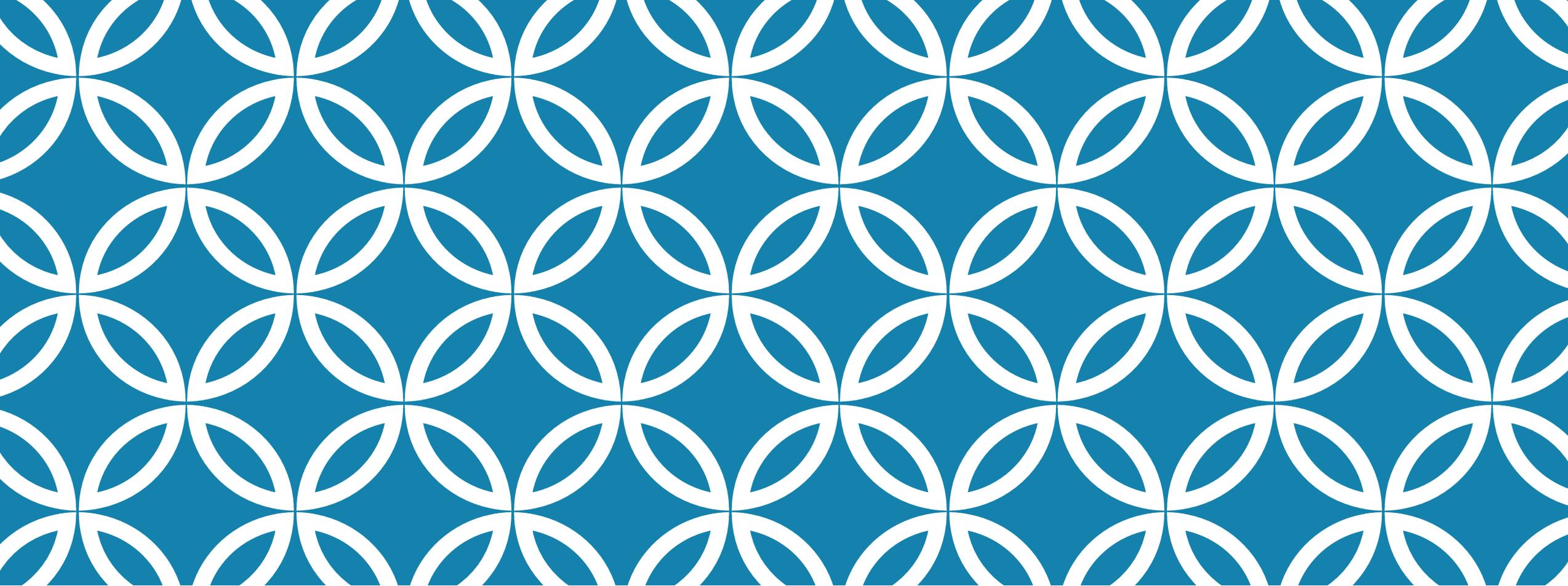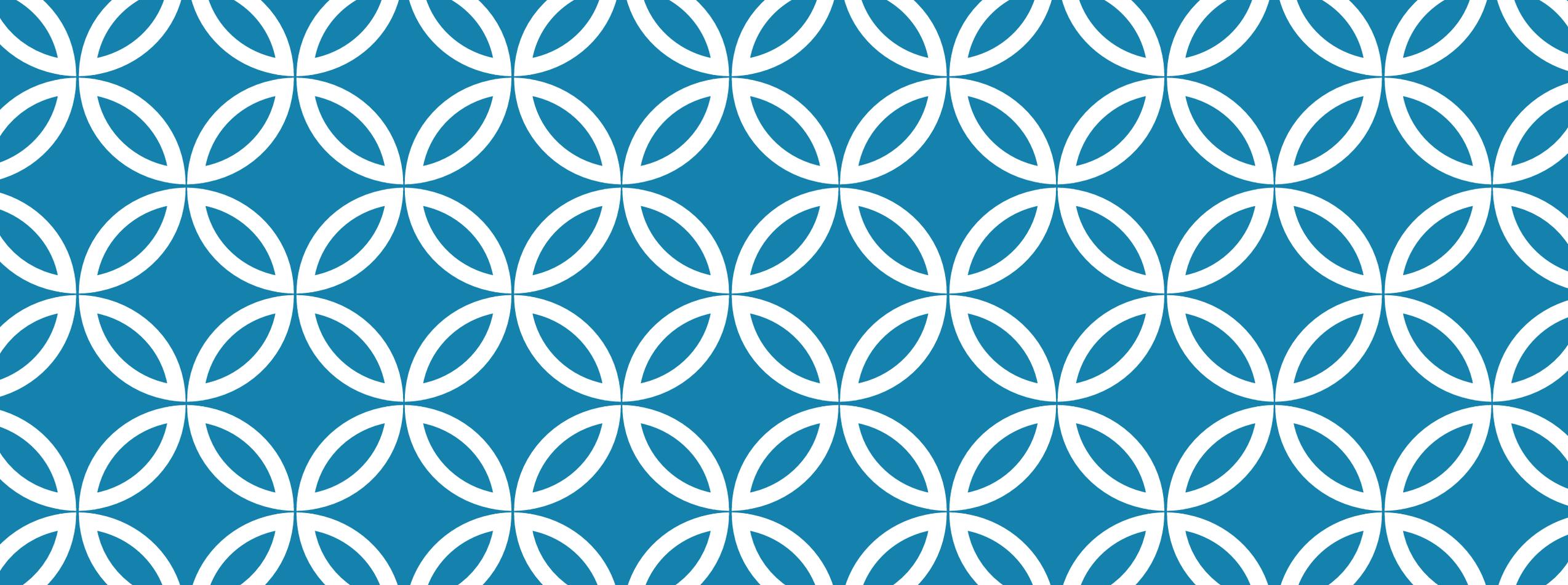# ADVICE ON HOW TO BE A GREAT CLIENT

