

HCD (UX) and Agile

Yesterday, Today, and Tomorrow

A look at how HCD has evolved, the ways in which it does (and doesn't) play well with Agile, and where we can go from here.

Brad Lehman

brad@humancenteredagile.com

HCD Practice Lead

Author of the forthcoming book *Human-Centered Agile*

On-air DJ in Takoma Park, MD (and takomaradio.org)



Why do we look at yesterday?

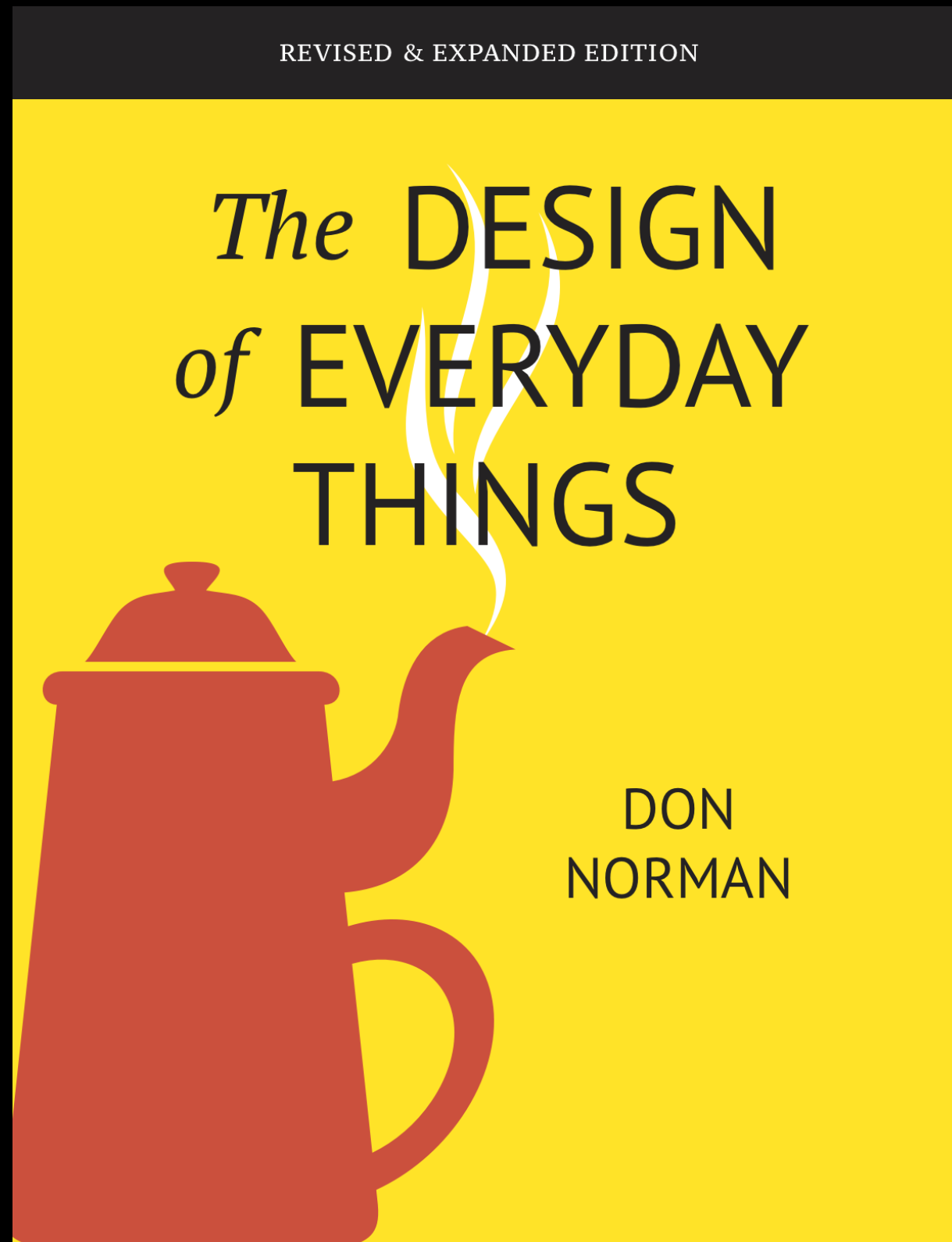
Whenever you remove any fence,
always pause long enough to ask
why it was put there in the first place.

