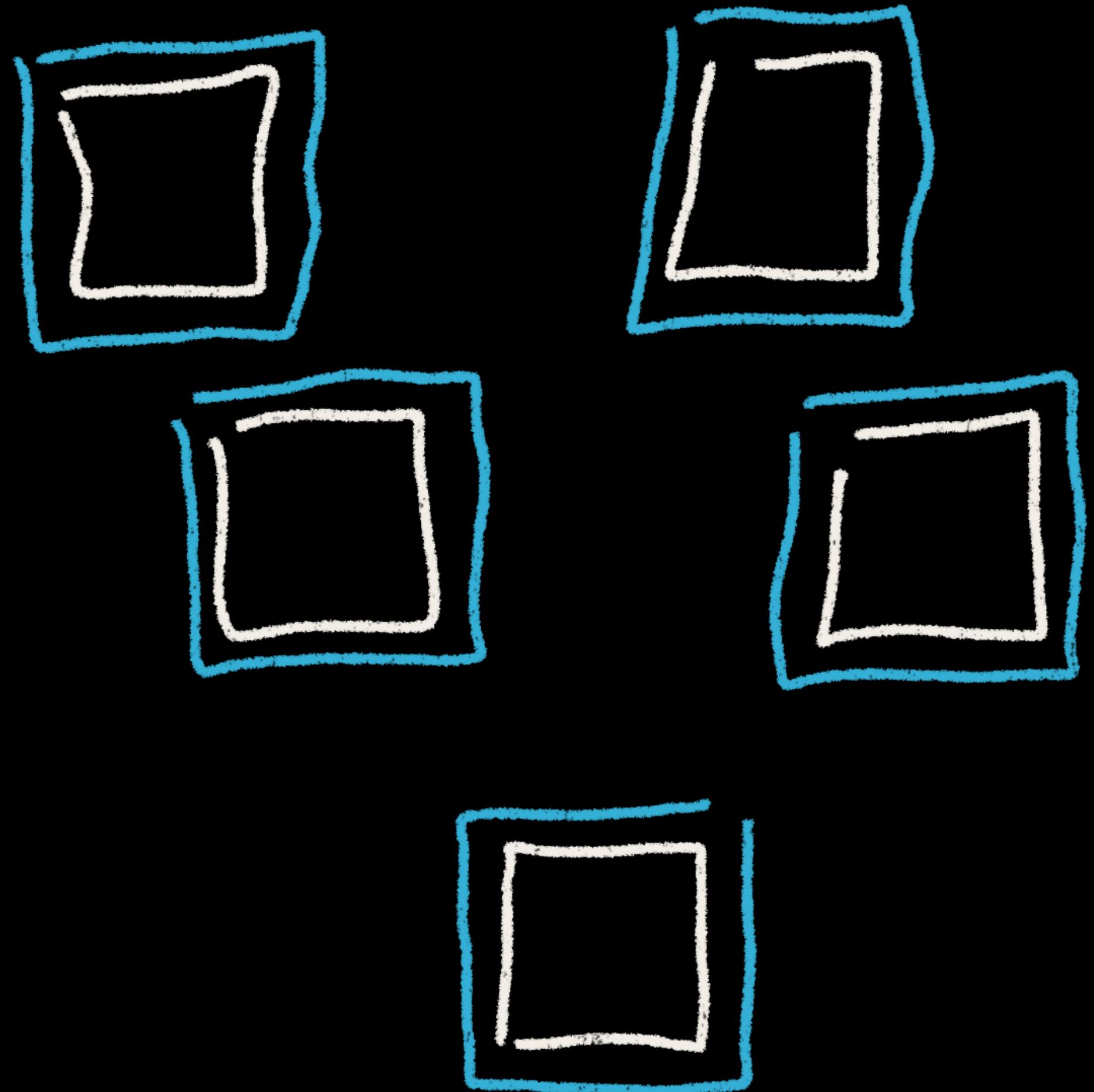


“each of our microservices has
duplicated the same object
model ... with twenty classes
and seventy fields”

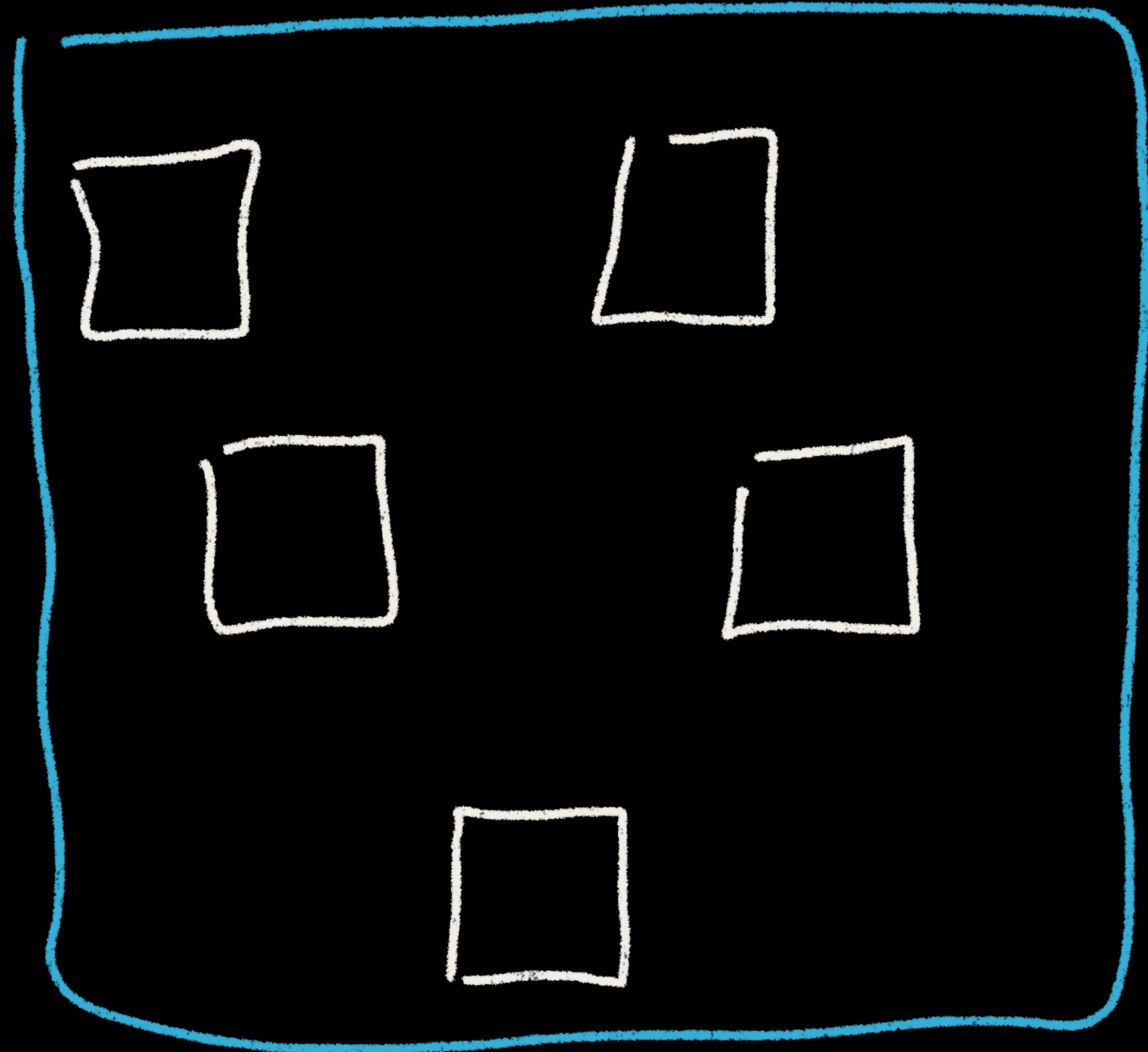
Microservice



Microservice
Domain



Microservice
Domain



“every time we change
code, something breaks”

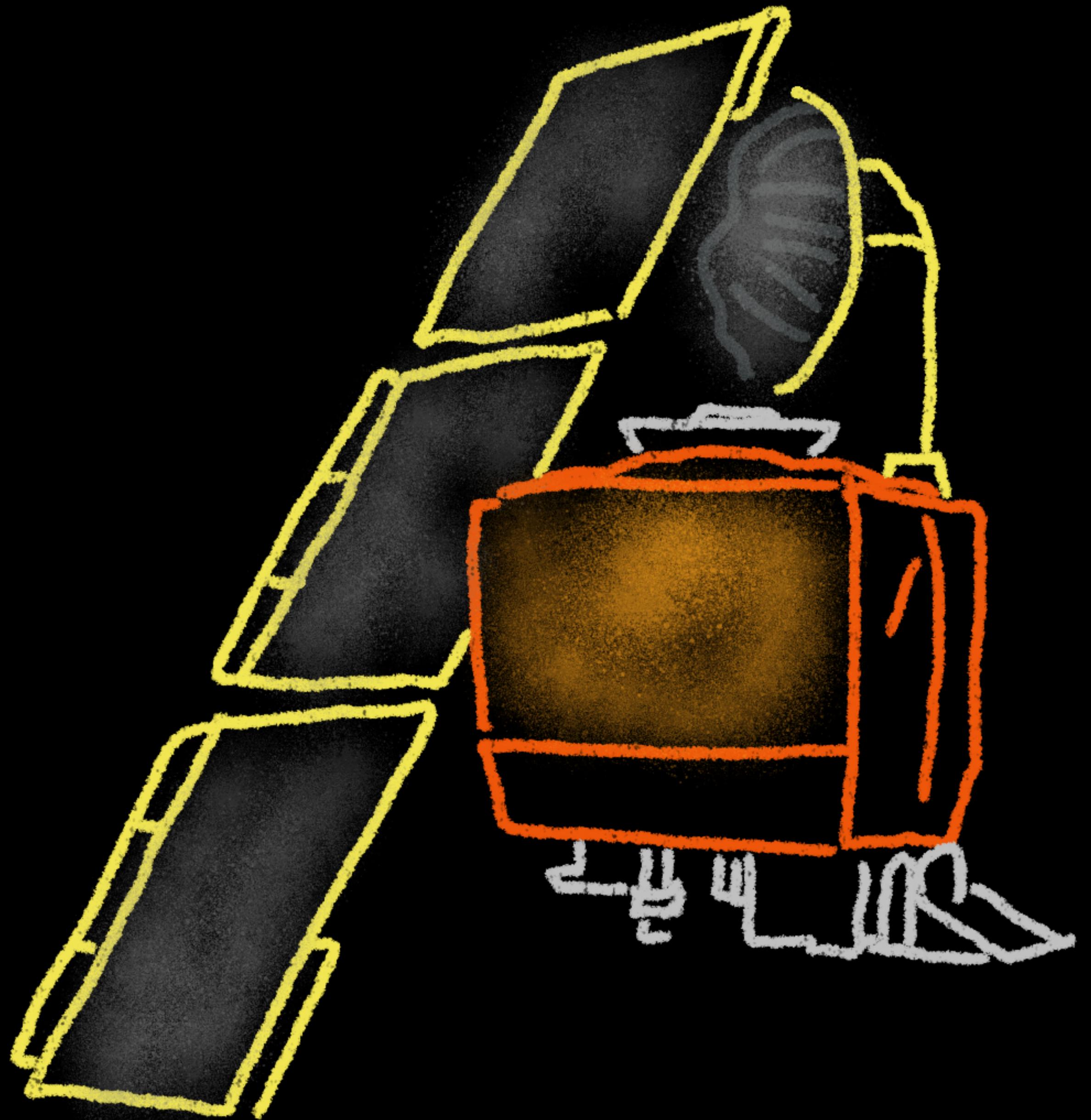
distributed monolith



cloud-native spaghetti is still spaghetti

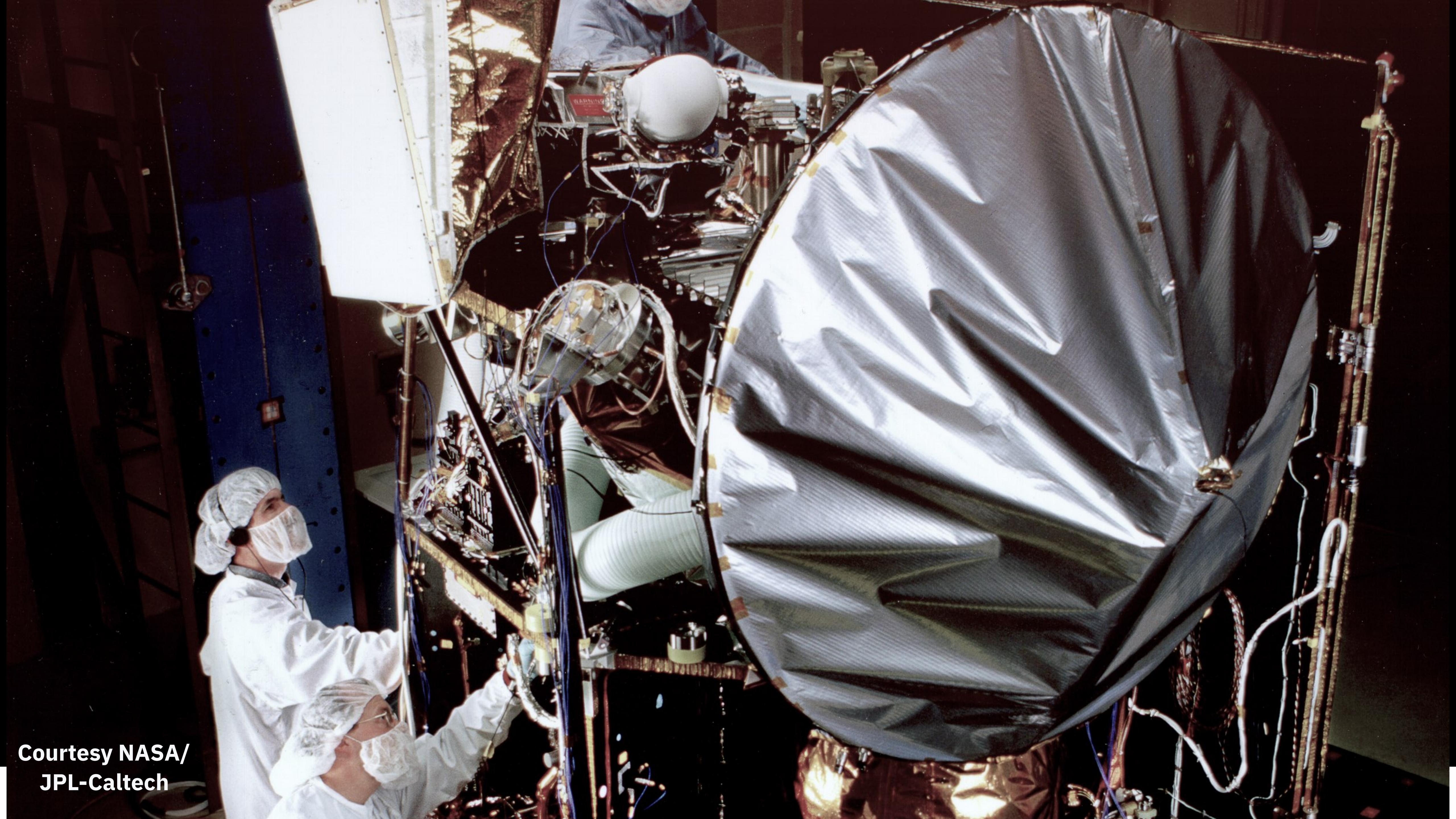
(Image: Cloudy with a Chance of Meatballs.)

