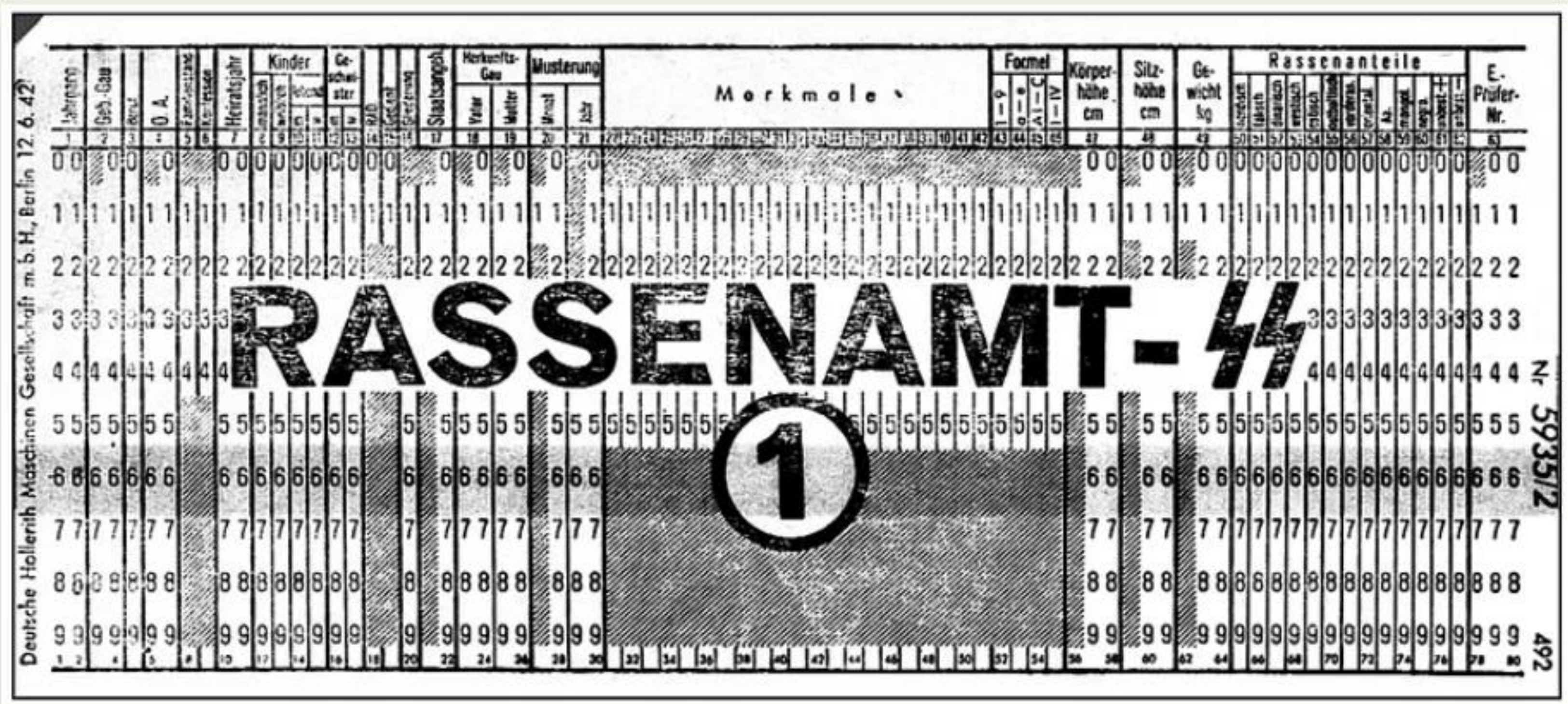
Yonatan Zunger • יונתן צונגר

What does AI Safety & Security Mean?

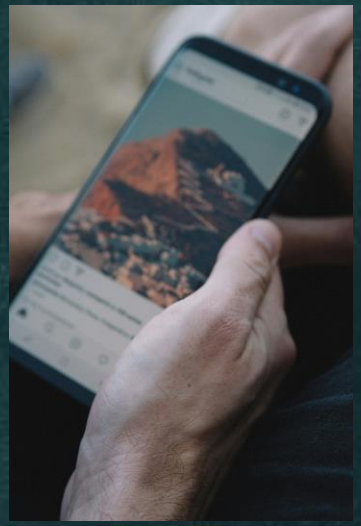
The history of CS is a history of rapid innovation.