Gilbert K. Chesterton

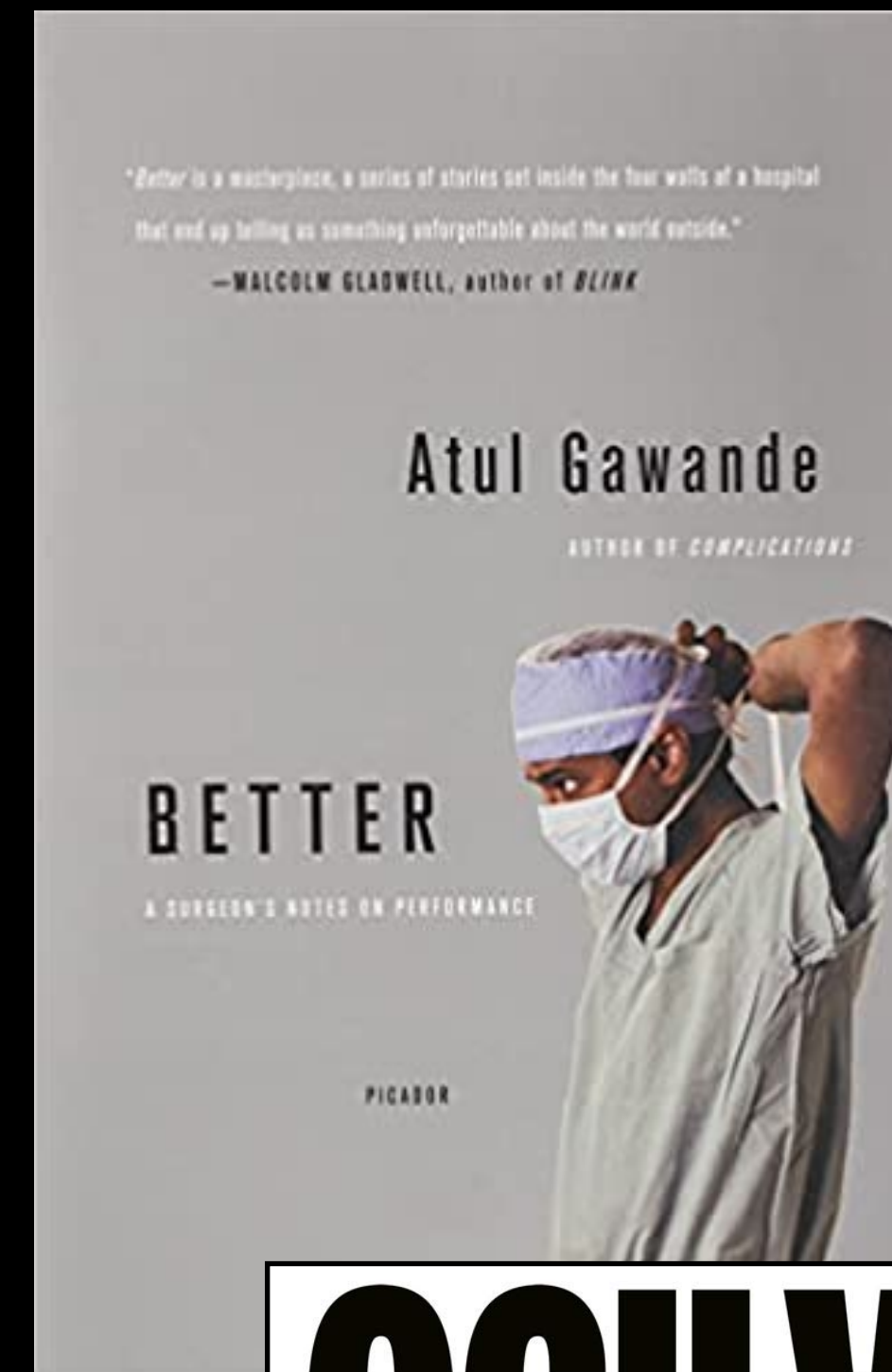
quote fancy

HCD is old

Not Just...



But Also...



**OGILVY
ON "I hate
rules"
ADVERTISING**



HCD is new

Shift to short delivery iterations

- Software delivery in the internet age is different than physical/industrial design.
- It's even different than software delivery through the 90s and early 2000s.
- Those differences change how we do things.

