

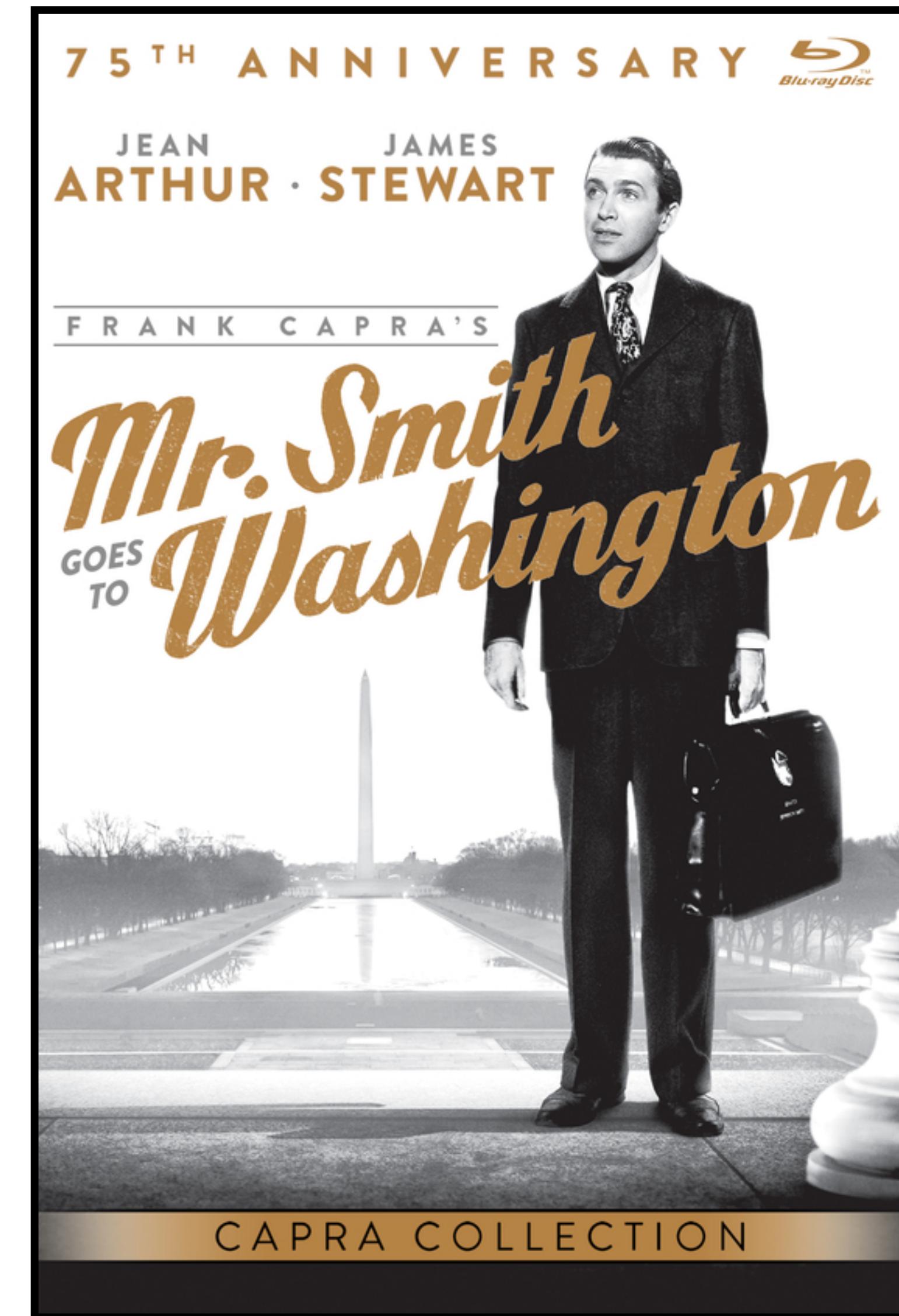
# Leadership Development

## Lessons Learned

Ian Eslick, August 22nd - Enterprise Tech Leadership Summit - Las Vegas, NV

# Inspiration for today

- How did an engineer find a way to thrive and grow into senior management in a traditional enterprise?
- What advice would have been helpful for me or has been helpful to those I've led or mentored?