RASSENAMT-SS
1
Merkmale
Formel
Körperhöhe cm
Sitzhöhe cm
Gewicht kg
Rassenanteile
E.-Prüfer-Nr.
Kinder
Herkunfts-Gau
Musterung
Staatsangeh.
Heiratsjahr
Jahrgang
Geb.-Gau
O. A.
Nr. 5935/2
492
Deutsche Hollerith Maschinen Gesellschaft m. b. H., Berlin 12. 6. 42

We tried to split up our profession

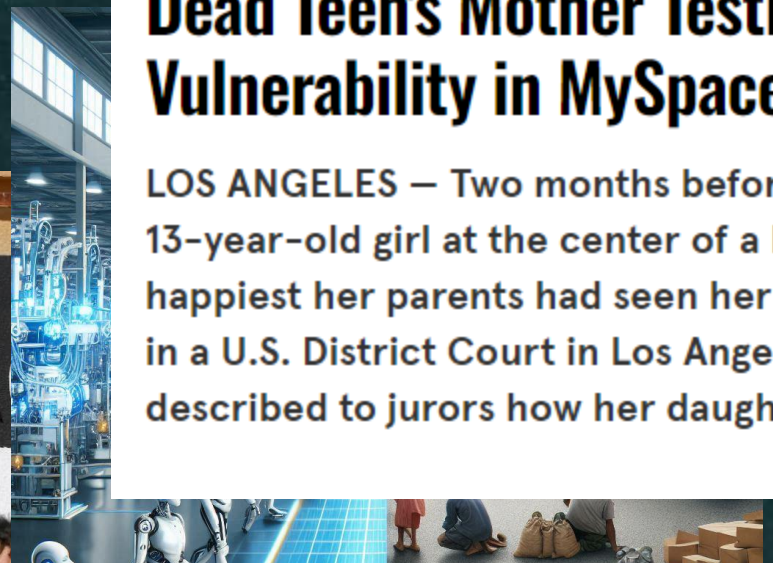
A few “serious” bits

The “exciting” stuff



And overall, we succeeded…

Sorta.



KIM ZETTER SECURITY NOV 20, 2008 6:09 AM

Dead Teen's Mother Testifies about Daughter's Vulnerability in MySpace Suicide Case -- Update

LOS ANGELES — Two months before she committed suicide in 2006, a 13-year-old girl at the center of a landmark cyberbullying case was the happiest her parents had seen her in a long time. Tina Meier, testifying in a U.S. District Court in Los Angeles on Wednesday afternoon, described to jurors how her daughter Megan [...]

LOCAL CRIME & PUBLIC SAFETY

D.C. 911 center under fire again after baby dies during computer outage

It is unclear whether delays in delivering advanced medical care contributed to the five-month-old's death.


What we do matters.
And with great power comes great responsibility.

All of engineering has two sides:

Product Engineering:

How the system should work

Safety Engineering:

How the system should fail

Most engineering disciplines don’t treat these as separate.

Neither should we.

So today --

- I’ll show you the three basic principles of safety engineering
- I’ll give a concrete example of what an AI component failure looks like
- And then we’ll walk through a full example of how to apply this to a real system.


Three Principles

First Principle of Safety Engineering

You are building systems, not software.
Everything is in scope.

"Your system" means the entire business process, including the people.

"Failures" means anything that might require your response.

Why? Because failures and fixes usually span component boundaries.

Why? Because if the human makes a mistake, the headline will still be about your AI system.

Why? Because engineers build systems to solve problems, and if any part of the system fails, you didn't solve the problem.

“User Error” is not an excuse.

- If it’s one-off and catastrophic, how was it so easy to make that mistake?
- If it’s routine, what aspect of the system was leading to it?
- Why wasn’t the system robust to it?

The human is a component of your system and the system must be robust to component failure.

Every system is a sociotechnical system.

Second Principle of Safety Engineering

Know what can go wrong in your system, and for each of those things, have a plan.

- “Know what can go wrong” means a multi-pronged analysis: system-first, attacker-first, target-first.
- With failure, what you don’t know **can** hurt you – so get many eyes on it.
- “A plan” can mean anything from a system change, to an incident response plan, to better comms.
- “Unknown unknowns” are one of the things that can go wrong.

Third Principle of Safety Engineering

You start thinking about this the day you get the idea for the project, and you do it continuously until the day it's shut down.

Planning for failure isn't an exercise; it's the parallel to feature design.

You update your vision for the features all the time.