just because a system runs across 6
containers doesn't mean it's decoupled

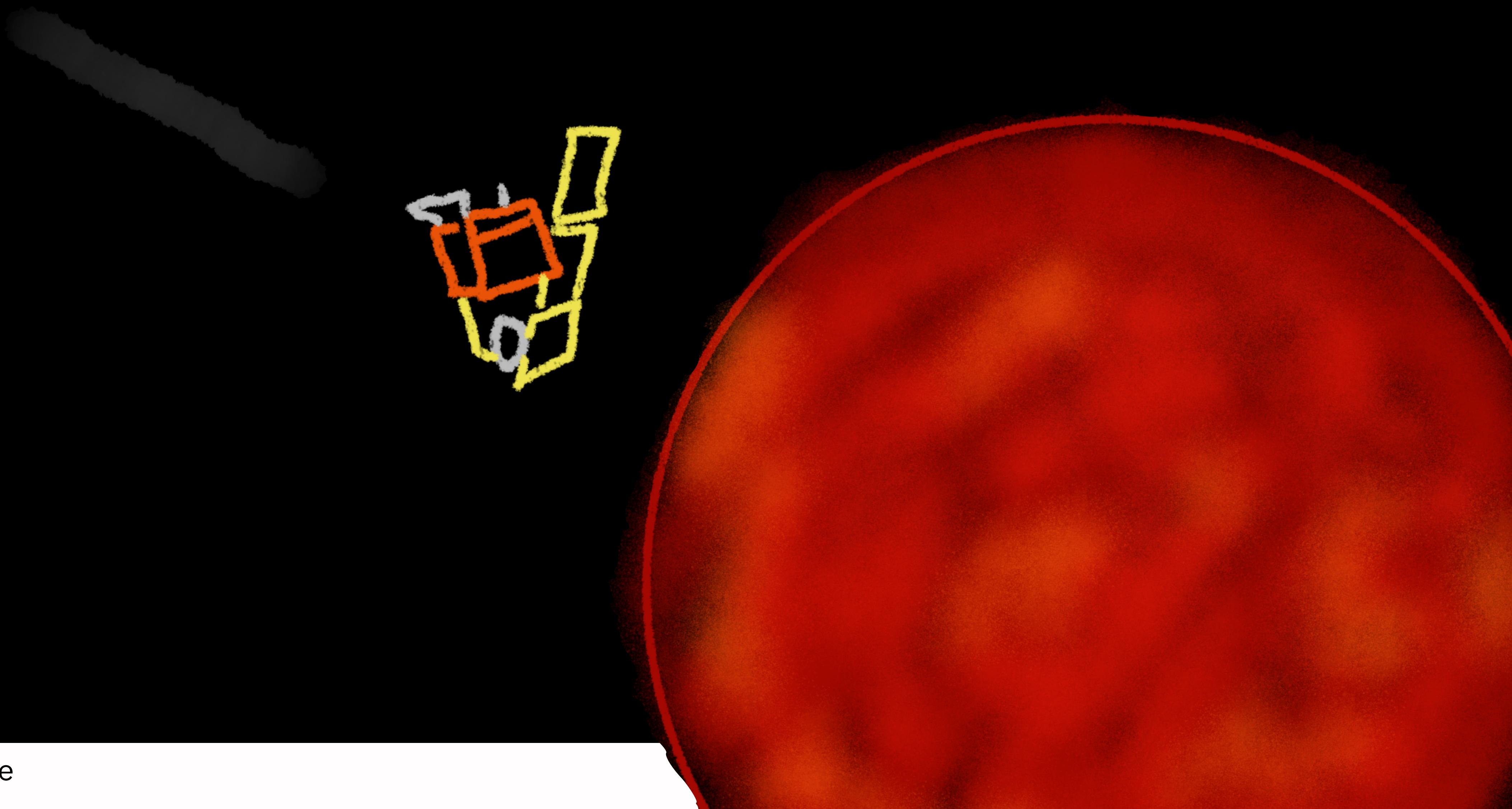


#IBMCLOUDGarage

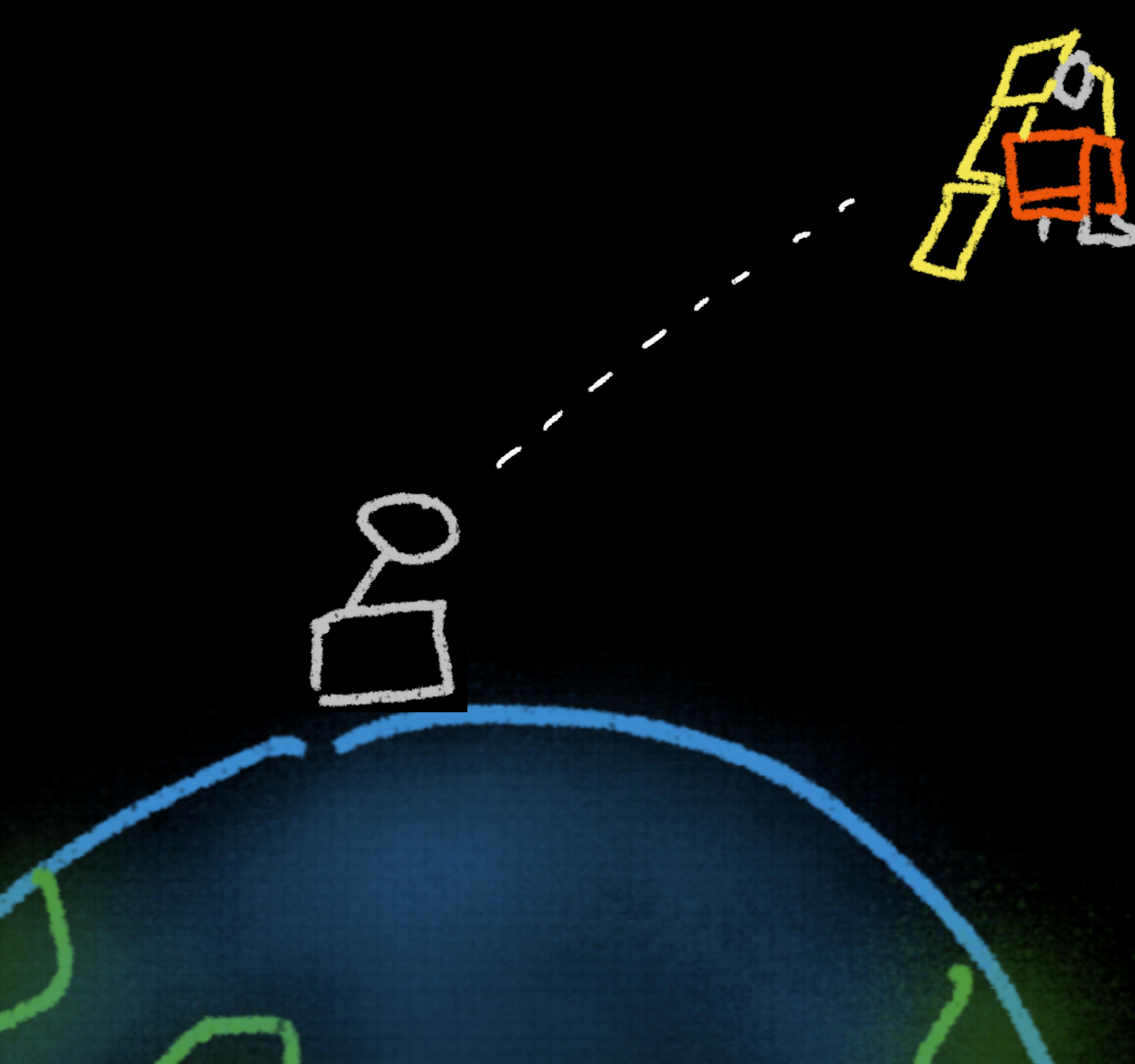
@holly_cummins

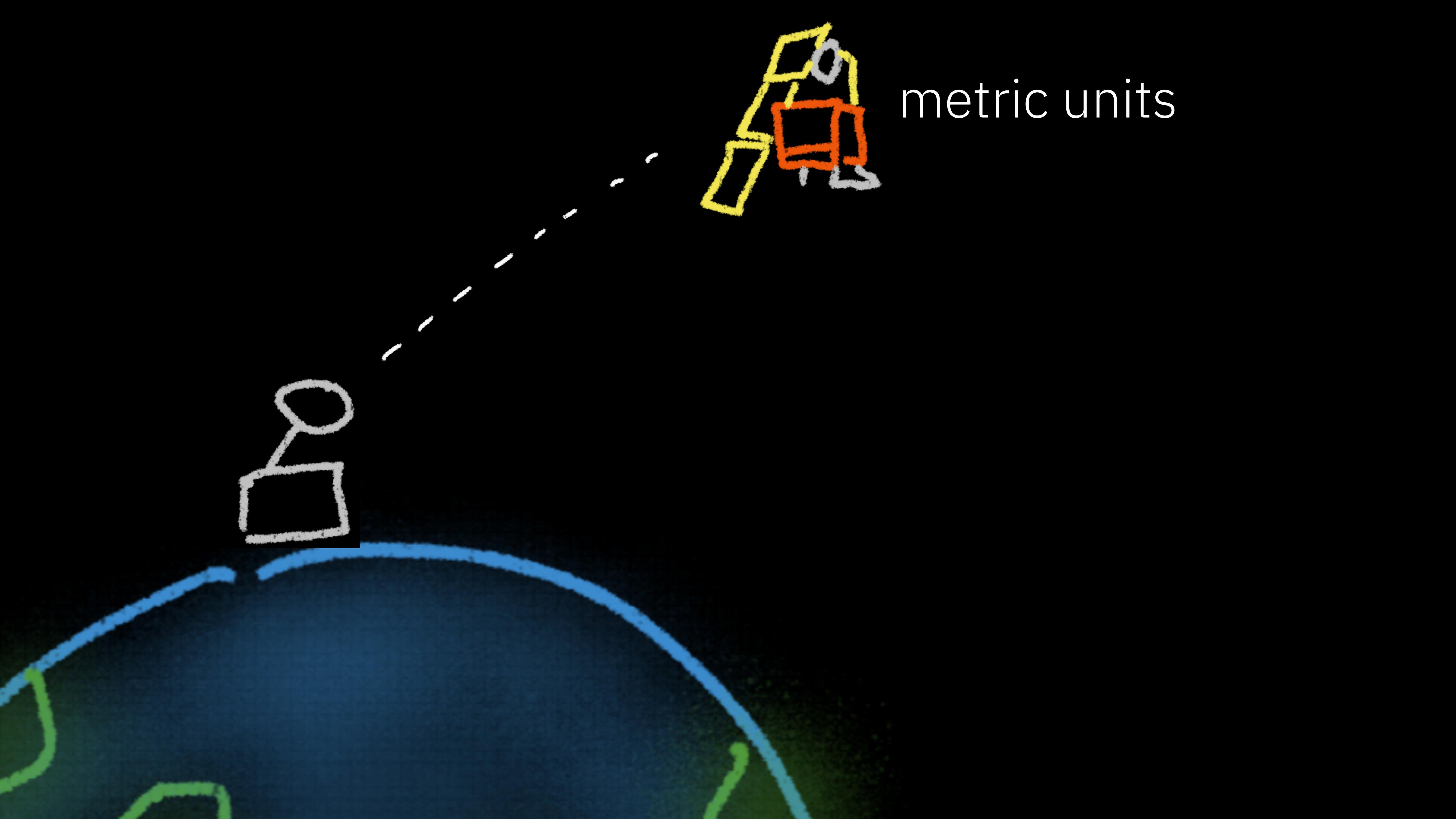


Courtesy NASA/
JPL-Caltech



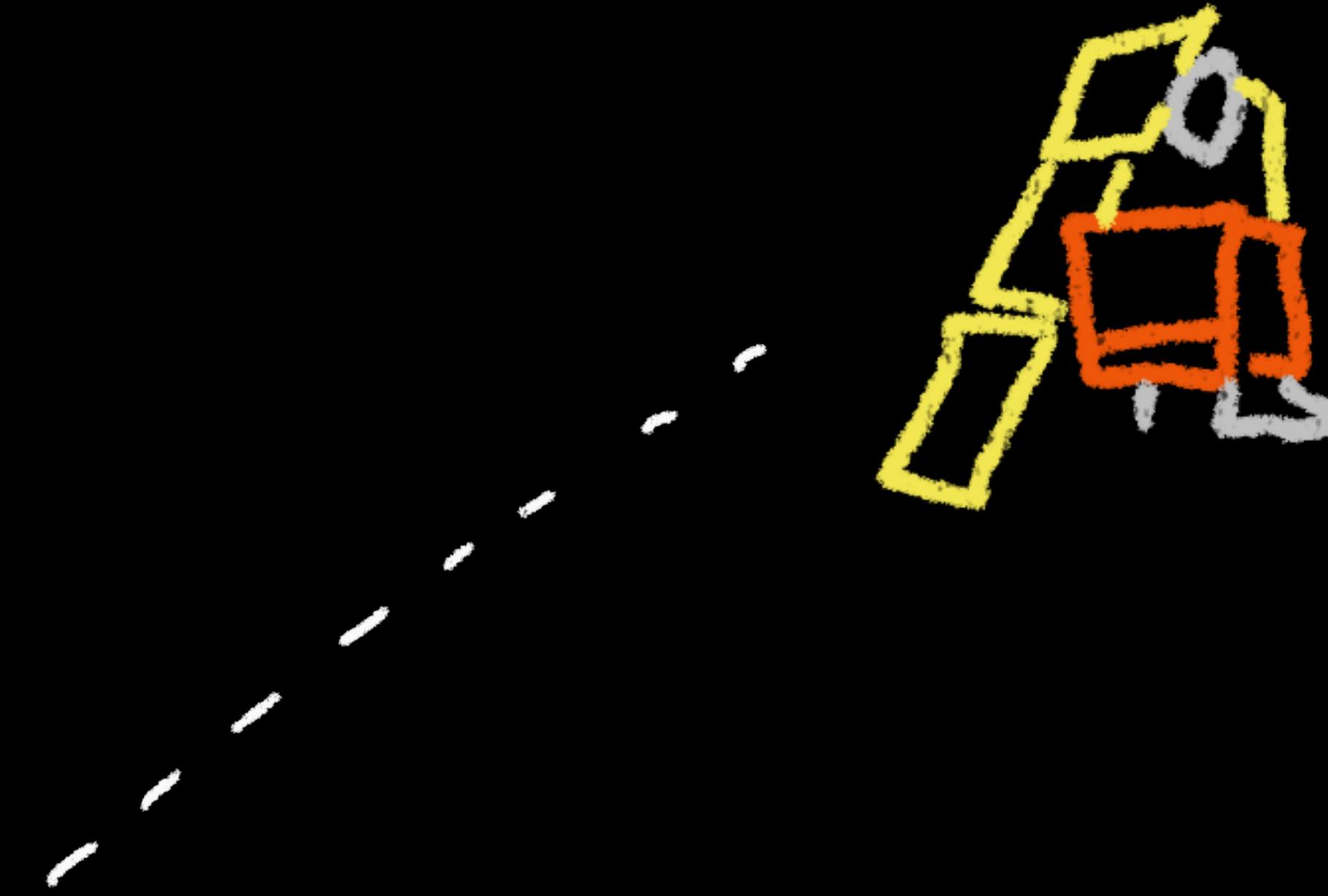
#IBMCLOUDGarage





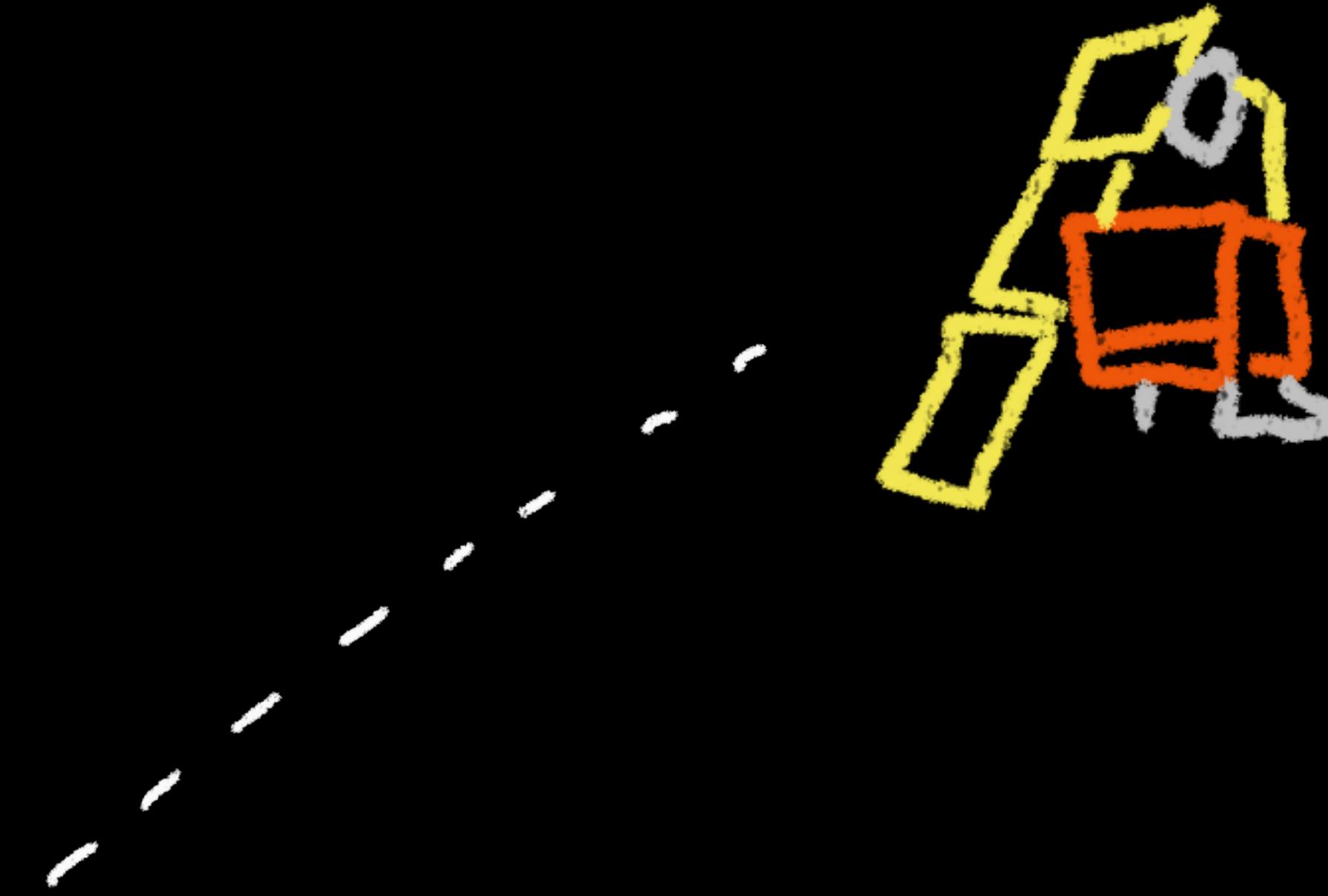
metric units

imperial
units



metric units

imperial
units



metric units

distributing
did not help

microservices **need**
consumer-driven contract tests

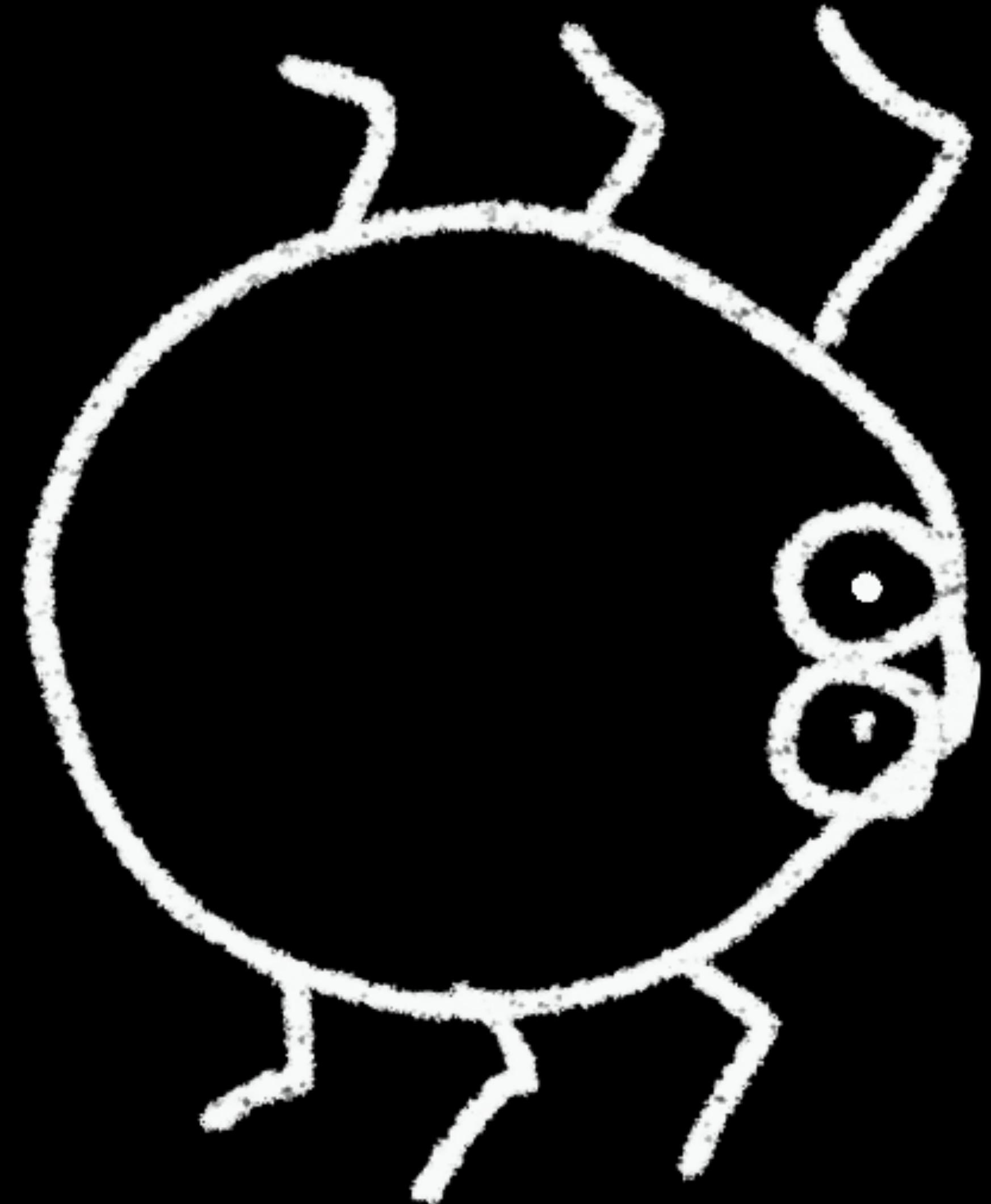
“our tests aren’t
automated”

“we don’t know if
our code works”