# A little context...

- I've been working in Technology for 35 years
- Managing teams for 20+ years – 10 in startups and 10 in large enterprises
- Across diverse industries and roles
- Most recent role was CTO for Enterprise Engineering / Cloud at U.S. Bank
- Started at SoFi Technologies last week!

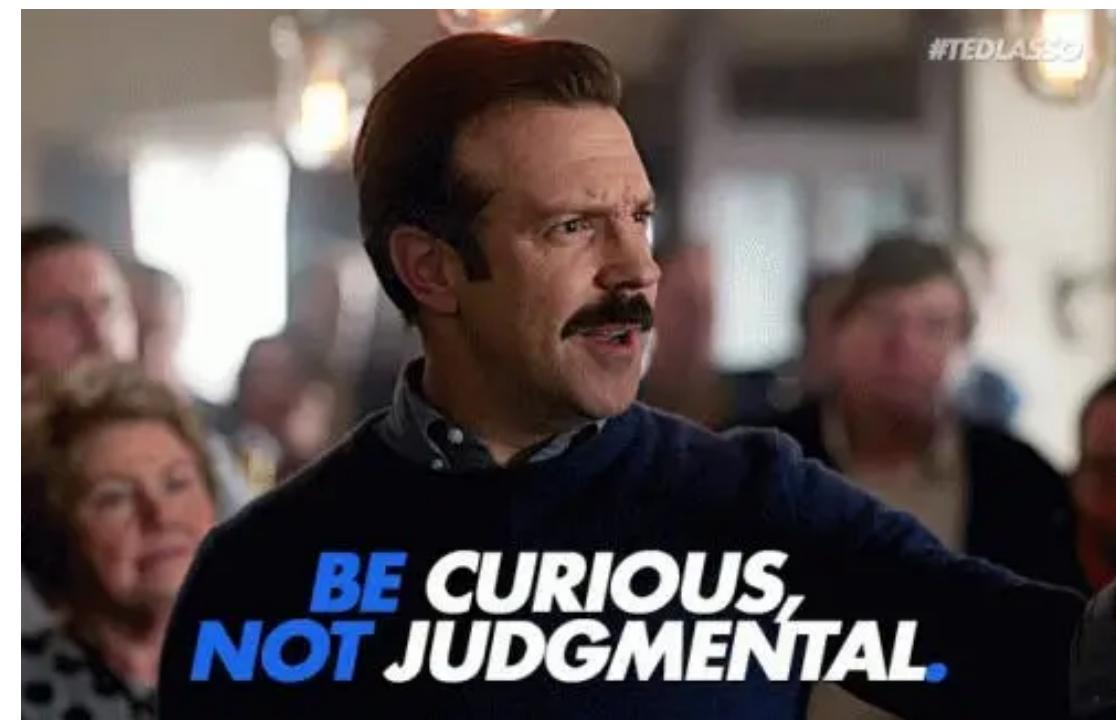


# Burying the Lead

- We bring one amazing superpower, **systems thinking**, which we can apply well beyond the engineering domain.
- But to use this skill, we have to learn to **hack the culture** and effectively translate our insights for our organization.

# The System: People, Process, Technology

Be Curious; Respect What Came Before.



Deepen Understanding; Build Empathy

Why?

Because policy/procedure...

Why?

Because audit finding...

Why?

Because non-technical person interpreted...