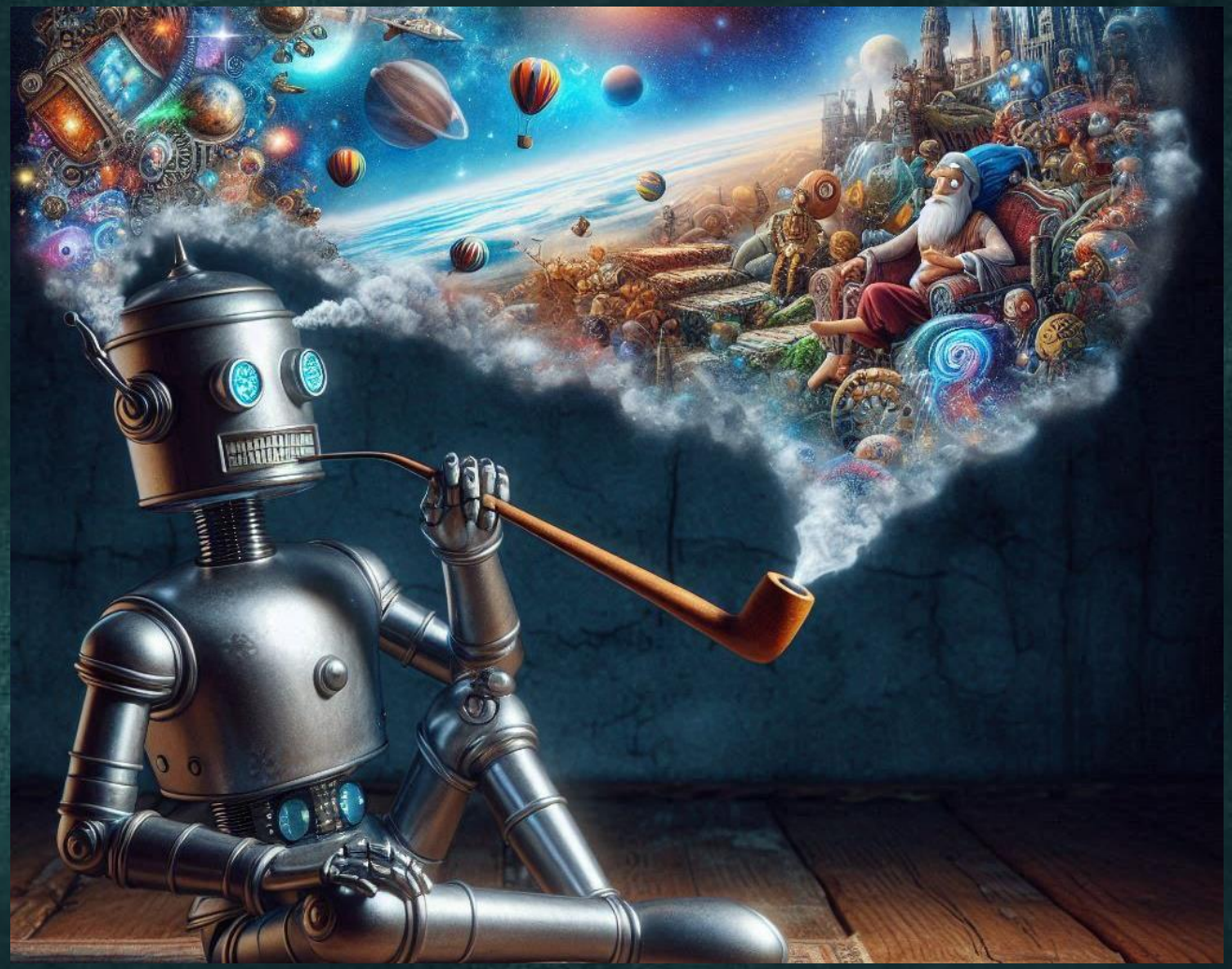
Update your vision of the failures just as often.

Use this by building a safety plan

- List the things that can go wrong; group similar failures.
- Look at the “failure chain” for each of them and find points of intervention.
- Look for common points of intervention to fix multiple issues.
- Write your plan: Failures, solutions, and a matrix to make sure each failure is covered.

OK, so where’s the AI in all this?

Let’s talk about hallucinations.