Lessons Learned From a Decade of Partnership

HOW TO GET THE MOST OUT OF YOUR VENDOR

Lessons Learned From a Decade of Partnership

BUILDING EFFECTIVE IIS VENDOR RELATIONSHIPS

Lessons Learned From a Decade of Partnership

ADVICE ON HOW TO BE A GREAT CLIENT
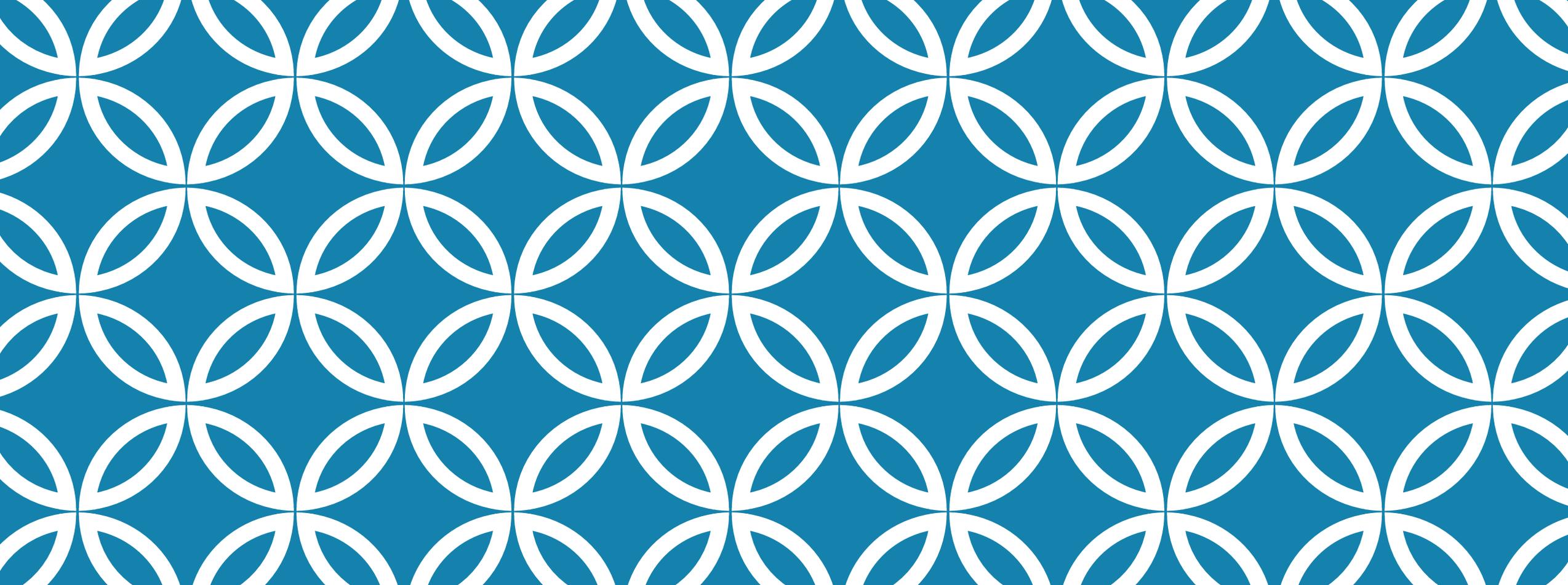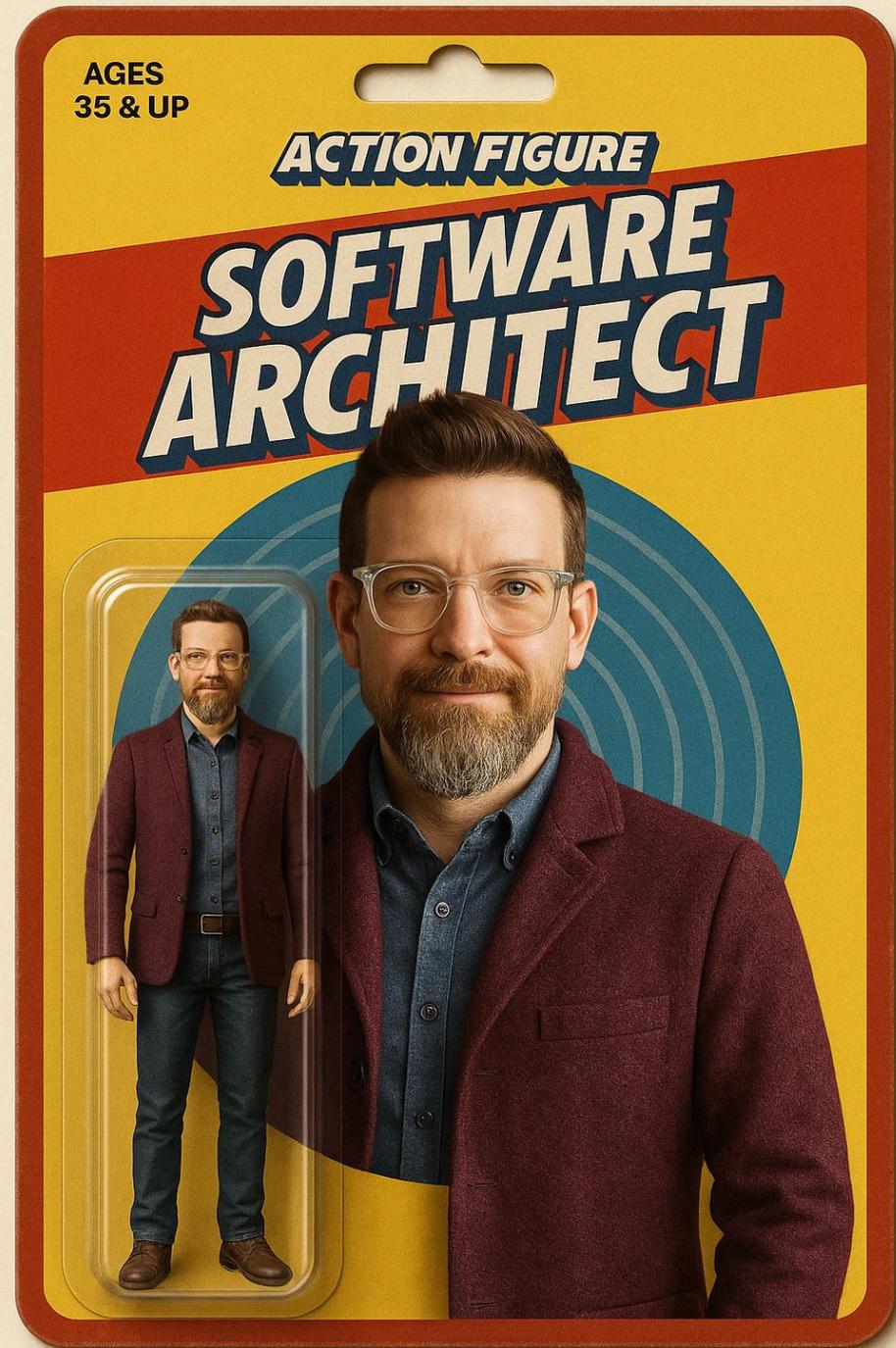
Lessons Learned From a Decade of Partnership