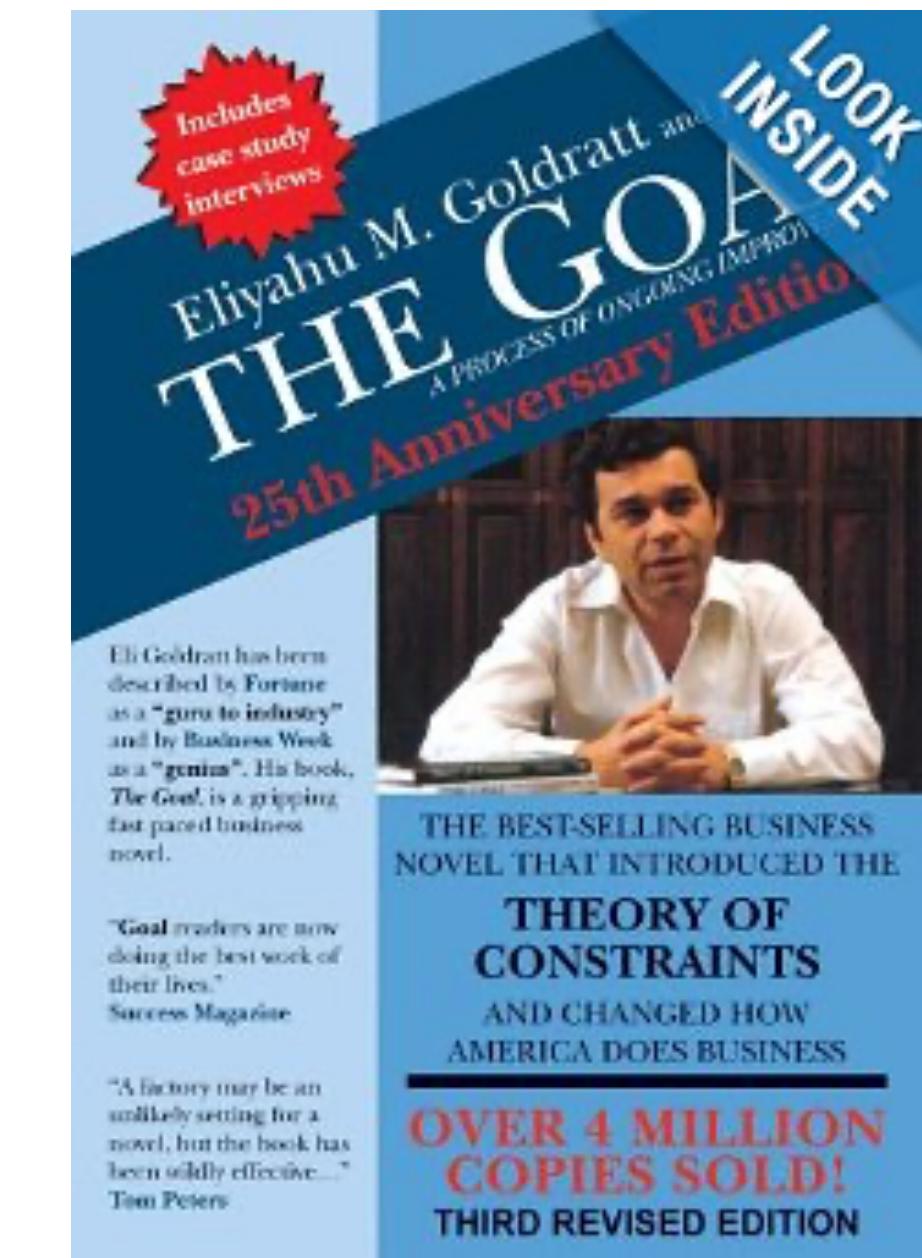
Why?

Because limited process for technical sign off...

Why?

Because no holistic target state...