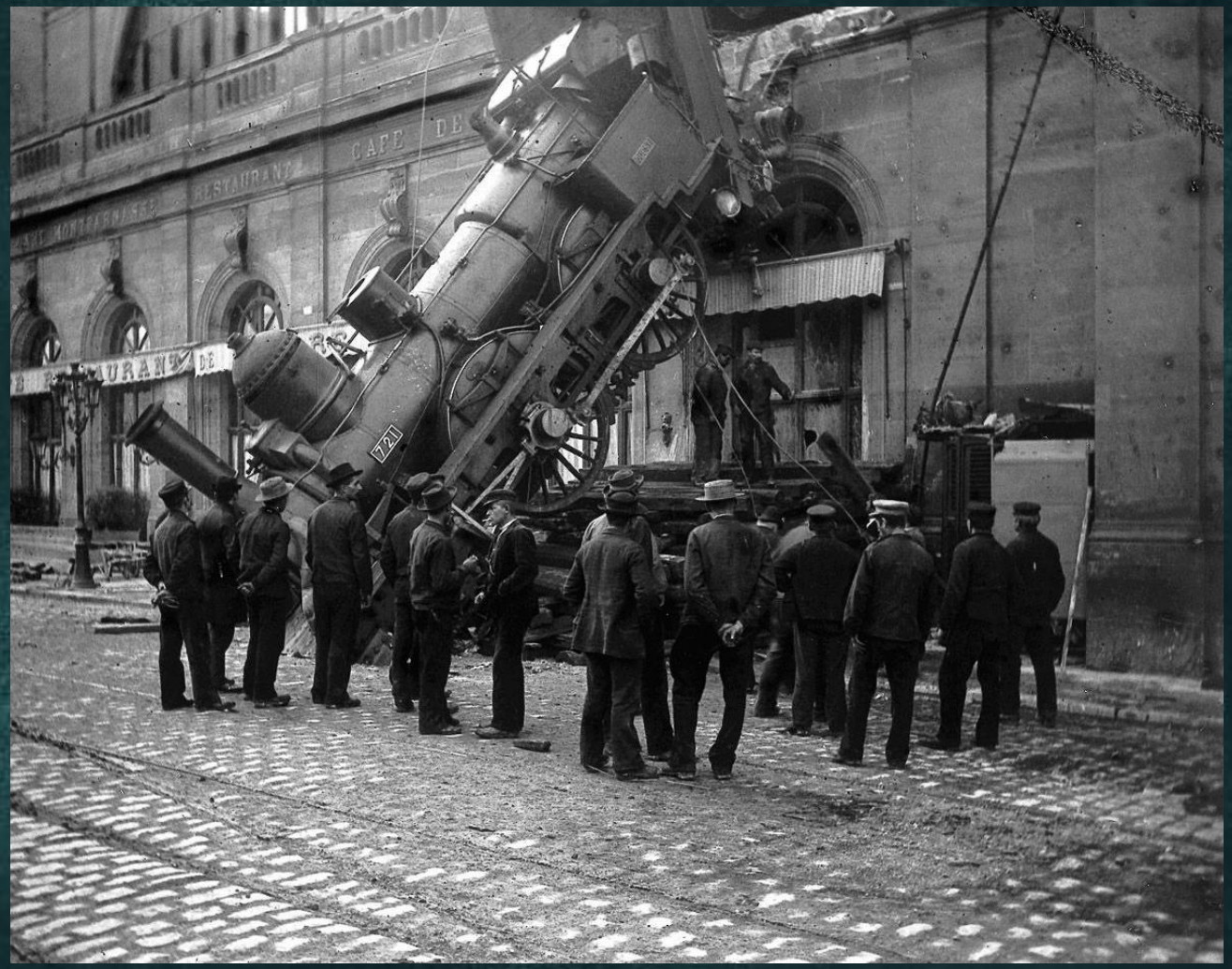


Error is inherent to AI.

Generative AI's can err in five basic ways:

- Hallucination
- Omission
- Misinterpreting data
- GIGO
- Unexpected preferences

You can trade off compute for error rate, but you can never eliminate it.



So what do we do about it?

Instead of “hallucination,” think “overreliance.”

Humans err, too. Problems happen when you rely on them inappropriately.

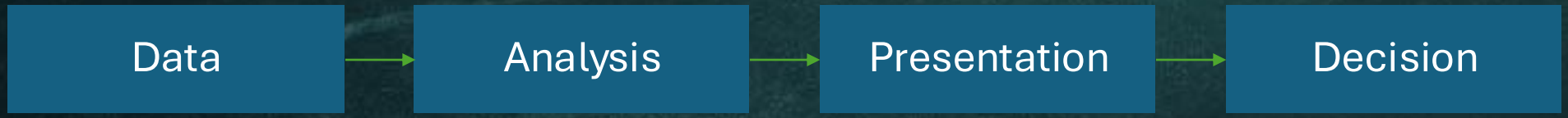
Don’t think of AI like “the computer” knowing the right answer;
Instead think of it like a new hire straight out of school.

Good at: Brainstorming, summarizing, tasks that you can verify.
Bad at: Deterministically processing data.

Enough theory!
I’m an engineer, show me how it works.

Imagine an app to help bank loan officers.

- It gathers all the requisite data, summarizes it, and then a human makes the decision.