AM I A VENDOR?

AM I A VENDOR?

Started in 2015 with Michigan Care Improvement Registry

- AIRA National Meetings

- MACAW

- Early Days: Bob Swanson, Therese Hoyle, Gerry Bragg and Kevin Garnett

AM I A VENDOR?

- **Relationship:** 10 years of experience as an IIS Awardee-Developed custom software vendor

- **Continuity:** Four Distinct IIS leadership seasons within the State of Michigan's IIS program.
  - (Robert Swanson - Tina Scott - Ryan De La Rambelje - Abbie Cheney - Talisha Rice)

- **Organization:** I work for a small (was 35, now 150 ppl) organization that has active engagements in many domains with 35-45 distinct software systems currently under development or maintenance.

- **Personally:** focus on continuous improvement and seeing to the root of things.

A FEW OPENING THOUGHTS

- Vendors are people.

- People relationships are complicated.

- Communication is more complicated when you have people from different organizations.

WHY DO WE NEED VENDORS?

- You shouldn't try to be an expert in everything.
- It's faster/cheaper to hire someone who has made the mistakes on someone else.
- You don't know if you'll need them forever

A FEW MORE THOUGHTS

- Some vendor organizations are (very!) good and some are bad.

- It's hard to tell the difference ahead of time.

- Figuring out how is not this talk (Sorry!)

BAD VENDORS

- Serve their own interests first
- Don't communicate well with you
- Twist your arm to do what works better for them
- Aren't nice
- Motivated by money
- Leave you with messes