Map constraints; find the bottleneck; remove the bottleneck



# Example 1: Month 4 on the job...COVID...

*“We need to run 100,000 transactions out of a mainframe through a manual data entry system designed for 100s a month without recruiting 1000s of employees to use a buggy 3rd party UI”*

# You have....



\* Armageddon, 1998

# You have 7 days...



# Typical dev cycle was 3 months

- Need to modify mainframe COBOL, CDC, and a service bus to speak to a 20 year old external SOAP interface.
- Ultimately boiled down to a couple of pages of code to convert format & data types, route the resulting data, and perform a web call and capture the result.
- Faced multiple limitations and impairments at the 3rd party that challenged how we would normally run a project like this.
- System Bottleneck — Alignment and proper approach to all impacted policy requirements and acquiring approvals from risk orgs, security, legal, and EA

# Hacking Culture: The Case Example

*“We had no idea it was possible to move so fast! How do we do this with more of our projects?”*

# Hacking Culture: Leading by Example

*I just watched an engineer make a legal point to a lawyer and win the argument in front of senior leadership...*

# Hacking Culture: What is valued?

*“It wasn’t so much that you got the result, it was HOW you got the result.”*

# Example 2: Building a Dev Platform

2020

**Ad-hoc ‘community of practice’ and advocacy via service teams**

*Change was hard:*  
- Policies and standards  
- Consensus decisions  
- Tight-coupling  
- Decentralized

*Point changes were nearly impossible.*  
Needed to change many things together.

2021

**Lead a large modernization program**

*“Keep the program green”*  
*Early proof points on evidence automation and modern practices. Build allies across org.*

2022

**Platform investment and process re-engineering attached to large initiative**

Addressed 3.5 of 4 key bottlenecks:  
- Decouple cloud platform from on-prem arch.  
- Process to revise policy driven procedures  
- Funding to staff platform product teams  
~ Improve cross-org decision making processes