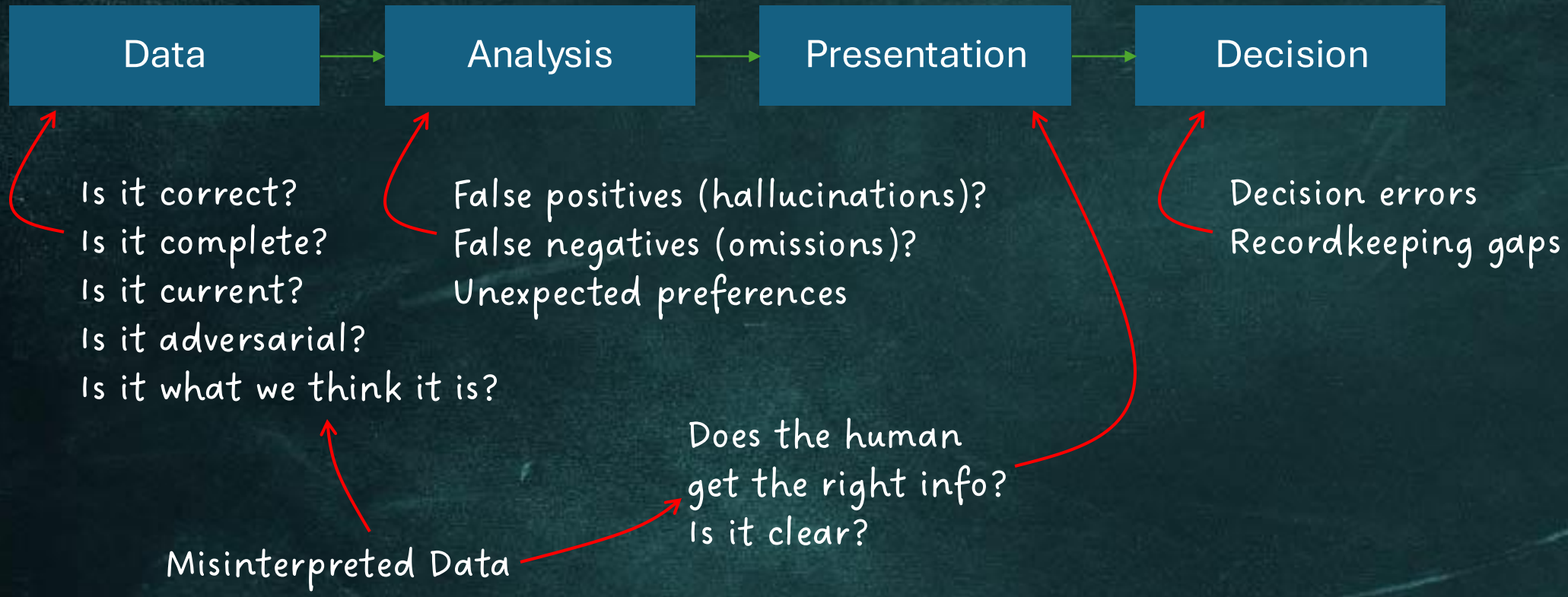


AI goes here

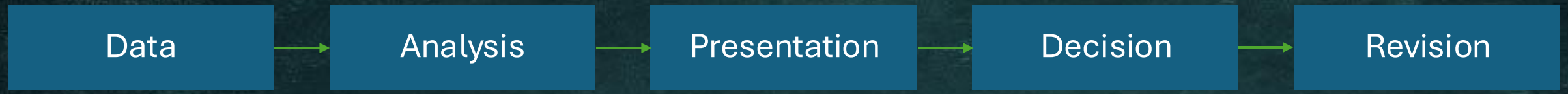
Look, a human is in the loop!
That makes the AI safe, right?

[Cue hysterical laughter]

What could possibly go wrong?



What should we do about it?



Is it correct?
Is it complete?
Is it current?
Is it adversarial?
Is it what we think it is?

False positives (hallucinations)?
False negatives (omissions)?
Unexpected preferences

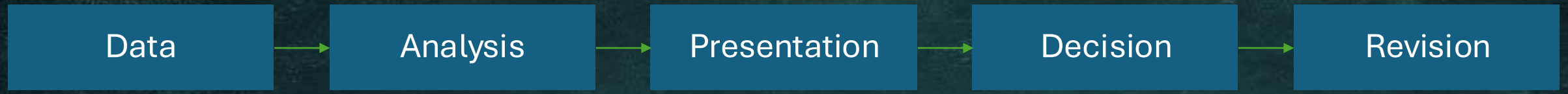
Decision errors
Recordkeeping gaps

No, it isn't. But this is an existing issue not aggravated by AI.

Does the human get the right info?
Is it clear?

Mitigate it with robust data-gathering and a good revision flow.

What should we do about it?



Is it correct?
Is it complete?
Is it current?
Is it adversarial?
Is it what we think it is?

False positives (hallucinations)?
False negatives (omissions)?
Unexpected preferences

Decision errors
Recordkeeping gaps

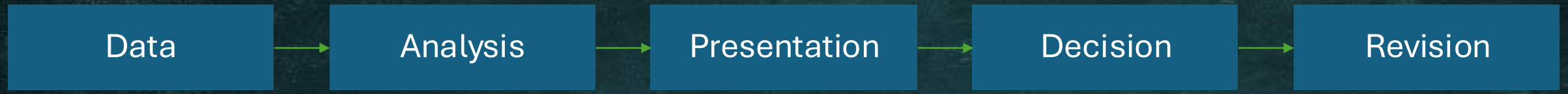
No, it isn't. But this is an existing issue not aggravated by AI.

Does the human get the right info?
Is it clear?

Mitigate it with robust data-gathering and a good revision flow.

The presentation issues will apply here, too.

What should we do about it?



Is it correct?
Is it complete?
Is it current?
Is it adversarial?
Is it what we think it is?