GOOD VENDORS

- Put your interests before their own
- Communicate excellently, and improve your team comms
- Work with you to discover what's best for you
- Are kind and patient
- Financially aren't dependent on your funds
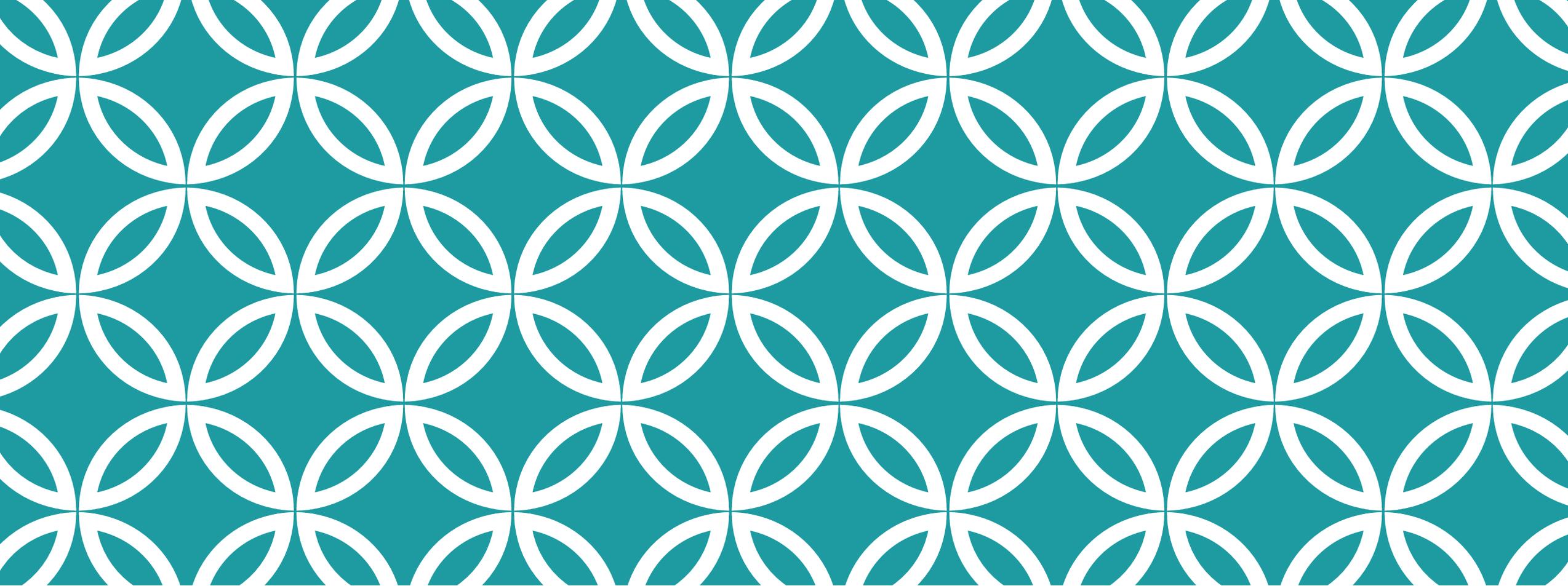- Constantly working to improve deliverables