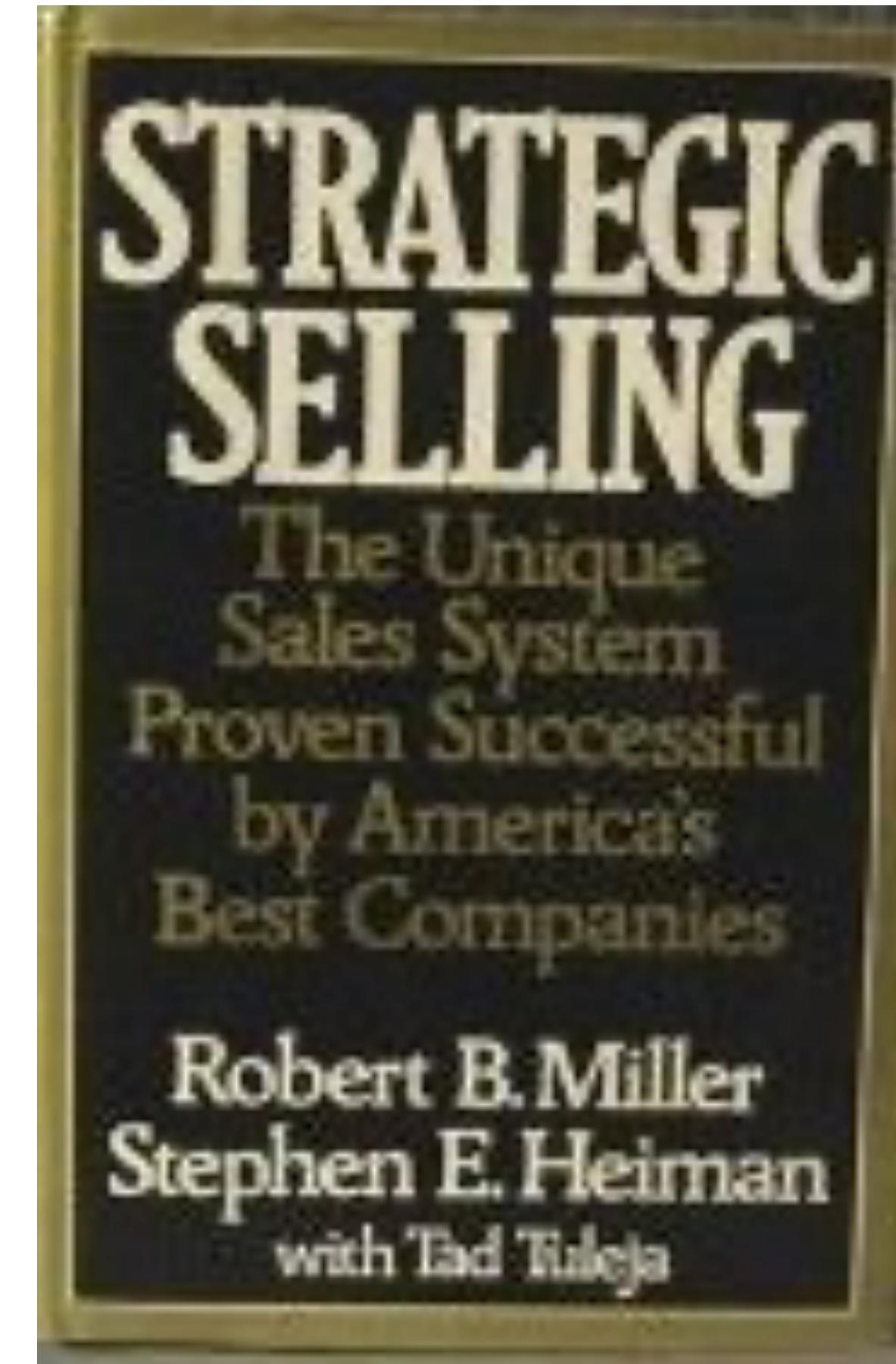
2023+

**Org and process optimization**

*Leveraging early successes and letting customers start to sell the value.*

# Strategic Selling to Internal Audiences

- Decision maker - Can and must say yes
- Gatekeepers - Can streamline or add friction to the decision process
- Bean counters - Enforce organizational guardrails
- Champions - Can influence the process



# Understand Decision Making

# Four sources of authority

- I report to you
- You have decision rights over my budget
- You own a policy and can tell me no
- Informal trust relationships + consensus

# Building Trust

# What drives senior leadership trust in your organization?

- Deep domain knowledge - demonstrate a profound understanding
- Delivery - “I don’t care how, just do it”
- Predictability over efficiency – call your shots and challenges; no surprises
- Personal Integrity - taking accountability for outcome
- Cultural conventions - do it the way we do it

# Experience at Amazon

- **Deep domain knowledge - demonstrate profound understanding**
- **Delivery - “I don’t care how, just do it”**
- Predictability over efficiency – call your shots and challenges; no surprises
- Personal Integrity - consistent behavior and accountability
- **Cultural conventions - do it the way we do it**