systems will behave in
unexpected ways

documentation can be
wrong

dependency updates
can change behaviour



#IBMCLOUDGarage

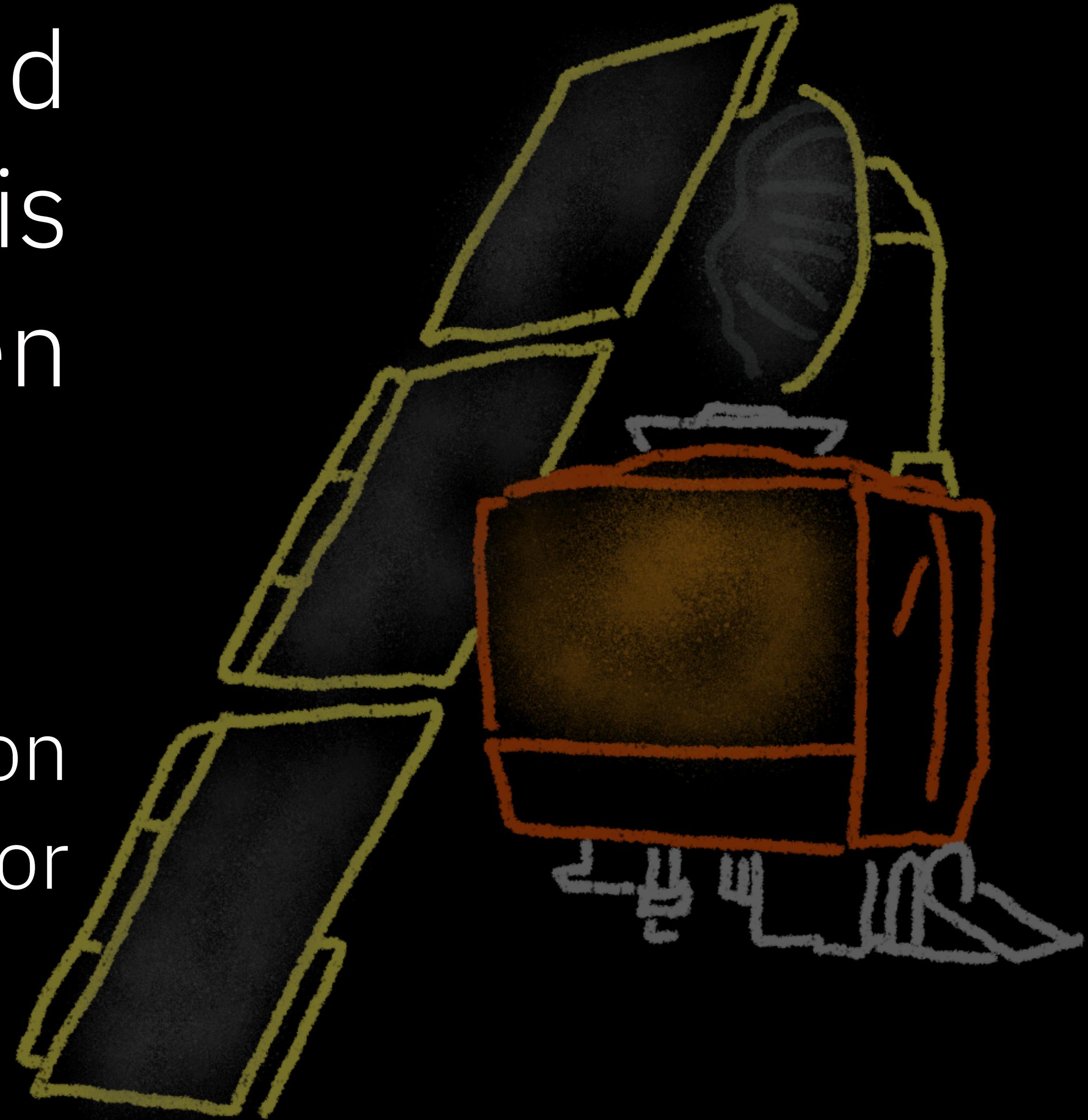
@holly_cummins

#IBMCLOUDGarage

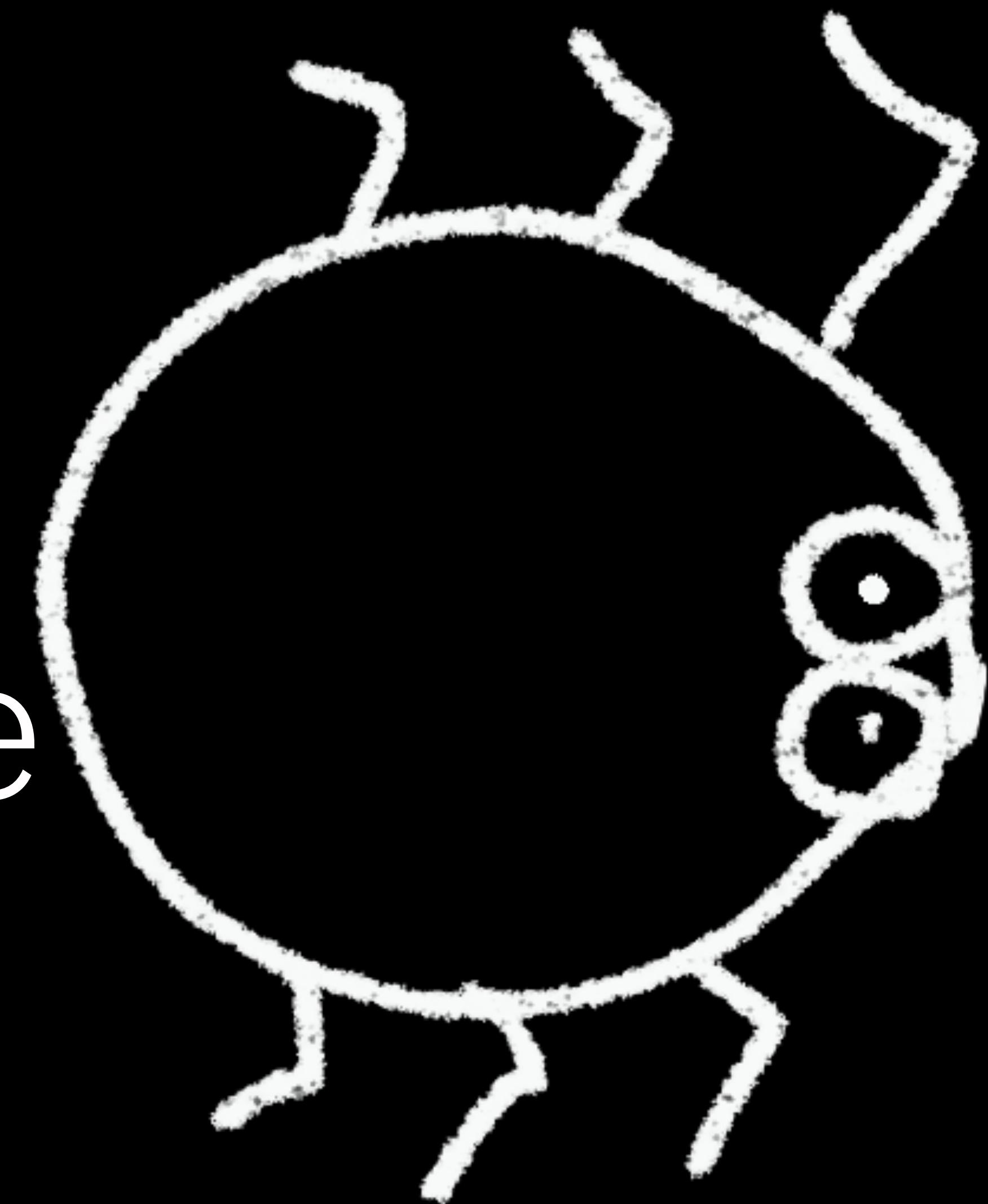
@holly_cummins

“Had we done end-to-end testing, we believe this error would have been caught.”

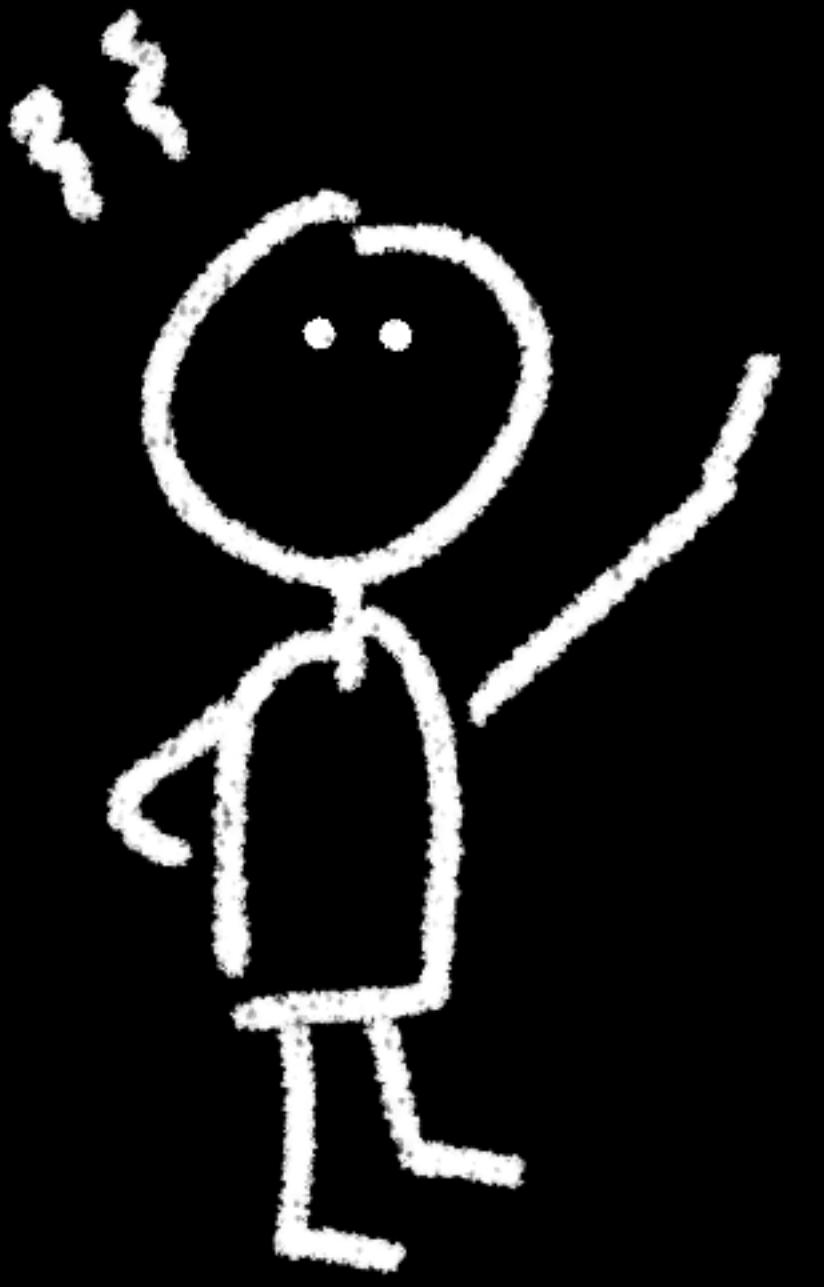
Arthur Stephenson
Chief Investigator



“we can’t ship
until we have
more confidence
in the quality”



microservices **need**
automated integration tests

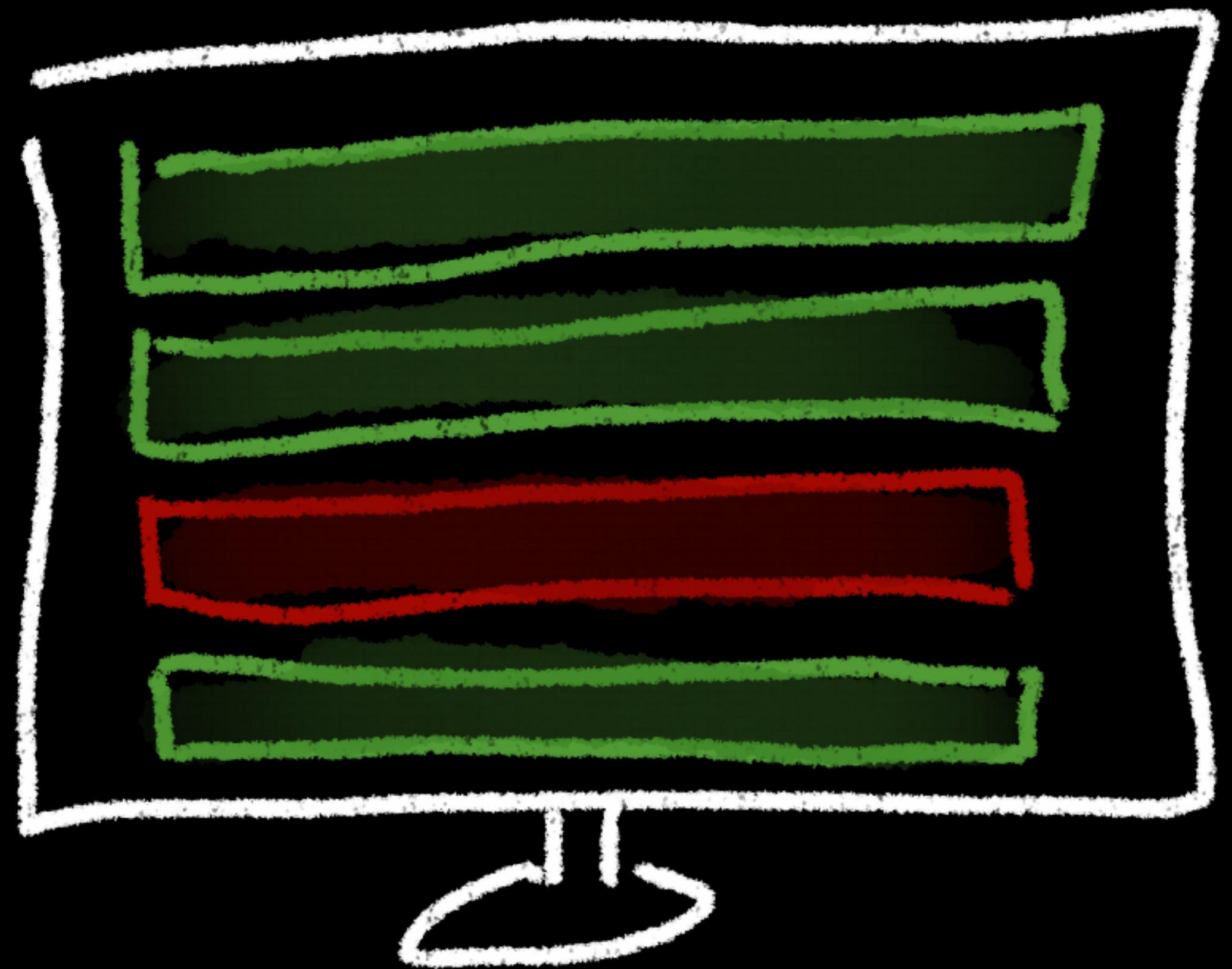


not a good CI/CD indicator

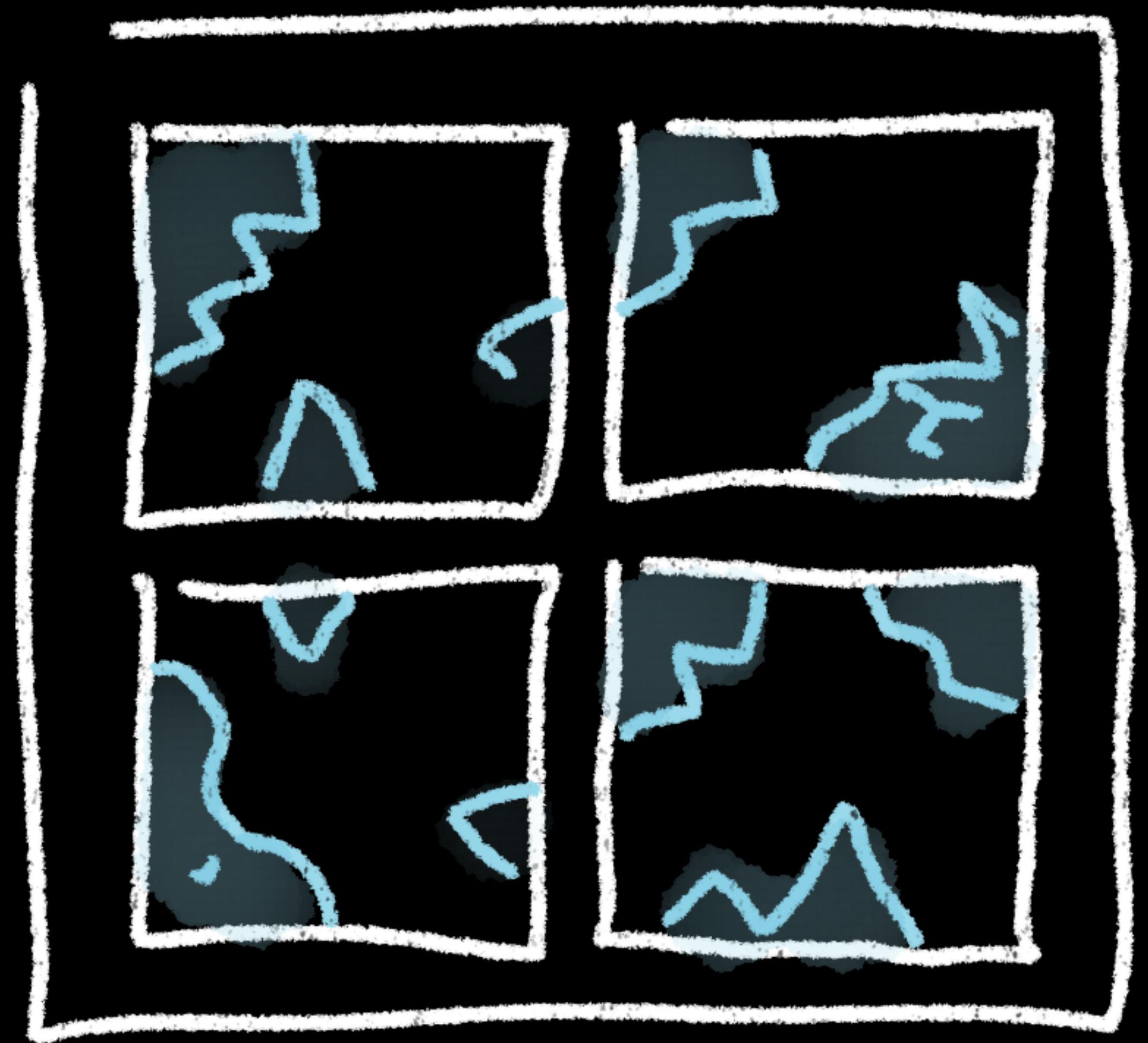


a good CI/CD indicator

“we don’t know when
the build is broken”