ASSUMPTIONS

1. Your vendor has your best interest in mind.

2. Your vendor has relevant expertise.

3. Your vendor asks you good questions and doesn't get ahead of you with solutions.

4. We're talking about software
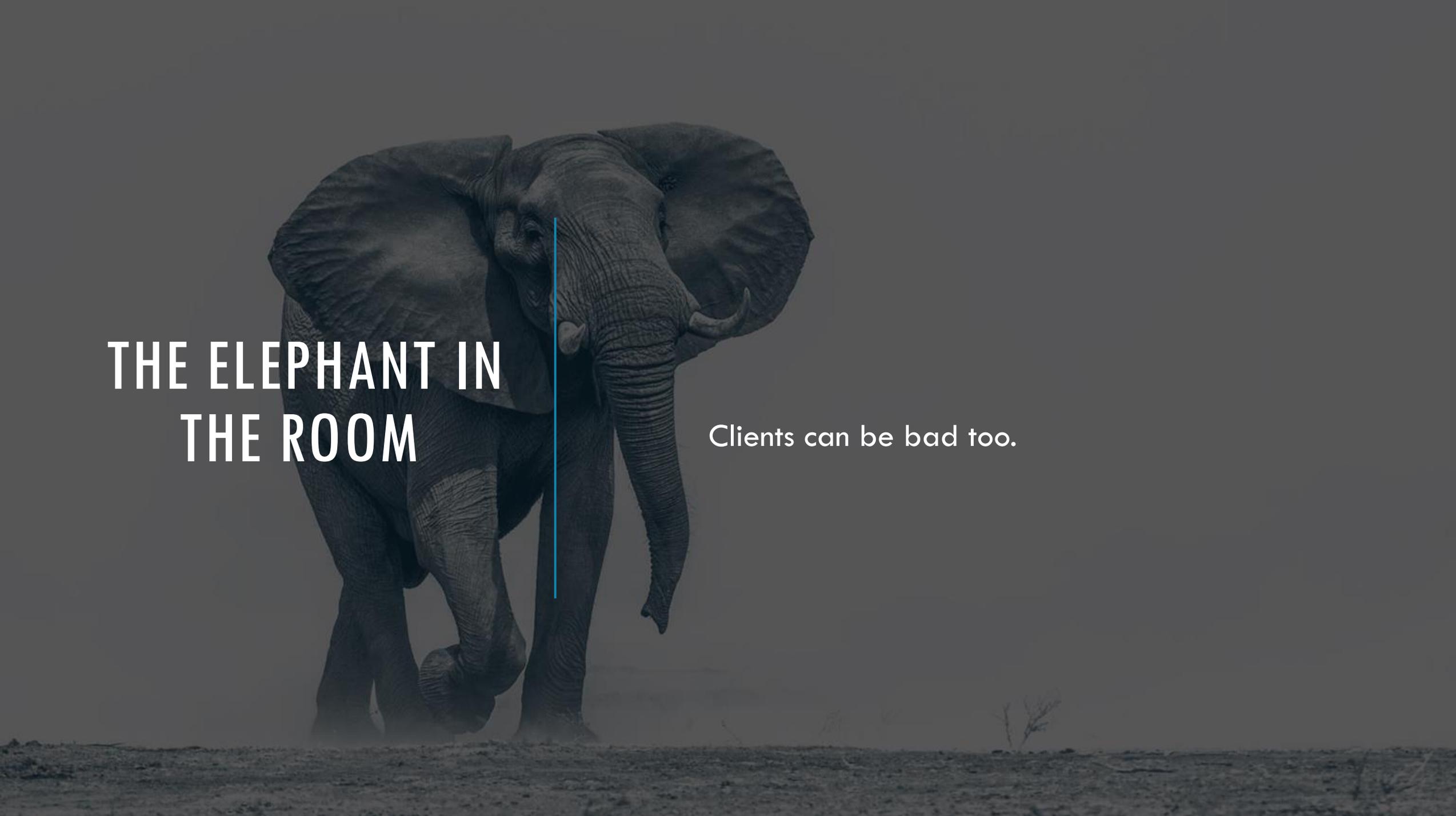
THE ELEPHANT IN THE ROOM

Clients can be bad too.

ADVICE ON HOW TO BE A GREAT CLIENT

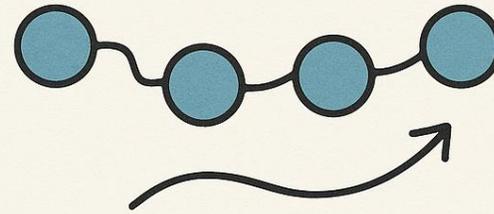From a vendor

HOW TO BE A GREAT CLIENT

be nice

FOUR UNCOMMON (BUT MAYBE OBVIOUS) PRINCIPLES

1. Be Outcome Focused

2. Assign a (True) Product Owner

3. Leverage the Currency of Trust
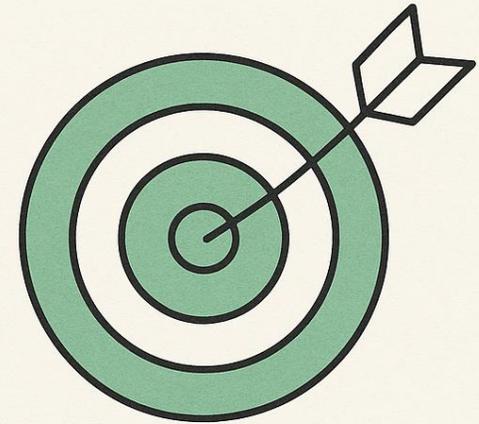
4. Acknowledge Every Solution is a Guess

BE OUTCOME FOCUSED

Outcomes v.s. Projects

What is an outcome?