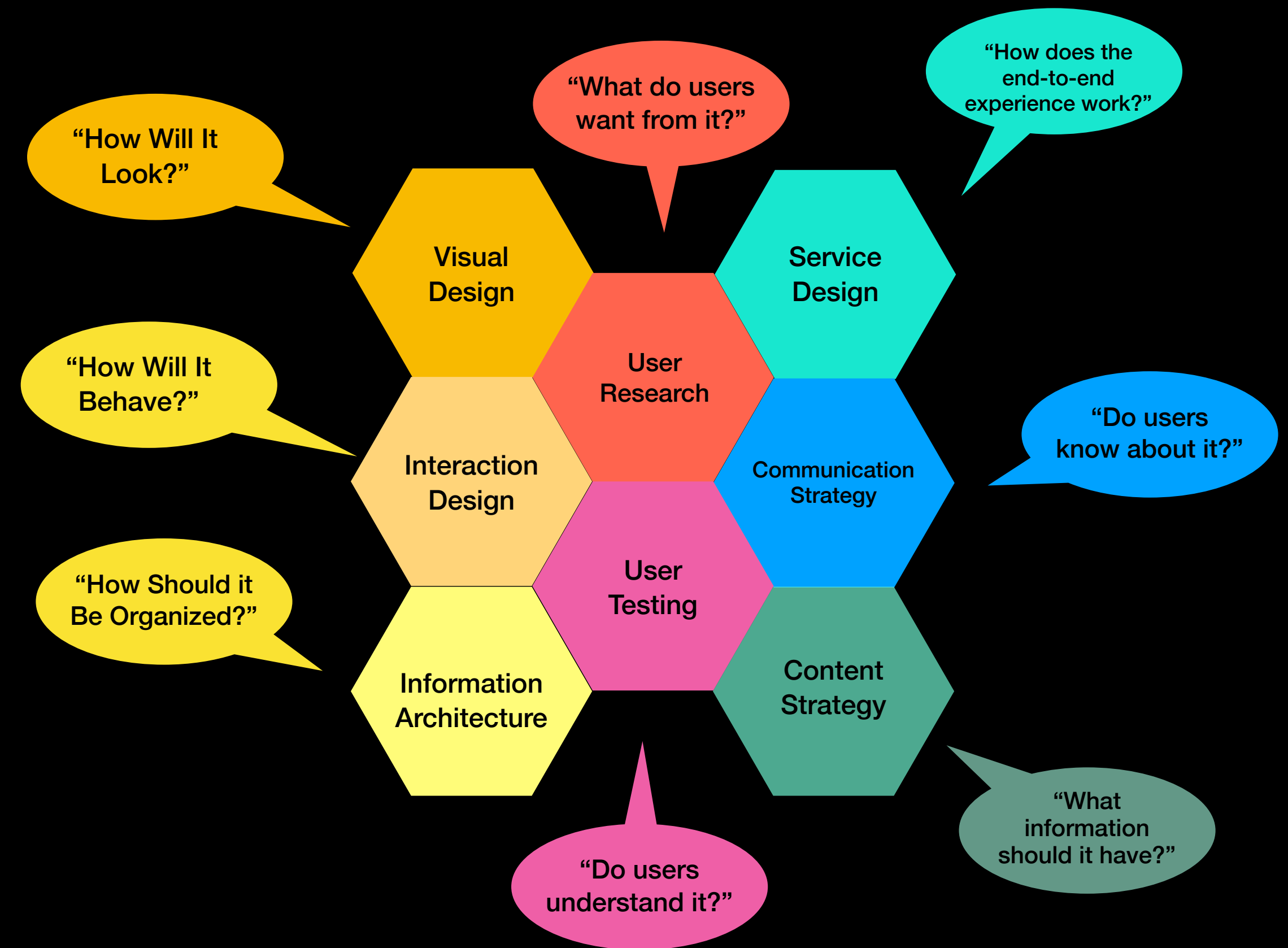


HCD is new

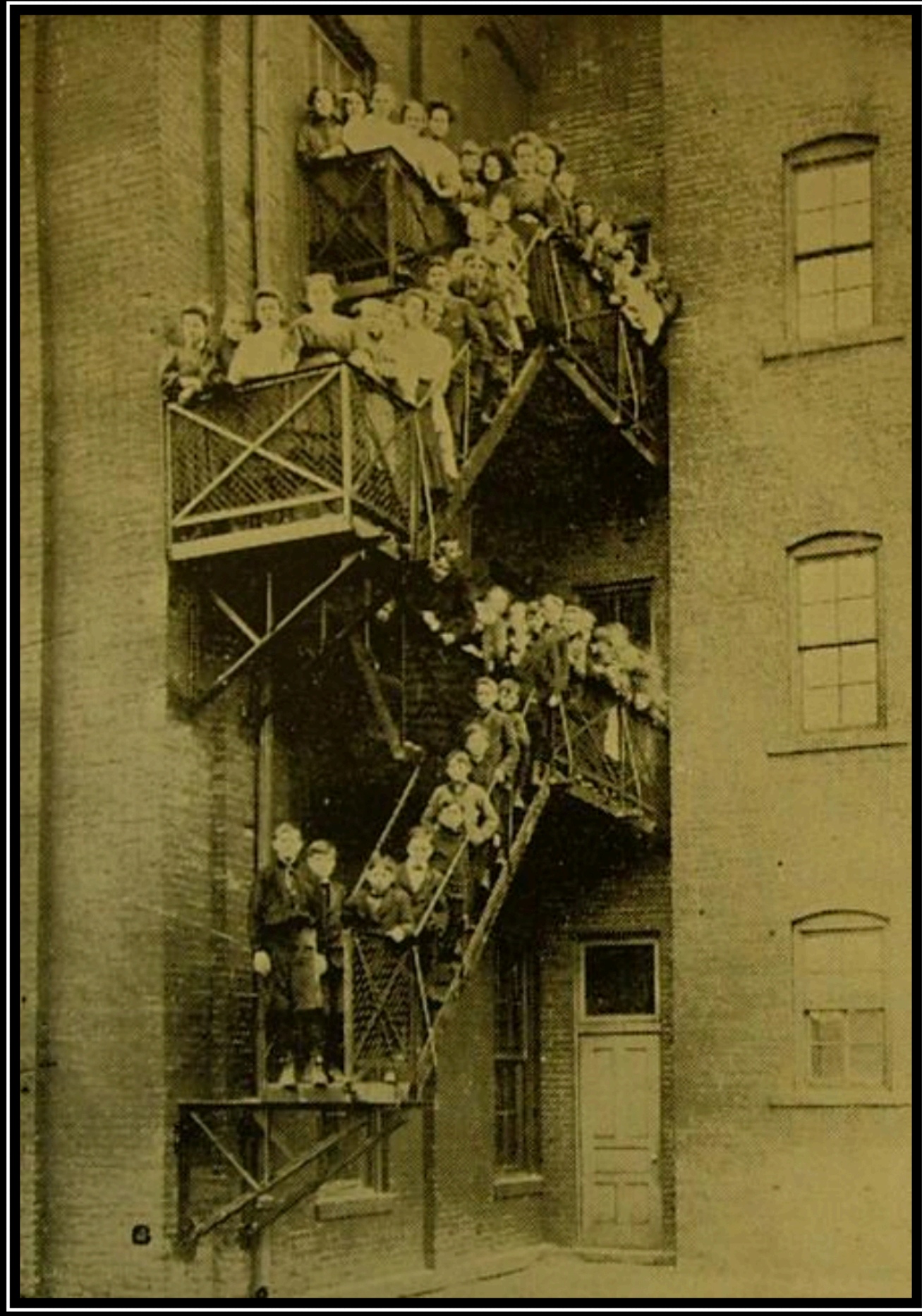
People still think we do this....



...But really we do this



Change takes time



Then...

... And Now



Life in 2010



- Endless requirements gathering. Occasionally this involved user discover, but often were just “requirements gathering.”
- Over-the-transom handoffs, followed by decisions that badly compromised design.
- Developers getting stuck in loops.
- Brad almost quits design entirely.

Starting with Agile... for me



- It was a 10 person team
- It was, itself, an experiment
- The spirit of collaboration was important. If you weren't open to discussion, you were out.
- The spirit of improvement was important. If you weren't willing to change, you were out.
- Clear shared values of learning as quickly as possible and updating your approach, and valuing experience in a 360° way

Starting with Agile... for me

ADVANTAGES

- Frequent collaboration
- Engaged Product Owner
- Commitment to Agile, not just Agile words
- Access to users, willingness to research

EASY
MEDIUM
HARD

CHALLENGES

- Joining the work in flight - catching up meant working on everything at once.