# Experience at U.S. Bank

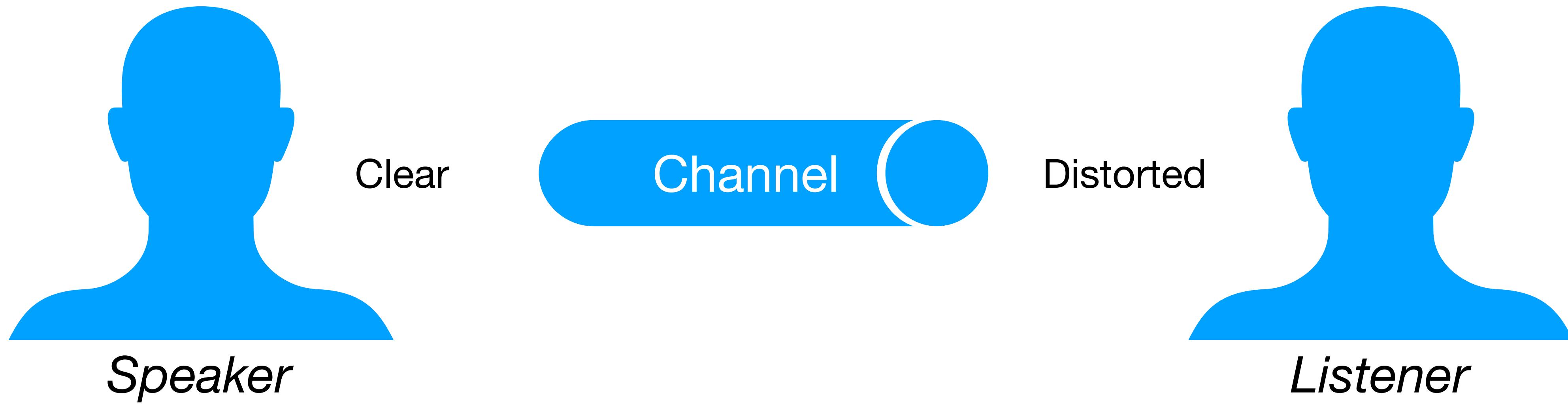
- Deep domain knowledge - demonstrate a profound understanding
- Delivery - “I don’t care how, just do it”
- **Predictability over efficiency – call your shots and risks; minimize surprises**
- **Personal Integrity - consistent behavior and accountability**
- **Cultural conventions - do it the way we do it**

# Bridging Communication Gaps

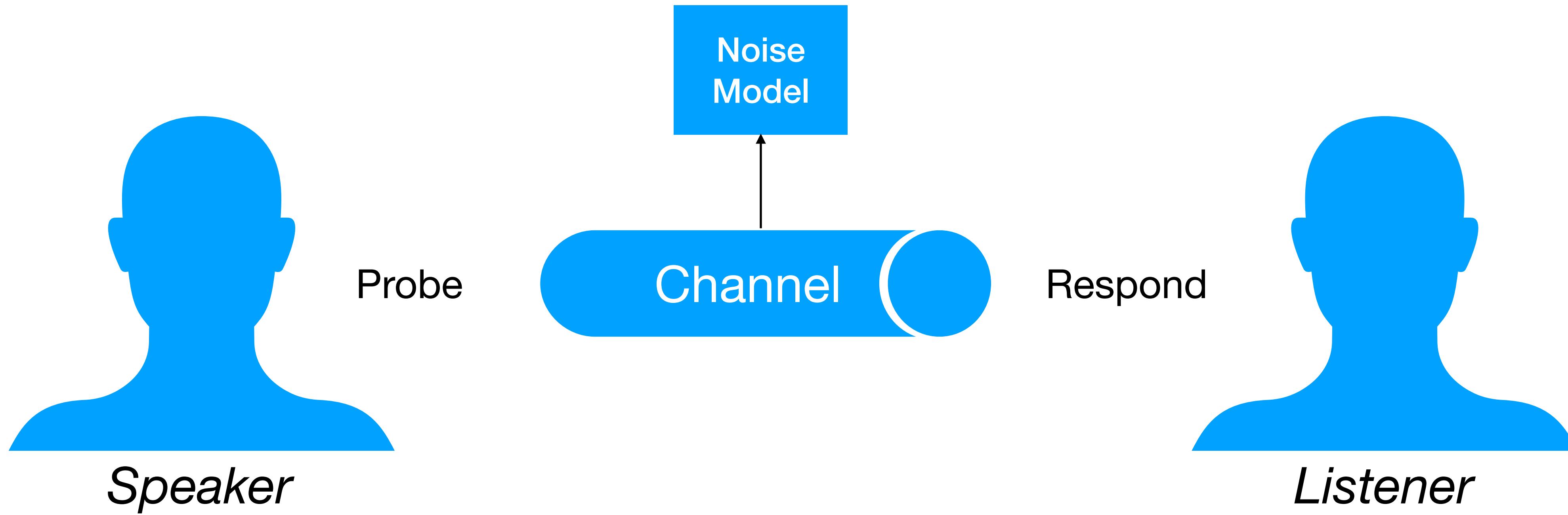
# Clear Communication is Your Responsibility