Projects	Outcomes
Reminder Recall Changes	Population Immunization Rate Increase
Auto-Merge Improvements	Staff Operational Efficiency
Data at Rest Measurement	Stakeholder Trust Increase
UI Improvement	Staff productivity gains (more stuff done)
Forecast Schedule Updates	Increased Provider Trust

THE BUILD TRAP

More Features = More Value

Specific

Measurable

Achievable

Relevant

**T**ime-bound

THE BUILD TRAP

~~More Features = More Value~~

More Features = More Maintenance

SIGNS YOU'RE NOT OUTCOME FOCUSED

- You haven't told your vendor why you want to do it

- You don't know what will happen if it works

- It's "The right thing to do"

- You immediately move on to the next project.

- Your project to-do list is long

- You track the timeline and deliverables but don't track the effect

- People working on it are unmotivated

- You never check back with the people who are impacted after it's done
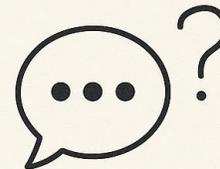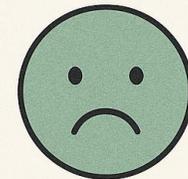

SIGNS THAT YOU'RE NOT OUTCOME-FOCUSED
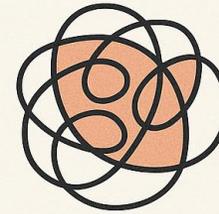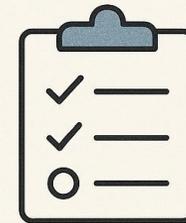
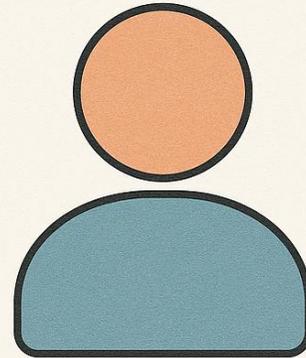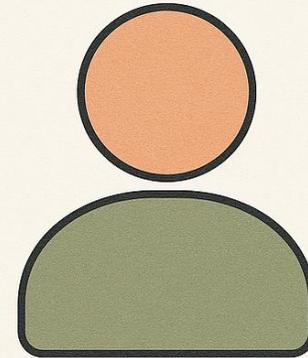LACK OF CLARITY   TOO MANY TASKS

LITTLE FEEDBACK   LOW MOTIVATION

ASSIGN A TRUE PRODUCT OWNER

Product vs. Project owner

PRODUCT VS. PROJECT OWNER

PRODUCT OWNER

PROJECT OWNER

Product	Project
Focus: Value creation and business outcomes	Focus: Timeline adherence and deliverable completion
Meetings are primarily about feedback, value metrics, and priority adjustments	Meetings primarily about timeline status, resource allocation, and deliverable checklists
Requirements evolving based on learnings and validated through user feedback	Requirements are fixed early, changes require formal change control process
Regular interaction with end users and beneficiaries of the solution	Regular interaction with project sponsors and management
Regularly adjusts approach based on market feedback	Conducts post-mortems after project completion
Partners with vendors to achieve outcomes, shares risks and rewards	FManages vendors to deliver contractual obligations