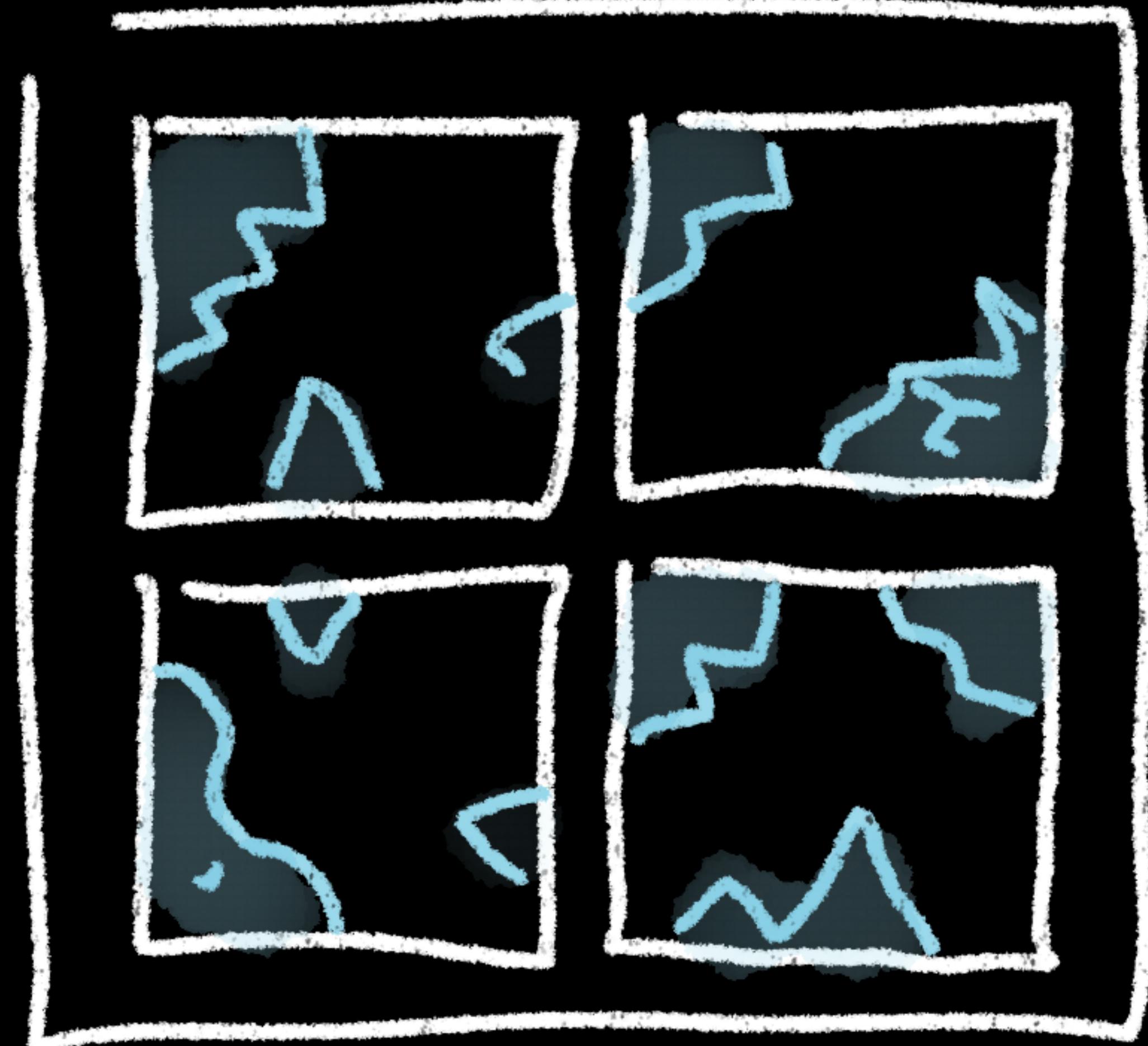


a good build radiator



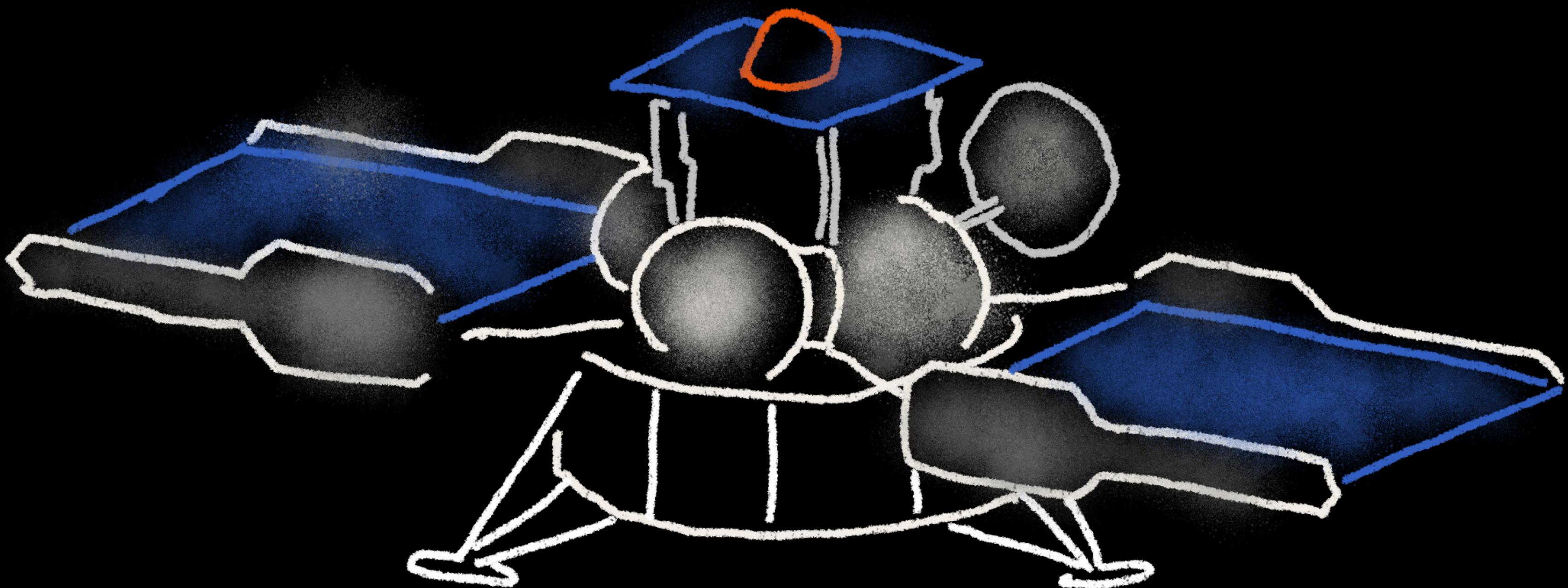
#IBMCLOUDGarage

@holly_cummins

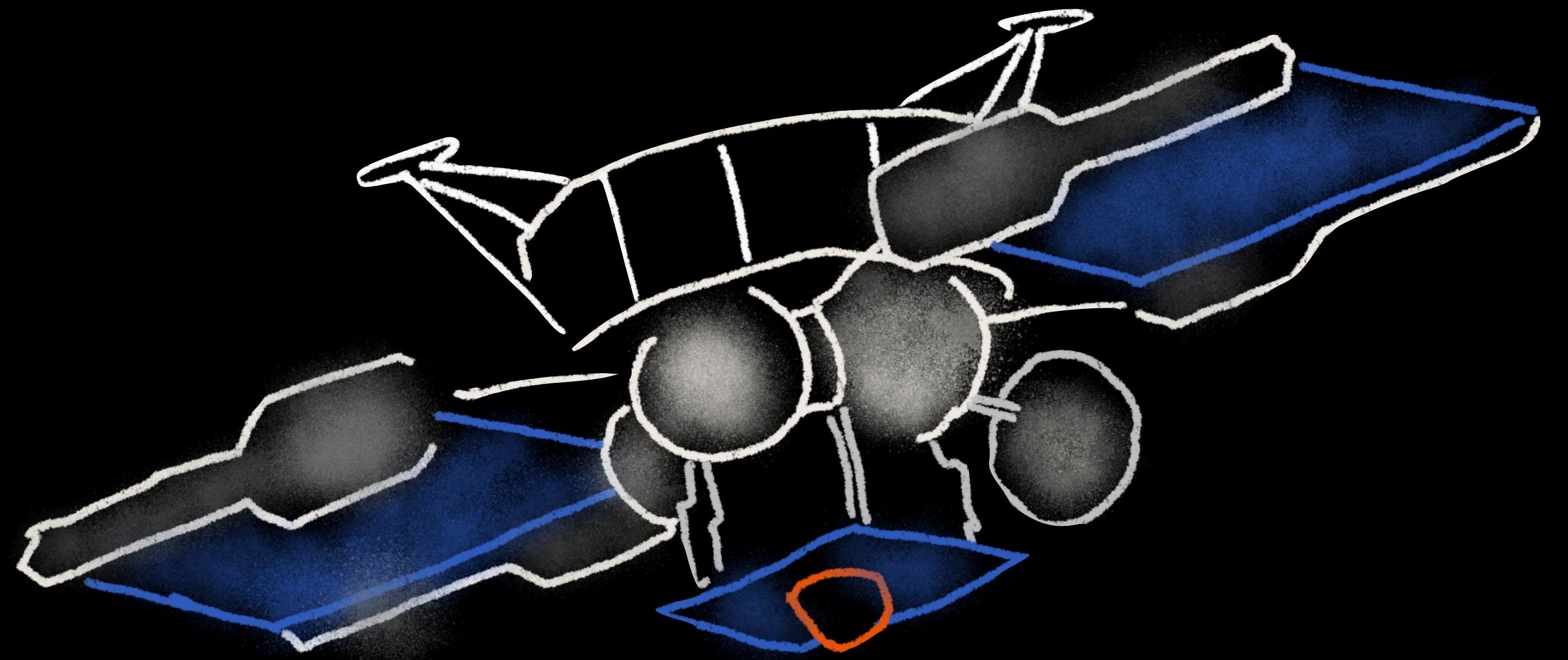


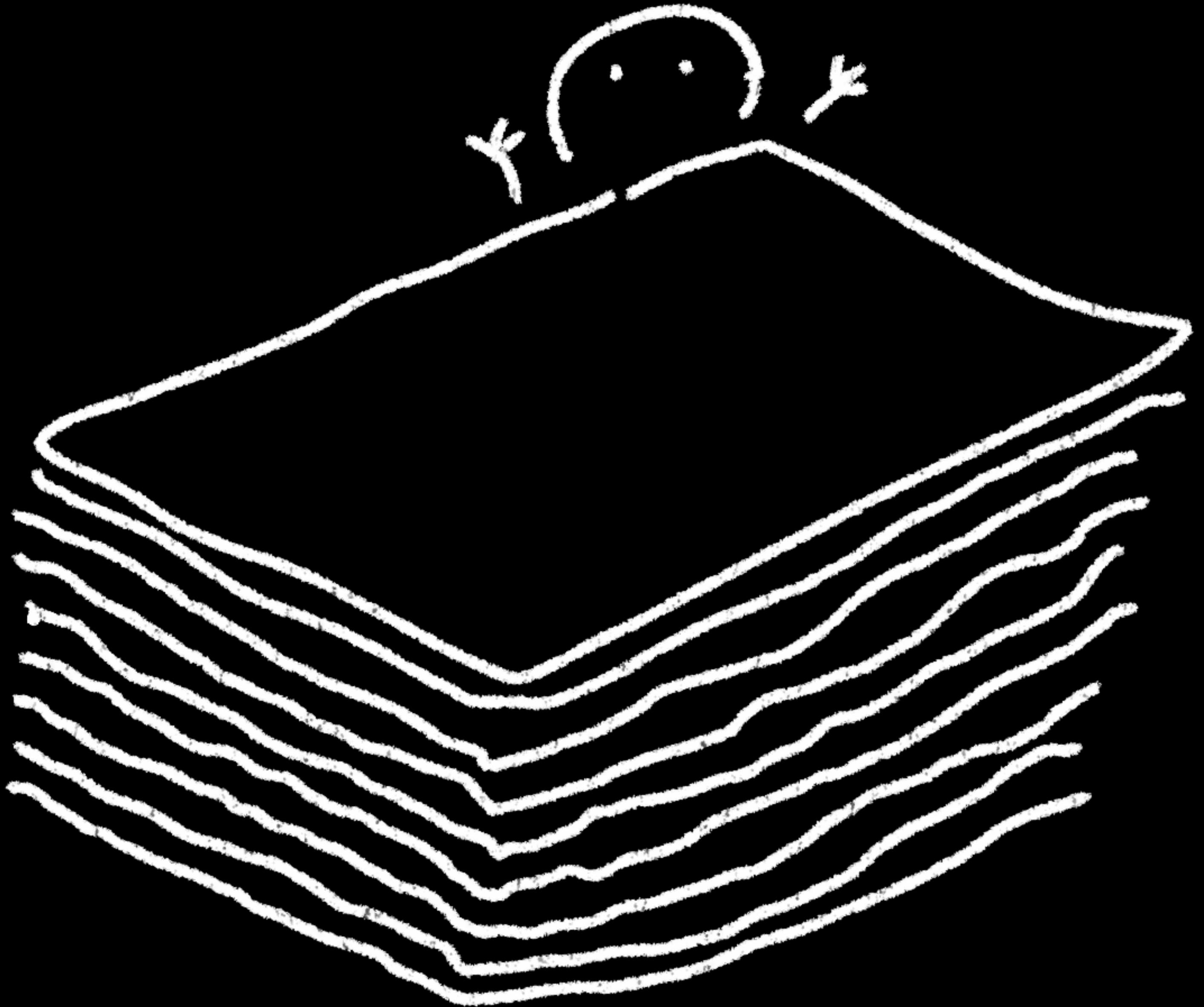
“oh yes, that
build has been
broken for a
few weeks...”