- The burden is on they who would be drivers of change
- 95%+ of “problems” are actually communications issues
- Do we agree on facts, goals, principles, tenets, logic?
- If we disagree, it’s an opportunity for me to learn or me to teach
- If we are at an impasse, to whom do we elevate/appeal to resolve?
- Never make it personal!

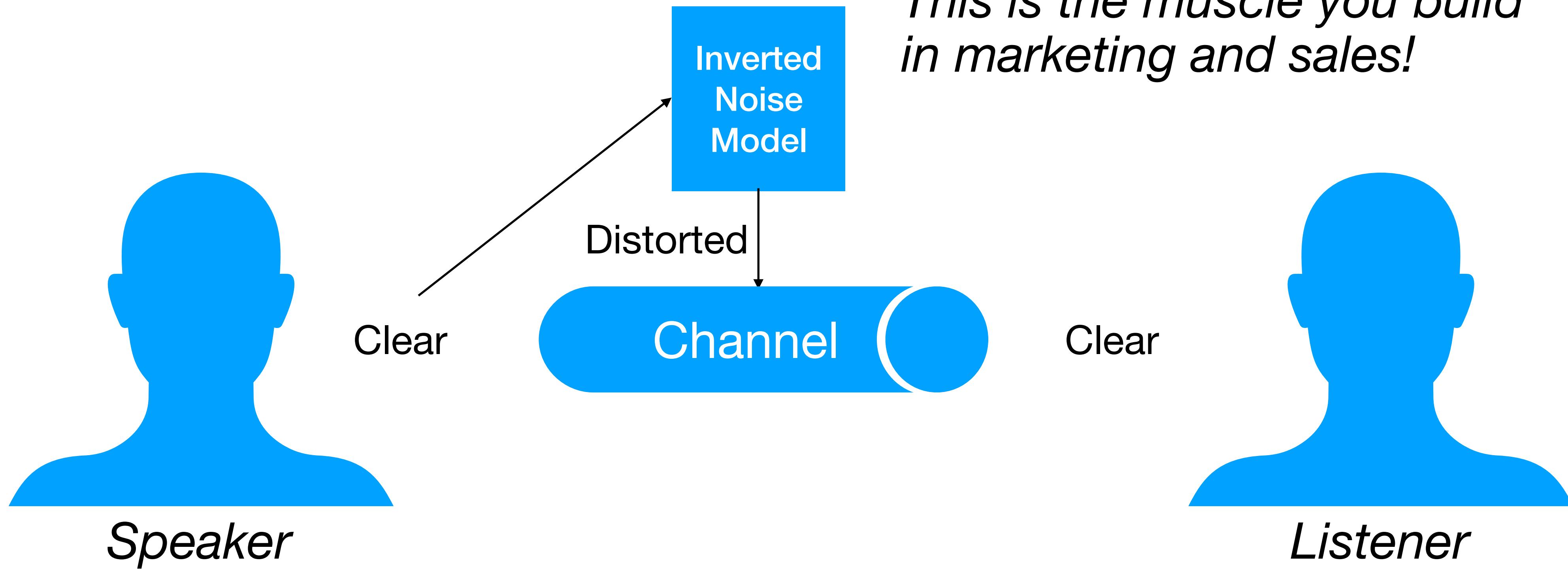
# Communication Channels are Noisy



# Communication Channels are Noisy



# Communication Channels are Noisy



# Supercharging your career

- Extend your **systems thinking** superpower to all parts of the company: finance, risk, security, etc.
- Map the constraints of your organization, identify the key mechanisms required to **hack** that system and practice, practice, practice.

**Thank you**