SIGNS YOU ACCIDENTALLY ASSIGNED A PROJECT OWNER

- Scope changes are scary

- They mostly talk to you about timelines and deliverables

- They seldom talk with users

- User research and testing are foreign concepts

- Measures productivity by output volume

- After release focused on "Did it release"

- Roadmap Discussions center on delivery rather than outcome expectations

- Your team isn't motivated to deliver

Signs You Accidentally Assigned a Project Owner

⚠️

Celebrates timeline adherence over business impact

Focuses on "percent complete" rather than value delivered

Reports emphasize activity metrics not business outcomes

Treats testing as a schedule risk not a learning opportunity

Resists scope changes even when they'd improve outcomes

Minimal engagement with end users or customers

Shows little curiosity about actual user experiences

Rarely questions if features are worth building

Shift focus from project delivery to product outcomes

Thanks For Nothing Claude...

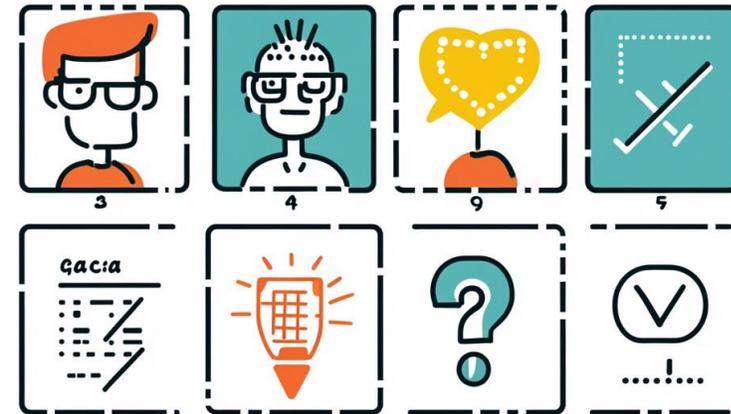# LEVERAGE THE CURRENCY OF TRUST

Trust v.s Force

Trust	Force
Focus on outcomes and acceptance criteria, emphasis on shared understanding	Contract-like specificity on deliverables, emphasis on protection of interests.
Collaborative meetings and lightweight approvals focused on key decisions	Meetings primarily about timeline status, resource allocation, and deliverable checklists
Focus on value of the change, Streamlined process for incorporating new insights	Requirements are fixed early, changes require formal change control process
Open and transparent discussions	Formal, documented exchanges. Emphasis on accountability
Meetings focus on solving problems together	Focus on status reporting

SIGNS
YOUR SYSTEM OF INTERACTION IS OVER ENGINEERED

- You and your vendor are annoyed with approval requirements

- Required documentation you have to write but nobody reads

- Your business requirements read like a contract, but you all think it's obvious already.

- Status meetings where nothing substantiative is discussed (because you're already aligned)

- Formal change request processes that teams together find workarounds to avoid

- Team energy focused on process compliance rather than solving customer problems

ACKNOWLEDGE THAT EVERY SOLUTION IS AN INFORMED GUESS

Hypothesis v.s. Certainty

Guesses

Reminder Recall Changes will cause more people to take action

Auto-Merge Improvements will free up staff to work on the hard cases.

Data at Rest Measurement will show that our data is trustworthy and increase stakeholder trust.

UI Improvements will make our system look more modern and people will like that.

Observed Results

10% More people visit the doctor when they get the letter

Staff report spending 50% more time connecting with providers to resolve duplicate entries in the system.

We found issues with particular providers. We worked hard to get them to change, and our action increased stakeholder trust way more than the measurement did.

People didn't care, but our staff is getting fewer calls for help because the system makes more sense now.

SIGNS
YOU ALWAYS THINK YOU'RE RIGHT

- You never share why you want to do something

- You're not open to new information

- You don't ask your team or vendor "Can you think of other things that might work"

- You don't document assumptions that led to design decisions

- You get defensive when your idea is questioned

- You never ask for people's opinions

- People are afraid to tell you their opinion