how to brick a spaceprobe

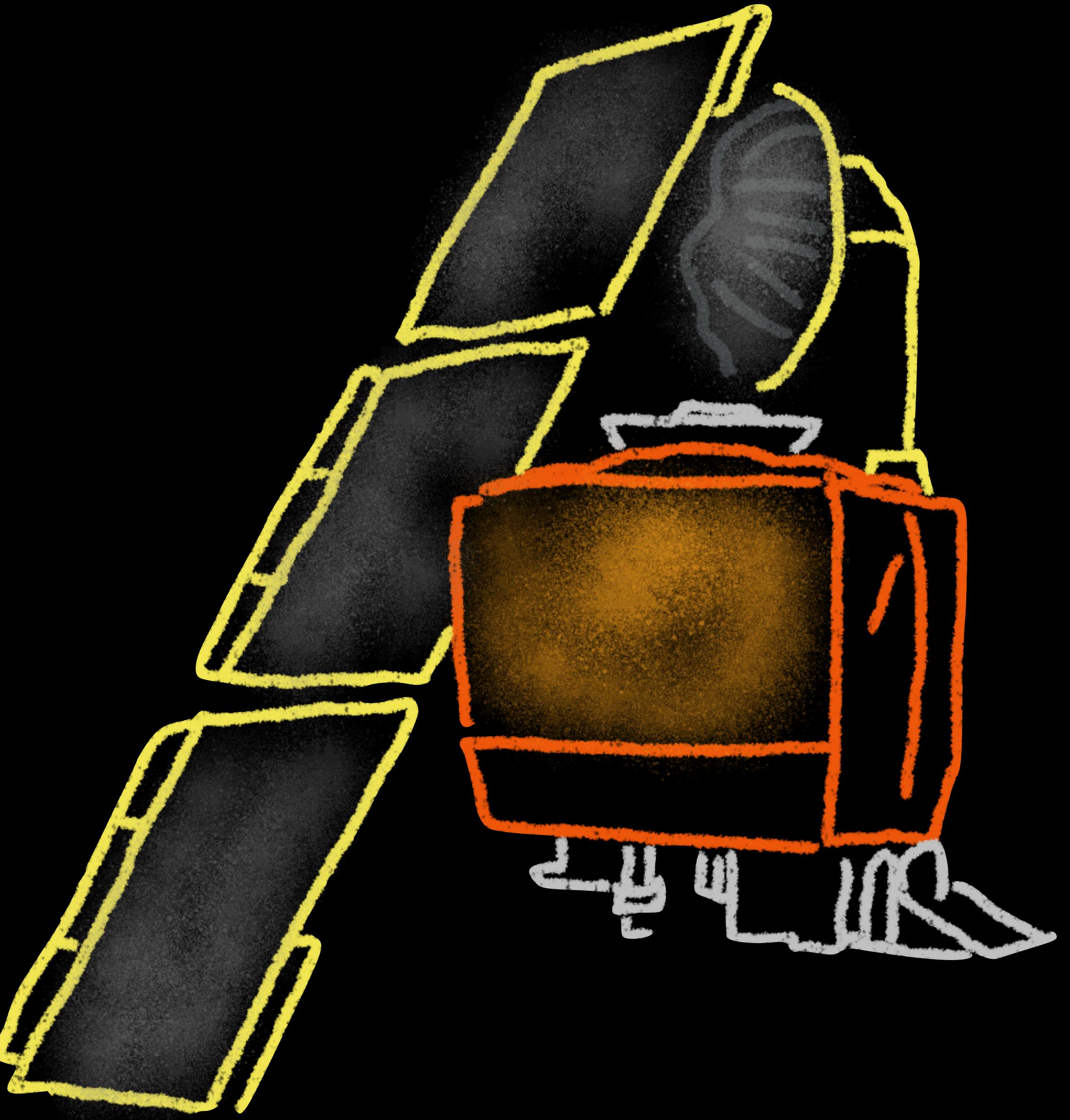


“we couldn’t get the
automated checks to work,
so we bypassed them”





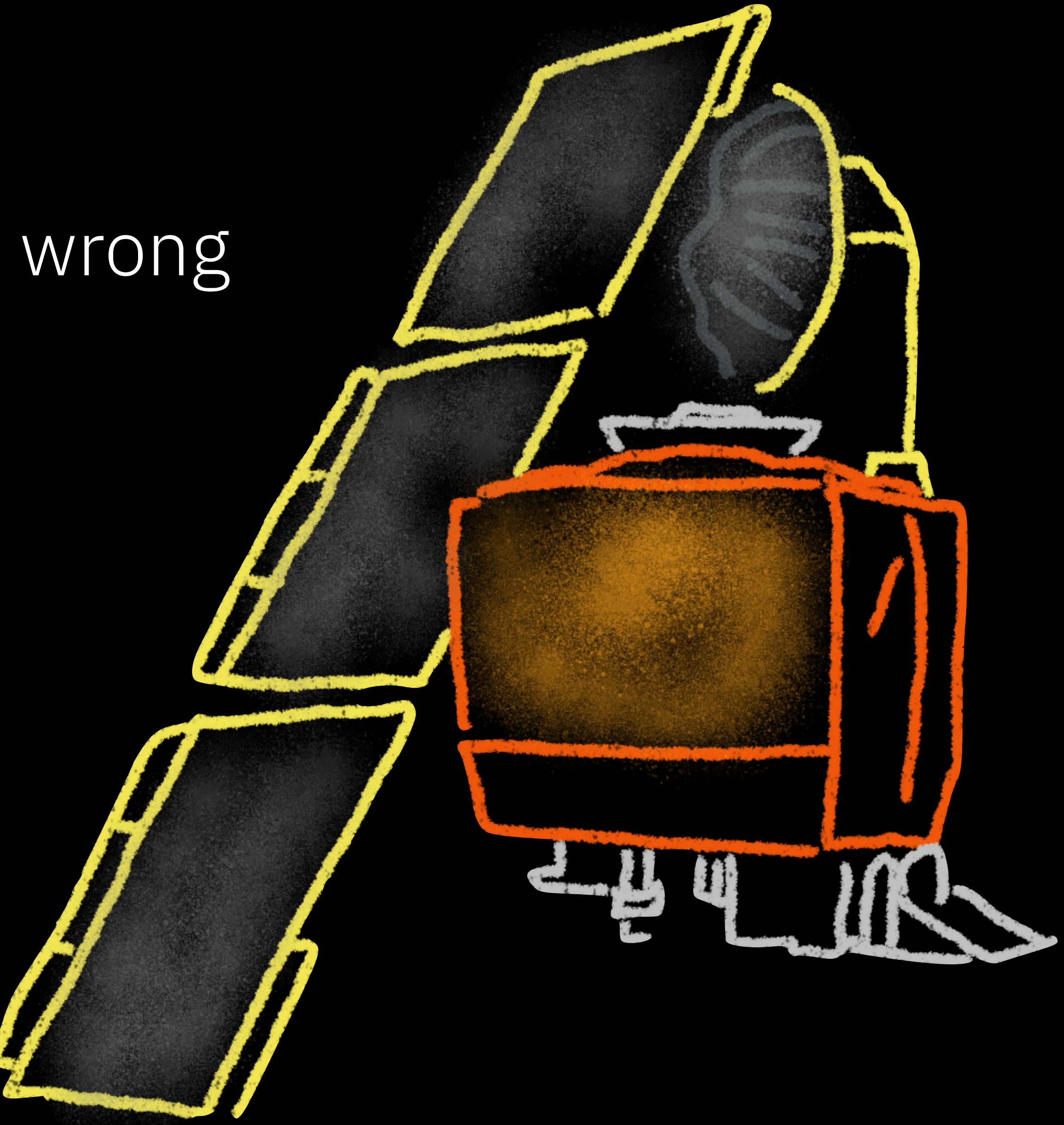
“we’ve
scheduled the
architecture
board review for
a month after the
project ships”



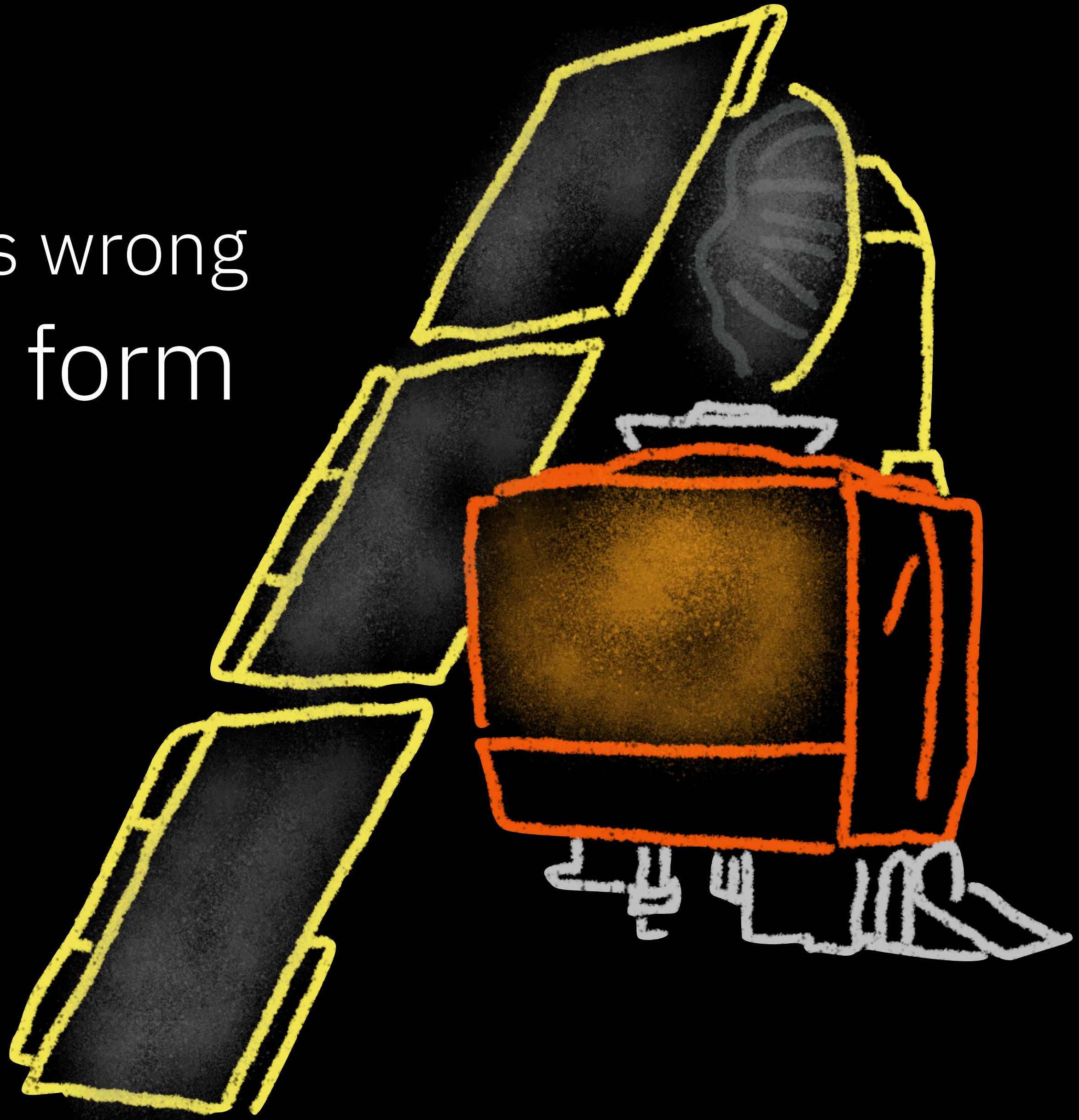
#IBMCLOUDGarage

@holly_cummins

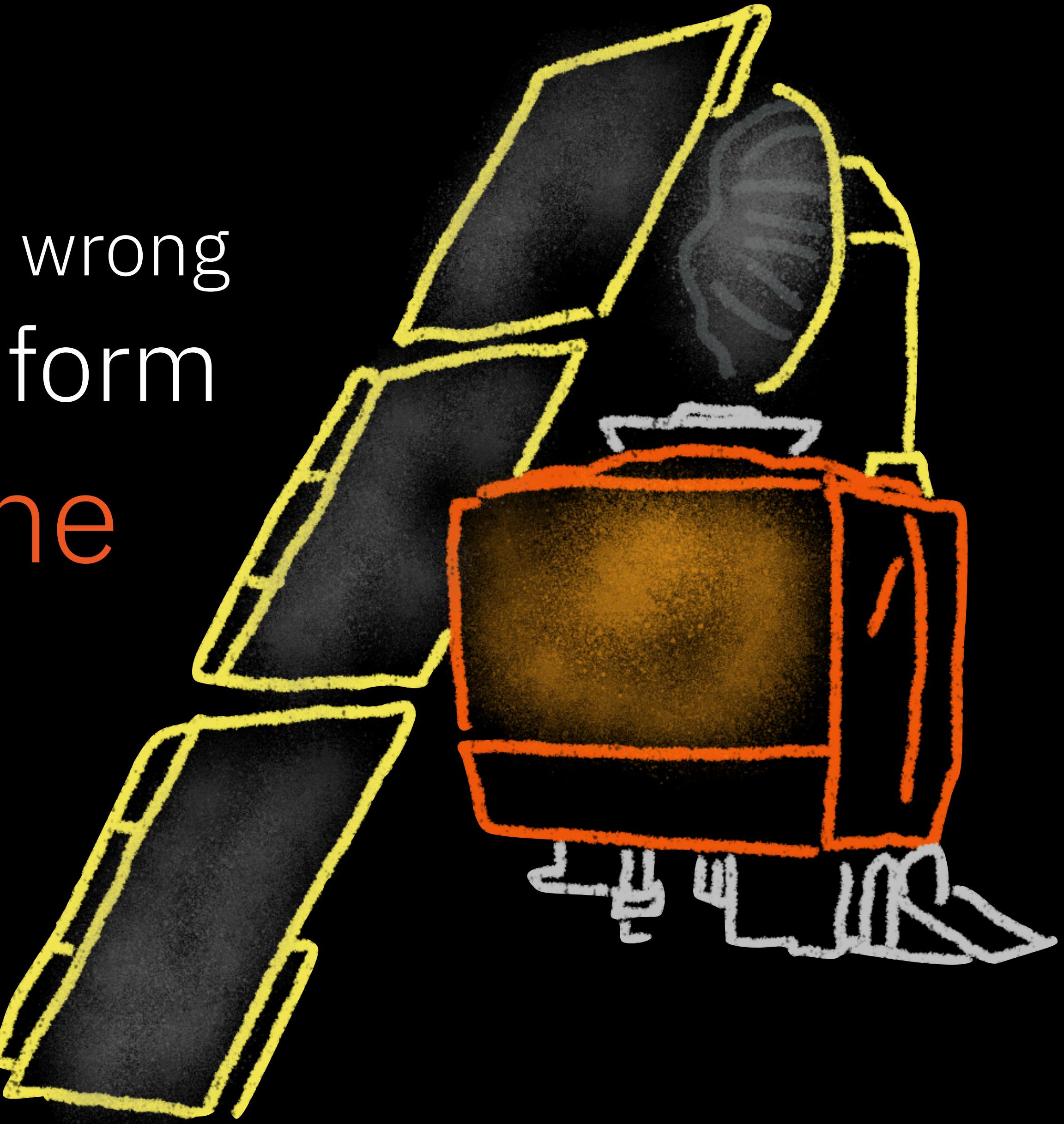
navigators warned something was wrong



navigators warned something was wrong
they didn't fill in the right form



navigators warned something was wrong
they didn't fill in the right form
so nothing was done



does the process add
value?