No, it isn't. But this is an existing issue not aggravated by AI.

False positives (hallucinations)?
False negatives (omissions)?
Unexpected preferences

Does the human get the right info?
Is it clear?

Decision errors
Recordkeeping gaps

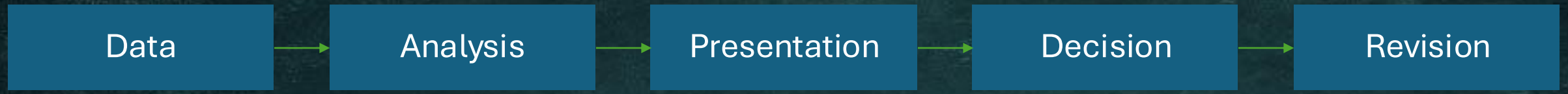
What does the user need to know?

- What changed?
- What factors set the original decision?

Mitigate it with robust data-gathering and a good revision flow.

The presentation issues will apply here, too.

What should we do about it?



Is it correct?
Is it complete?
Is it current?
Is it adversarial?
Is it what we think it is?

False positives (hallucinations)?
False negatives (omissions)?
Unexpected preferences

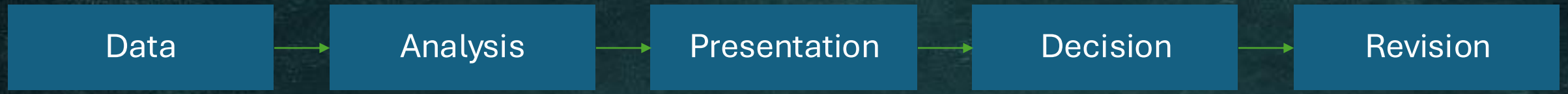
Decision errors
Recordkeeping gaps

Does the human get the right info?
Is it clear?

Harden against XPIA: Treat all external data as hostile until proven safe.

And test very hard for this!

What should we do about it?



Is it correct?
Is it complete?
Is it current?
Is it adversarial?
Is it what we think it is?

False positives (hallucinations)?
False negatives (omissions)?
Unexpected preferences

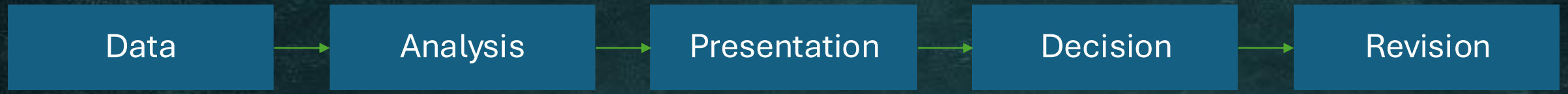
Decision errors
Recordkeeping gaps

Does the human
get the right info?
Is it clear?

Think about each datum we *want* and compare it to what we *have*.

Where do they differ?

What should we do about it?



Is it correct?
Is it complete?
Is it current?
Is it adversarial?
Is it what we think it is?

False positives (hallucinations)?
False negatives (omissions)?
Unexpected preferences

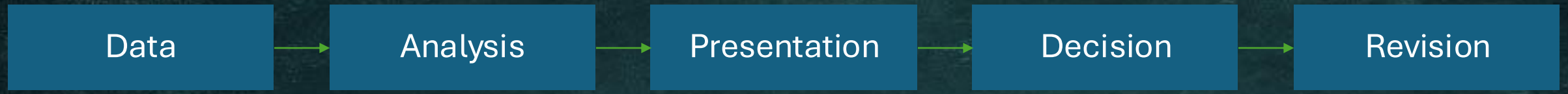
Decision errors
Recordkeeping gaps

Can harden the code some, but it's not 100%.

Does the human get the right info?
Is it clear?

Surface data sourcing in the UI, so if anything is unsourced or doesn't match the source, it's obvious to the user.

What should we do about it?



Is it correct?
Is it complete?
Is it current?
Is it adversarial?
Is it what we think it is?