extreme programming is
the right kind of rigour

extreme programming is
the right kind of rigour

test-driven development

extreme programming is
the right kind of rigour

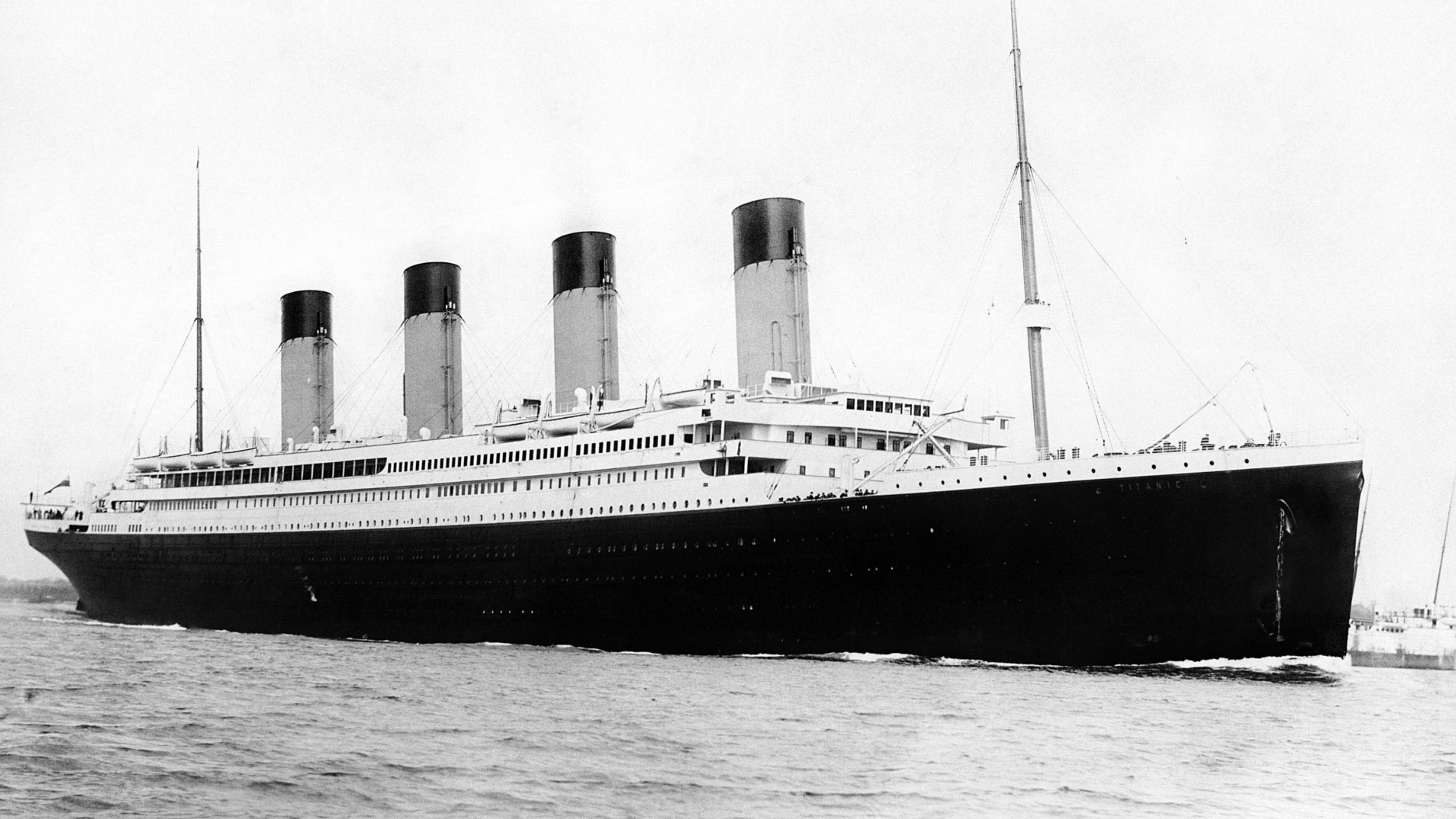
test-driven development
pair programming

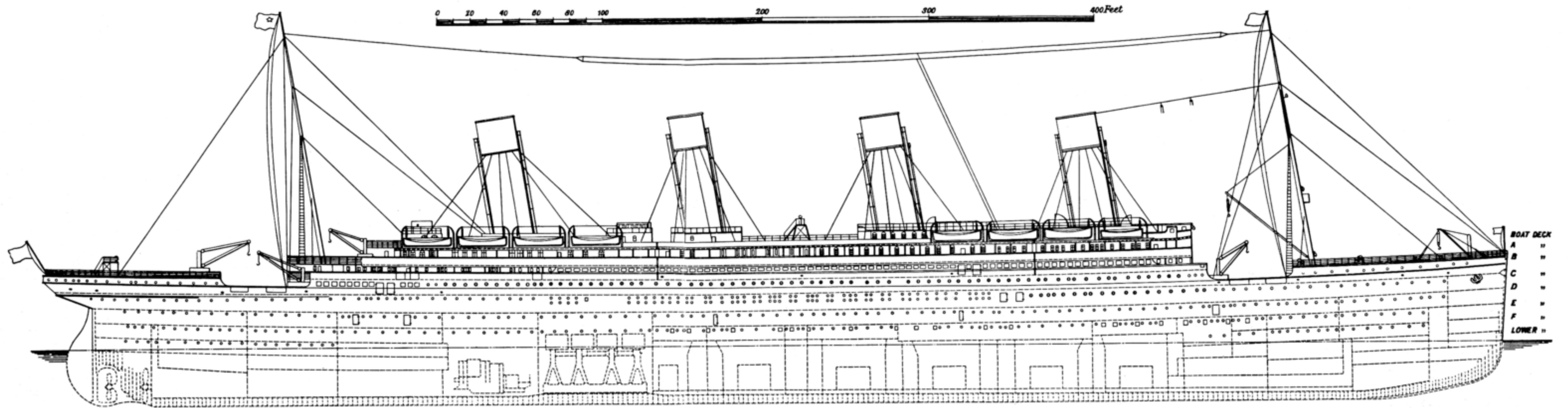
extreme programming is
the right kind of rigour

test-driven development
pair programming
optimise for feedback

“but it’s in the plan”

“but it’s not in the plan”





lots of bulkheads

it was too big

lookouts **saw** the iceberg
but the ship wasn't nimble enough to avoid it

BASED ON THE UNTOLD TRUE STORY

HIDDEN FIGURES



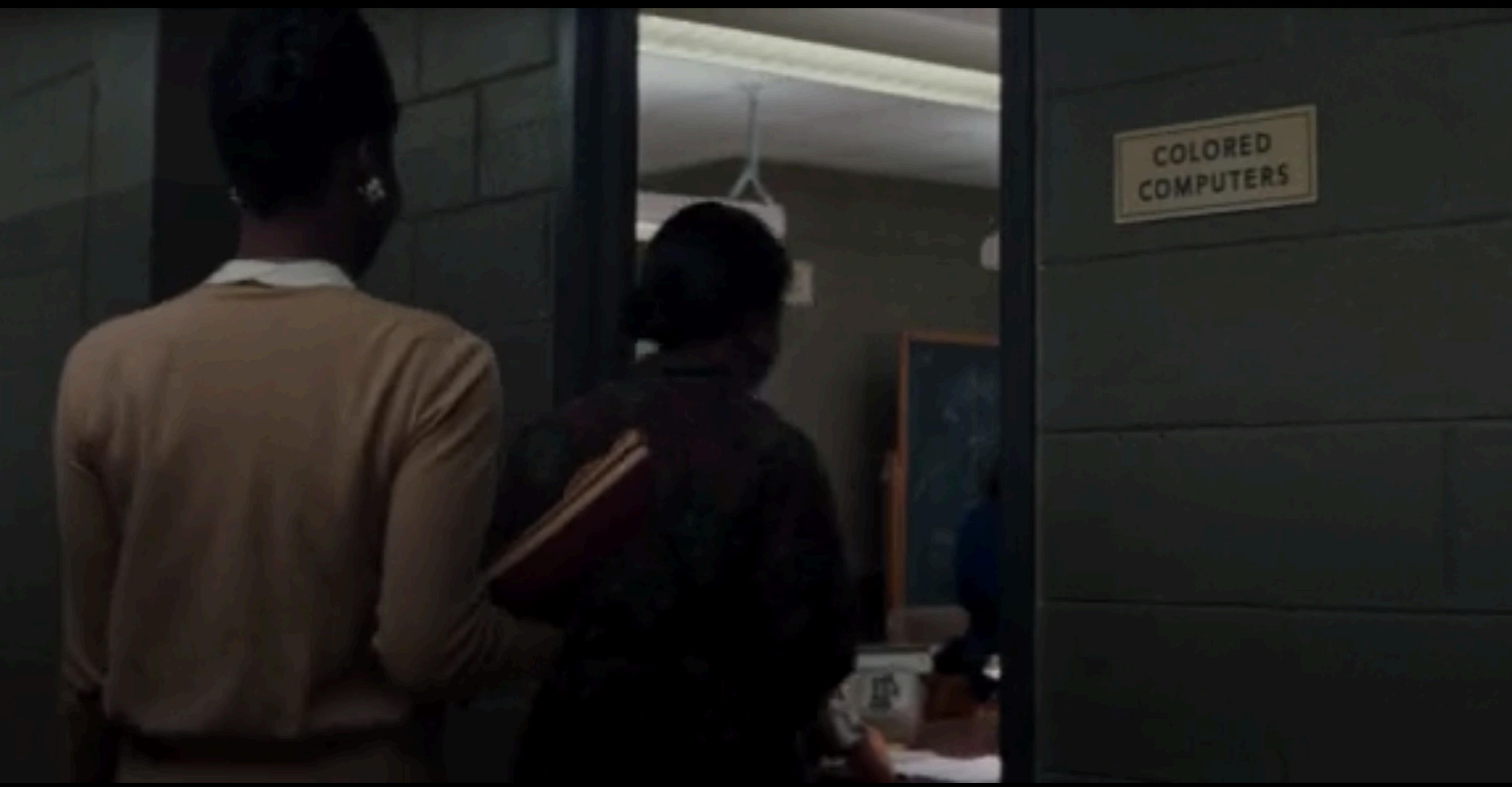
#IBMCloudGarage

@holly_cummings



#IBMCloudGarage

@holly_cummins



#IBMCloudGarage

@holly_cummins



#IBMCLOUDGarage

@holly_cummins



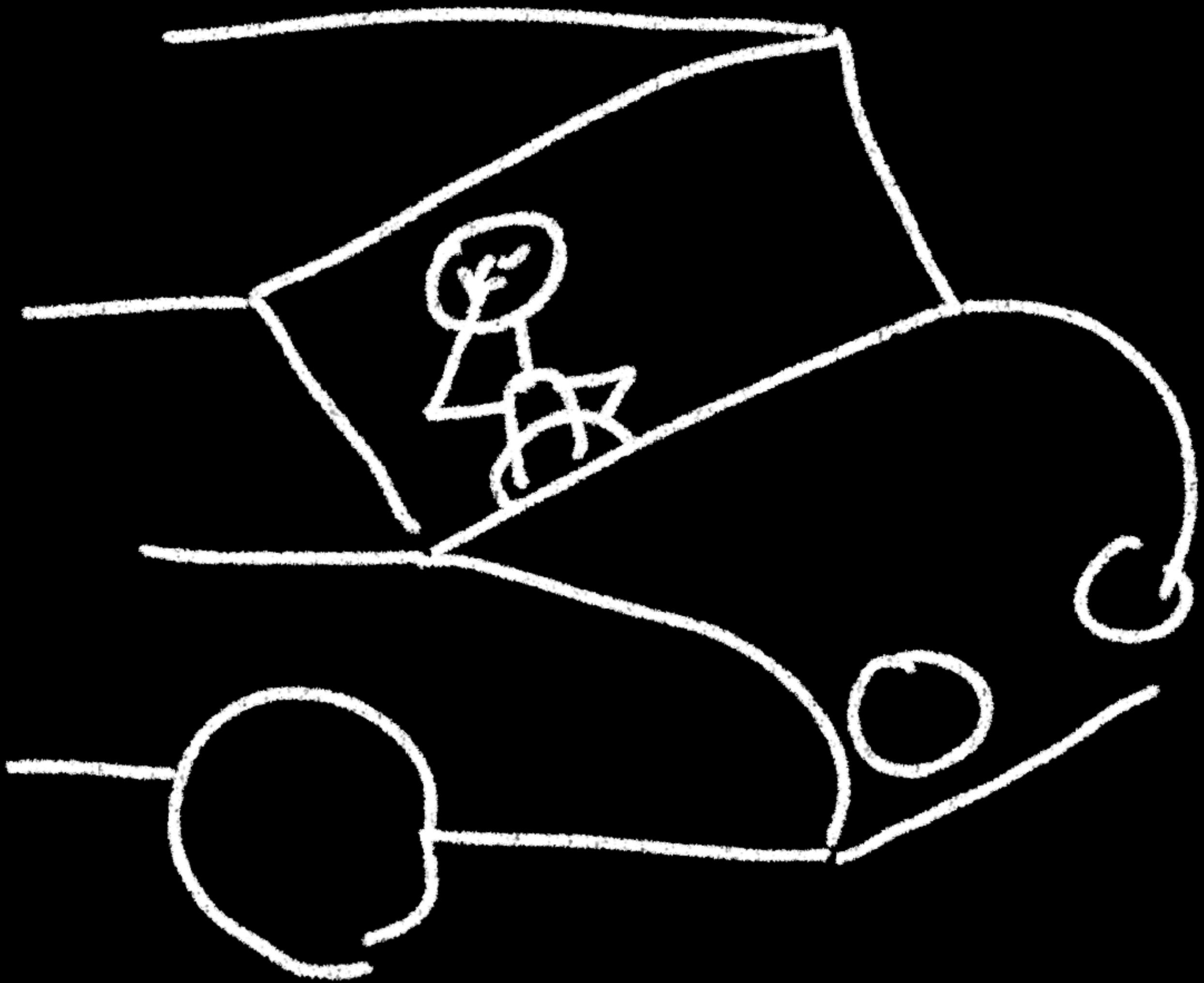
#IBMCLOUDGarage

@holly_cummins

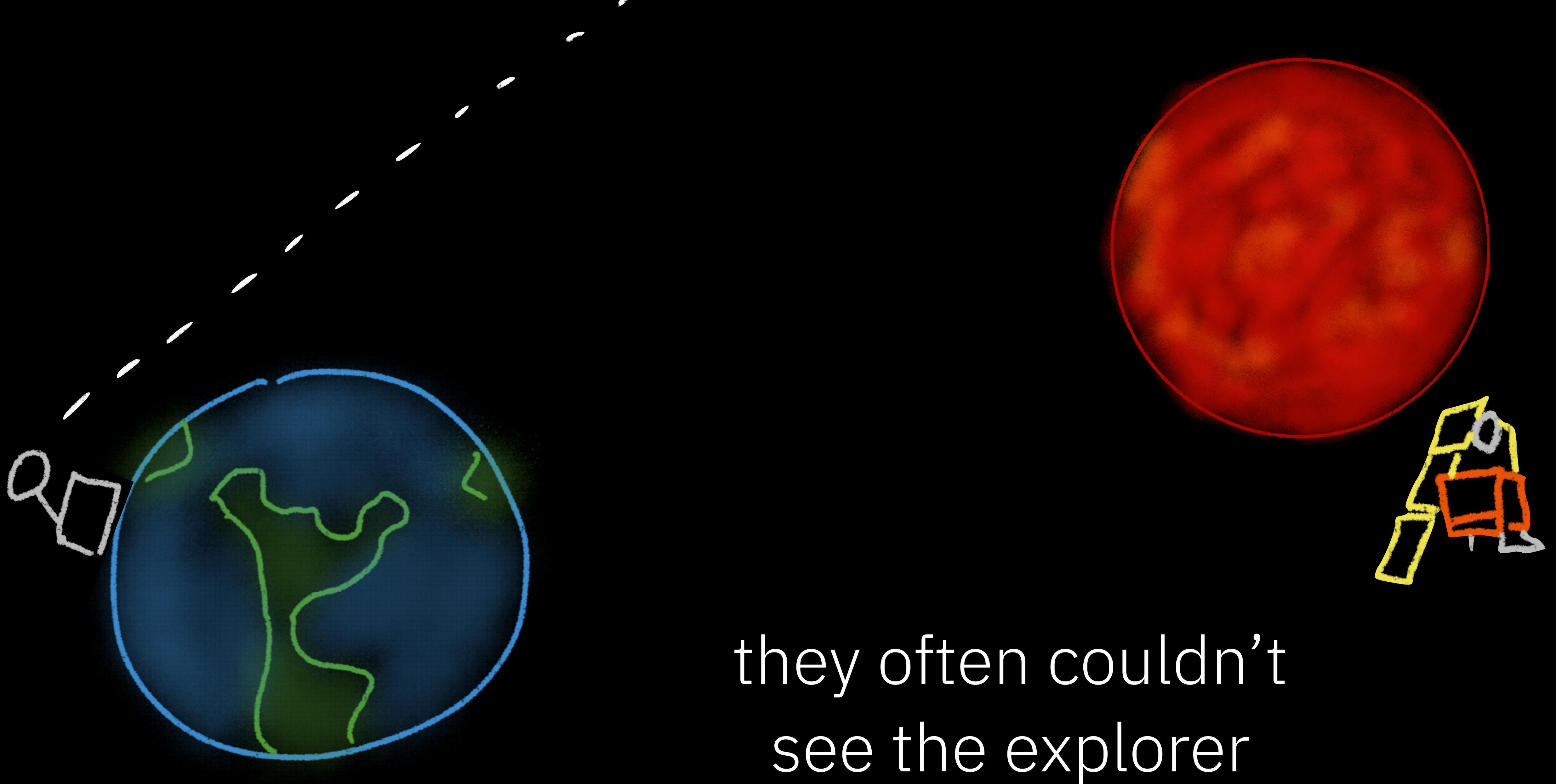


plans are **always** wrong
success is in how you respond

“we can’t ship until
every feature is
complete”



how **not** to
drive a car



they often couldn't
see the explorer

feedback is good business

feedback is good engineering

an mvp hurts

if you're not embarrassed
by your first release it
was too late