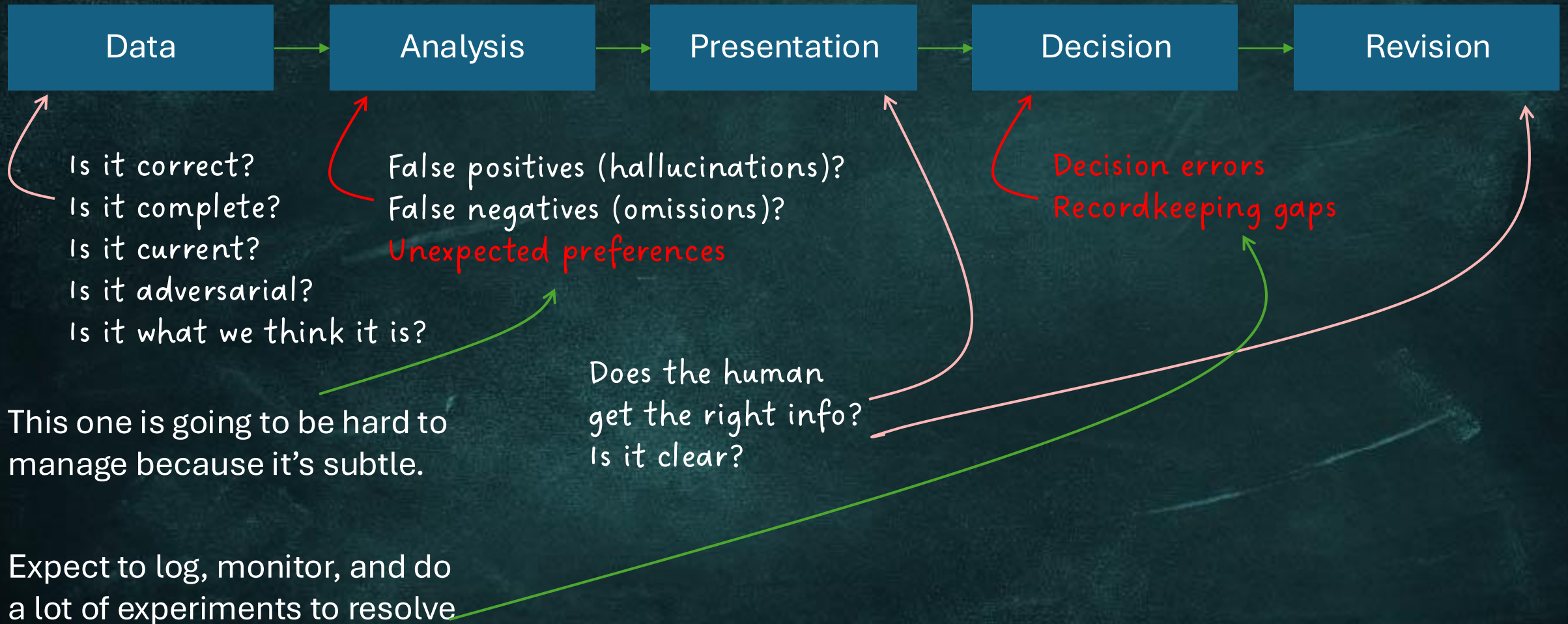
False positives (hallucinations)?
False negatives (omissions)?
Unexpected preferences

Decision errors
Recordkeeping gaps

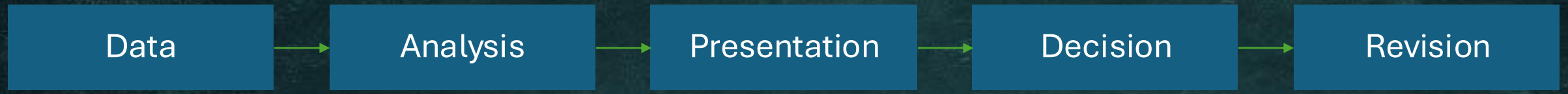
This one is going to be hard to manage because it's subtle.

Does the human get the right info?
Is it clear?

What should we do about it?



What should we do about it?



Is it correct?
Is it complete?
Is it current?
Is it adversarial?
Is it what we think it is?

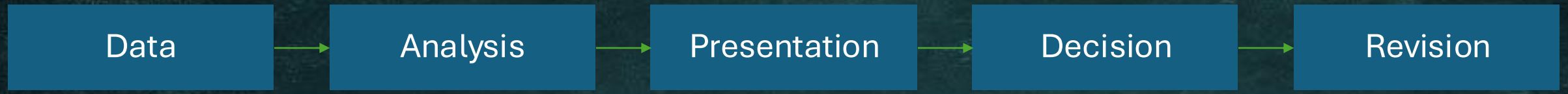
False positives (hallucinations)?
False negatives (omissions)?
Unexpected preferences

Decision errors
Recordkeeping gaps

Does the human
get the right info?
Is it clear?

Thou Shalt Not Skimp on Thy UX Design

What should we do about it?



Is it correct?
Is it complete?
Is it current?
Is it adversarial?
Is it what we think it is?

False positives (hallucinations)?
False negatives (omissions)?
Unexpected preferences

Does the human
get the right info?
Is it clear?

Decision errors
Recordkeeping gaps

Oh wait, you were trusting humans?
You poor fool.

Better cross-check.

Some percent goes to multiple humans in parallel.

Putting it all together, here's our safety plan:

- **Adversarial input** → Harden inputs against XPIA and test
- **Incorrect, incomplete, stale, misinterpreted input** → Good data collection, revision flow
- **Analysis error** → Presentation UX needs to make sourcing clear
- **Unexpected preference and misinterpretation** → Monitor to check for patterns indicating bias based on both loan officer and applicant; continuous research
- **Human error** → Clear presentation UX; log all decision factors; multiple humans cross-check a sample and priority items
- **The revision flow** → Show what factors affected the previous decision and what's changed

So how do we do safety?

- Design the business process, not just the software system
- Know what can go wrong, and have a plan for it
- Do this continuously
- Test and monitor, especially in nondeterministic systems
- Apply similar strategies to human and AI component failures
- Build a safety plan and write it down

Above all:

What you build matters.

It doesn’t have to be a medical or a military system to have serious impact.

Safety includes everything.

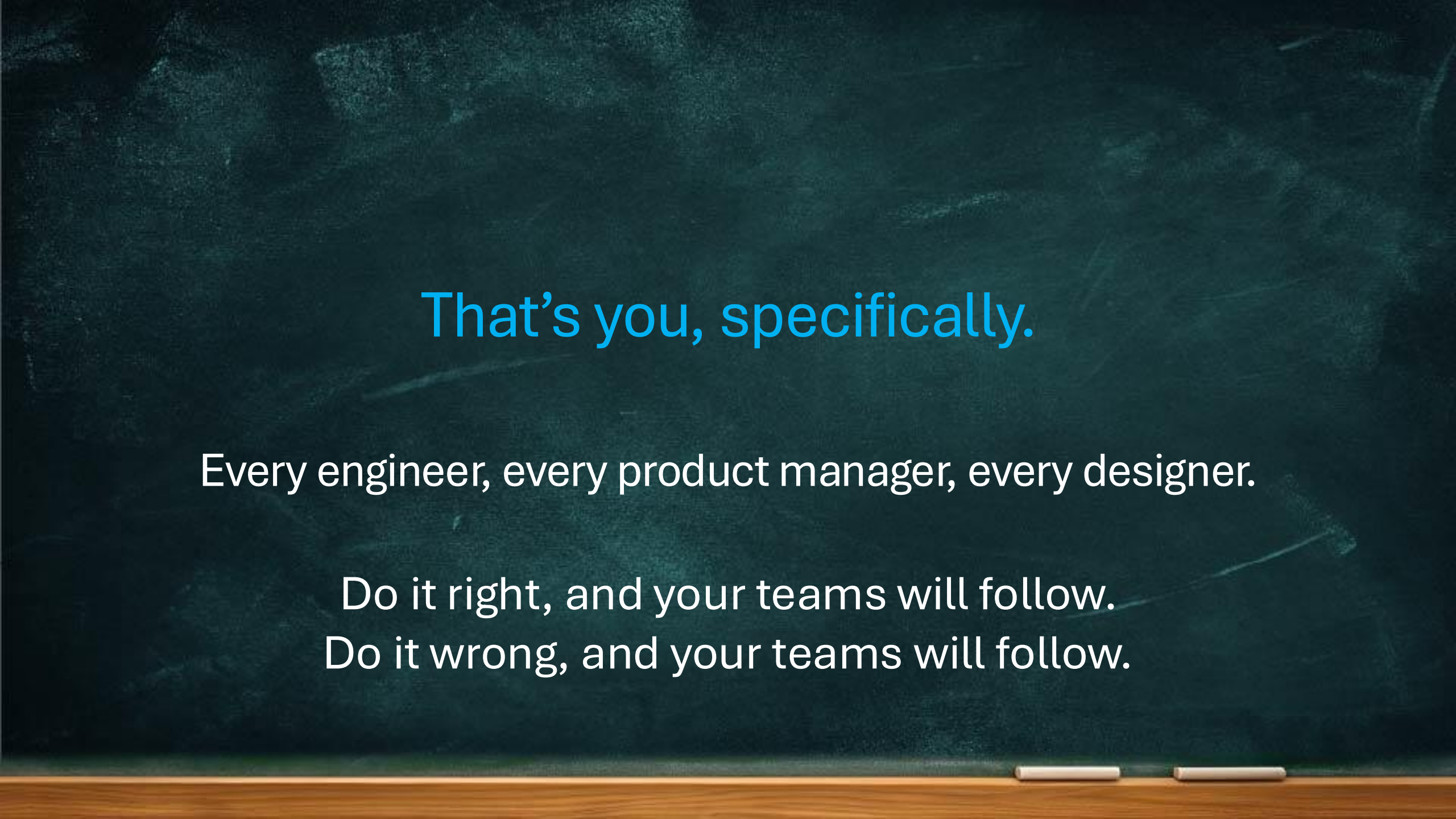
There ain’t no such thing as “out of scope.”

Just because someone used a system in a way we didn’t expect doesn’t mean it’s not our problem.

If it’s your system and someone or something can get hurt, you need a plan.

Know your system.

Model threats from the moment you imagine the system.
Know them as well as you know your features.


That’s you, specifically.
Every engineer, every product manager, every designer.
Do it right, and your teams will follow.
Do it wrong, and your teams will follow.