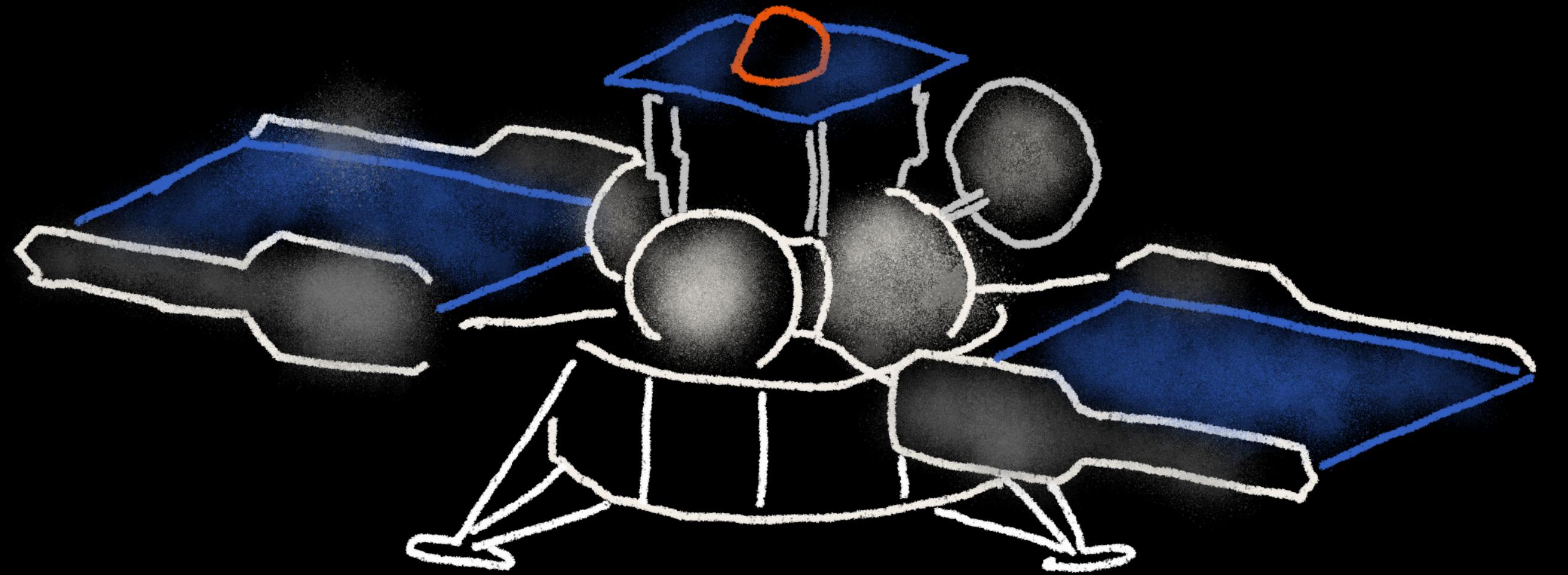
experiments can mean failure

experiments mean failure

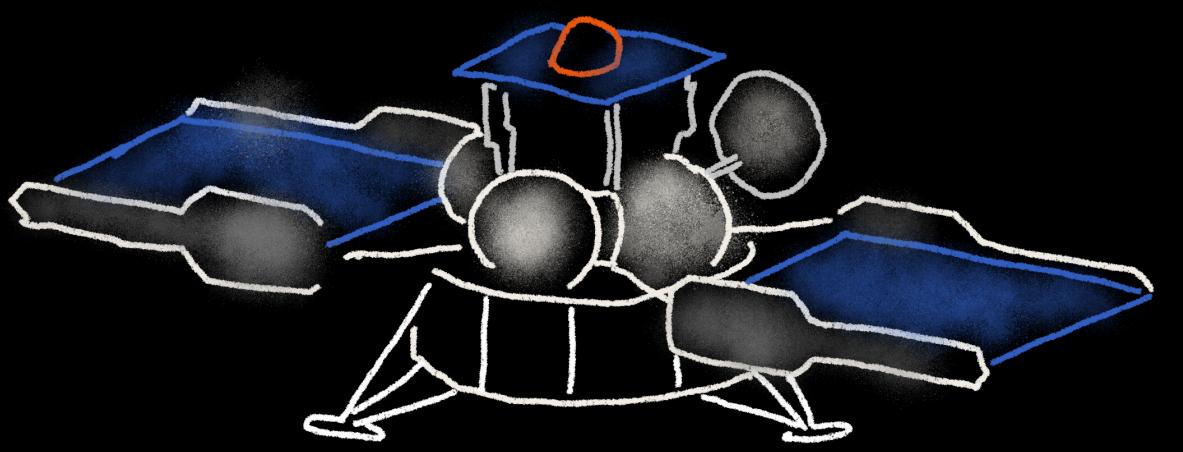


users will have
weird behaviours

optimise for recovery



unrecoverable



bricked

manual
intervention

handoffs

fast, but
data lost



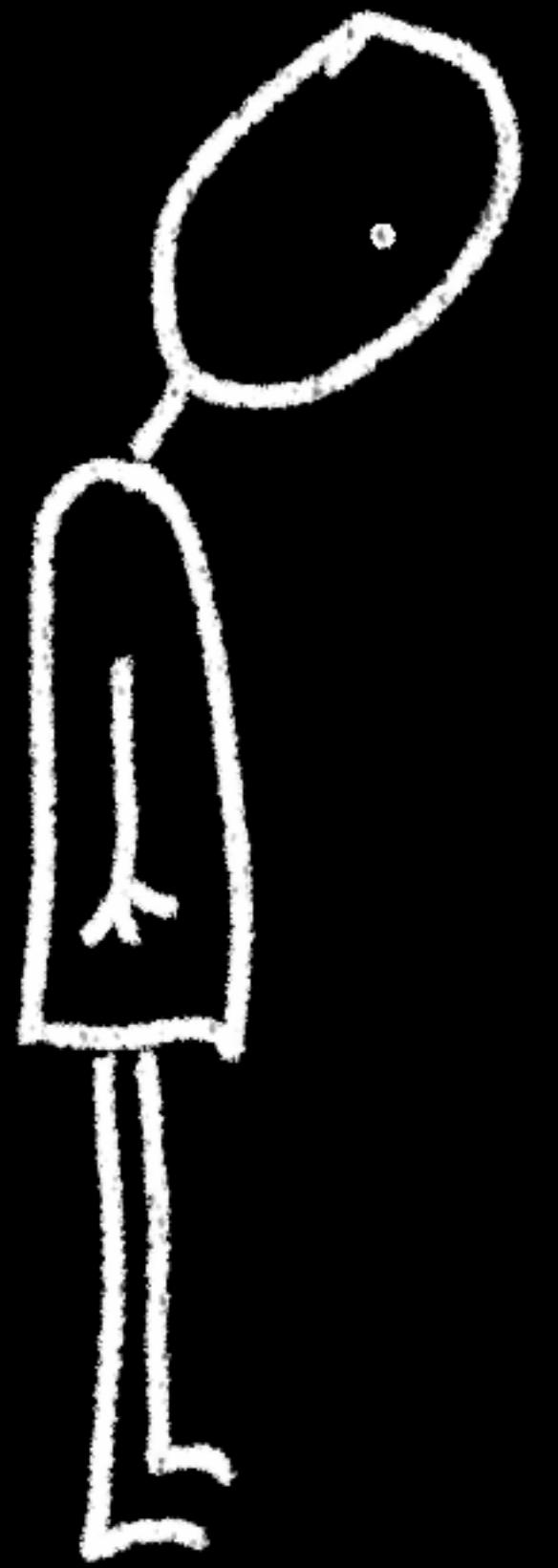
back in ms
no data loss

business recoverability

remember, users will
have **weird** behaviours



speed



slow is
demoralising
for teams

fast is
good
business



A late change in requirements
is a competitive advantage.

–Mary Poppendieck



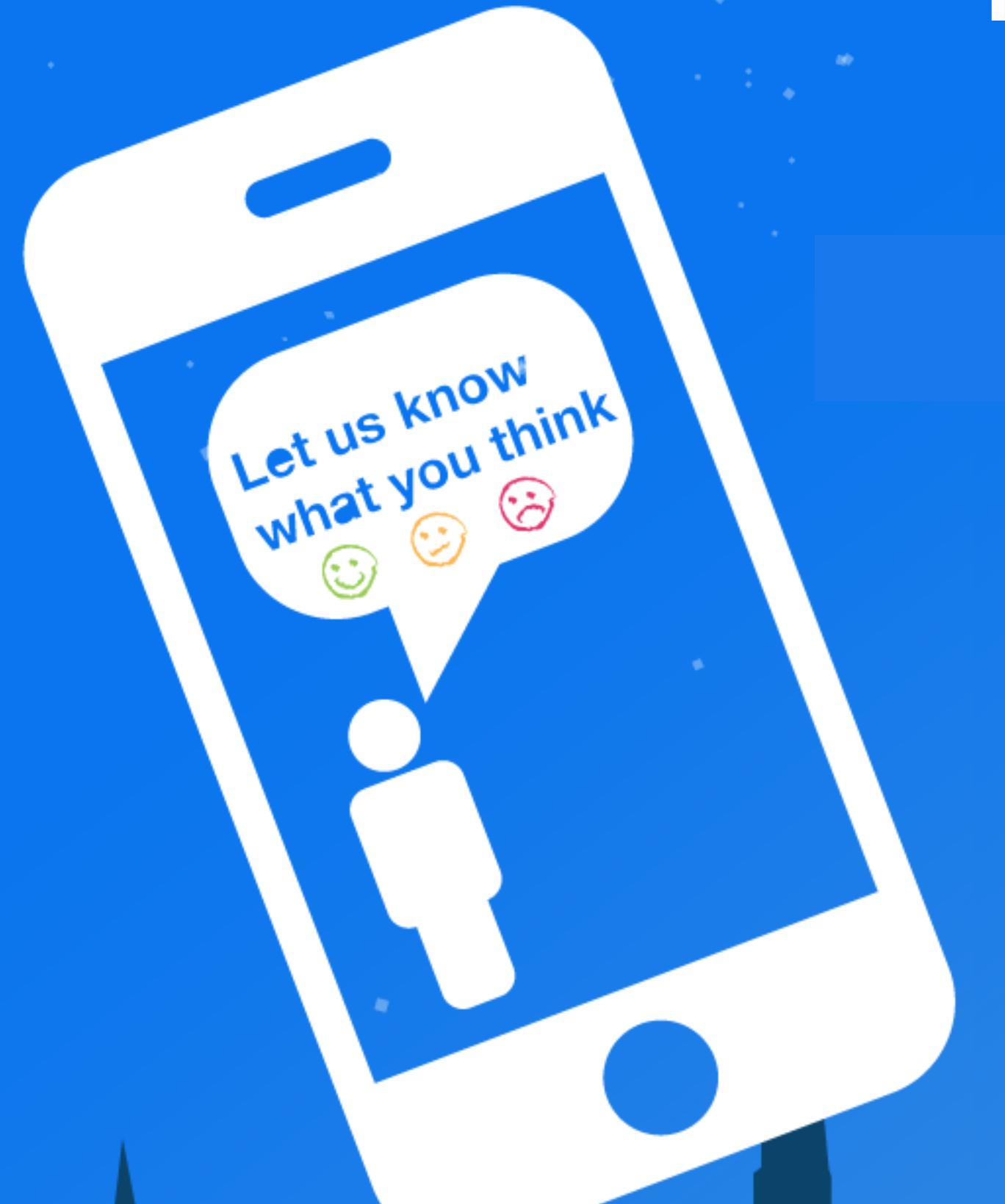
more feedback → more accuracy



cloud rescued
developers from tedium

cloud native
should feel
fun





please remember to
rate this session
thank you

@holly_cummins
#IBMCLOUDGarage