Problems that Agile solved

- **User stories changed the conversation to user-based value delivery**
- Being embedded meant no bad handoffs, and an ability to influence design from the earliest moments
- Collective sizing prevented getting stuck
- Retrospectives meant we improved as we went

I didn't know this yet, but this was my first real experience with outcome-driven design.



Today



Problems that Agile... didn't solve

- Not enough time in the 'Problem Space'
- General acceptance and adoption of HCD
- Short work cycles and Commoditized HCD - wireframes and user tests "by the pound"
- Iteration vs Incrementation



Which leads to these outcomes

- *'We'll do discovery in Sprint 0' and 'Feed the Beast'*
- Overtasked and siloed designers
- Poor consistency across efforts
- "Race to the bottom" MVPs that don't get improved.

Why is this?

- Design maturity is lagging behind Agile adoption.
- Despite it's intended focus on user value delivered, Agile prioritizes time in the Solution Space.
- Scaling up reproduces old-school PM techniques - blind hand-offs, pressure on predictability, etc.

Why is this?

Conway's Law

Any organization that designs a system (defined broadly) will produce a design whose structure is a copy of the organization's communication structure.

“If you have four groups working on a compiler, you'll get a 4-pass compiler.”

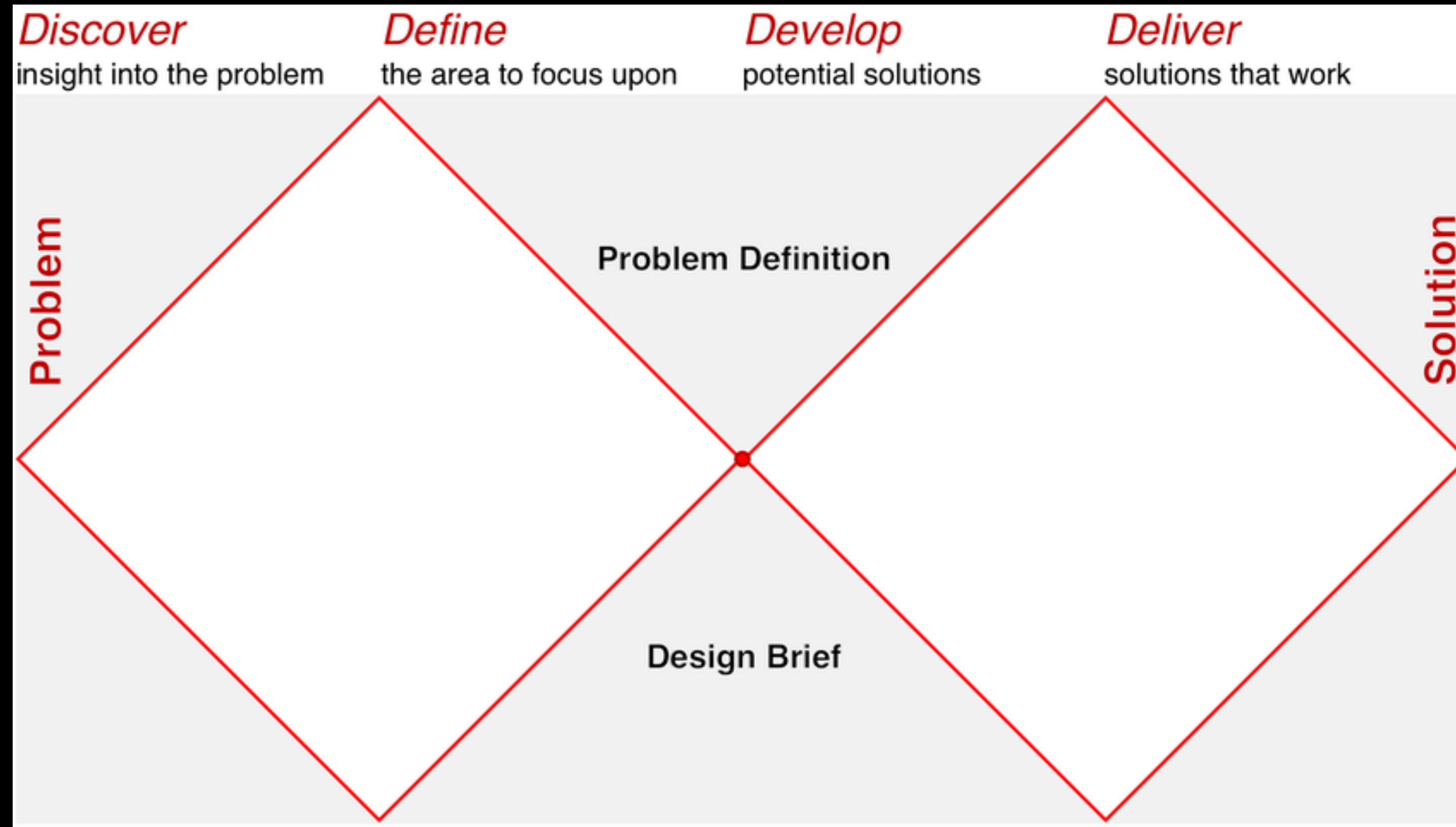
Byrne's Stage



All creative work is adaptive to the environment in which it is created.

Why is this?

Everyone's favorite Double Diamond...

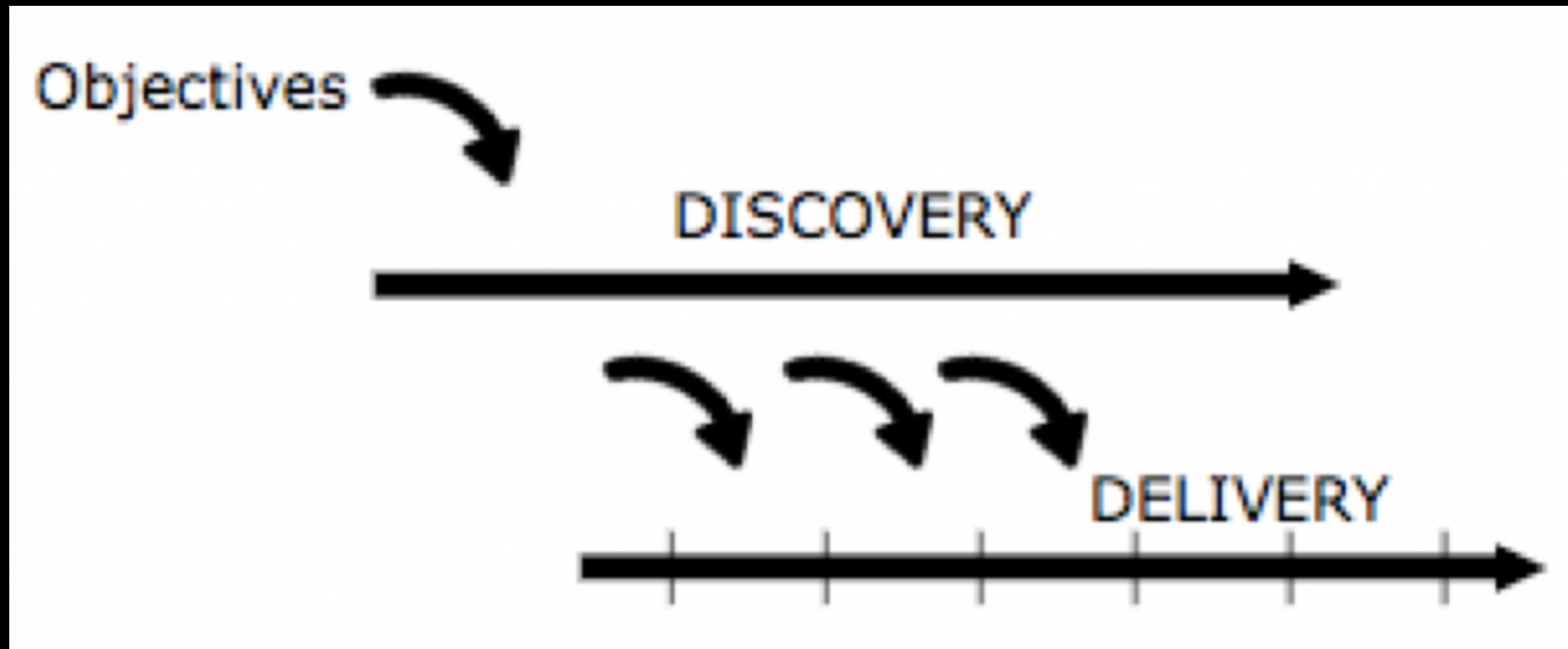


Why is this?

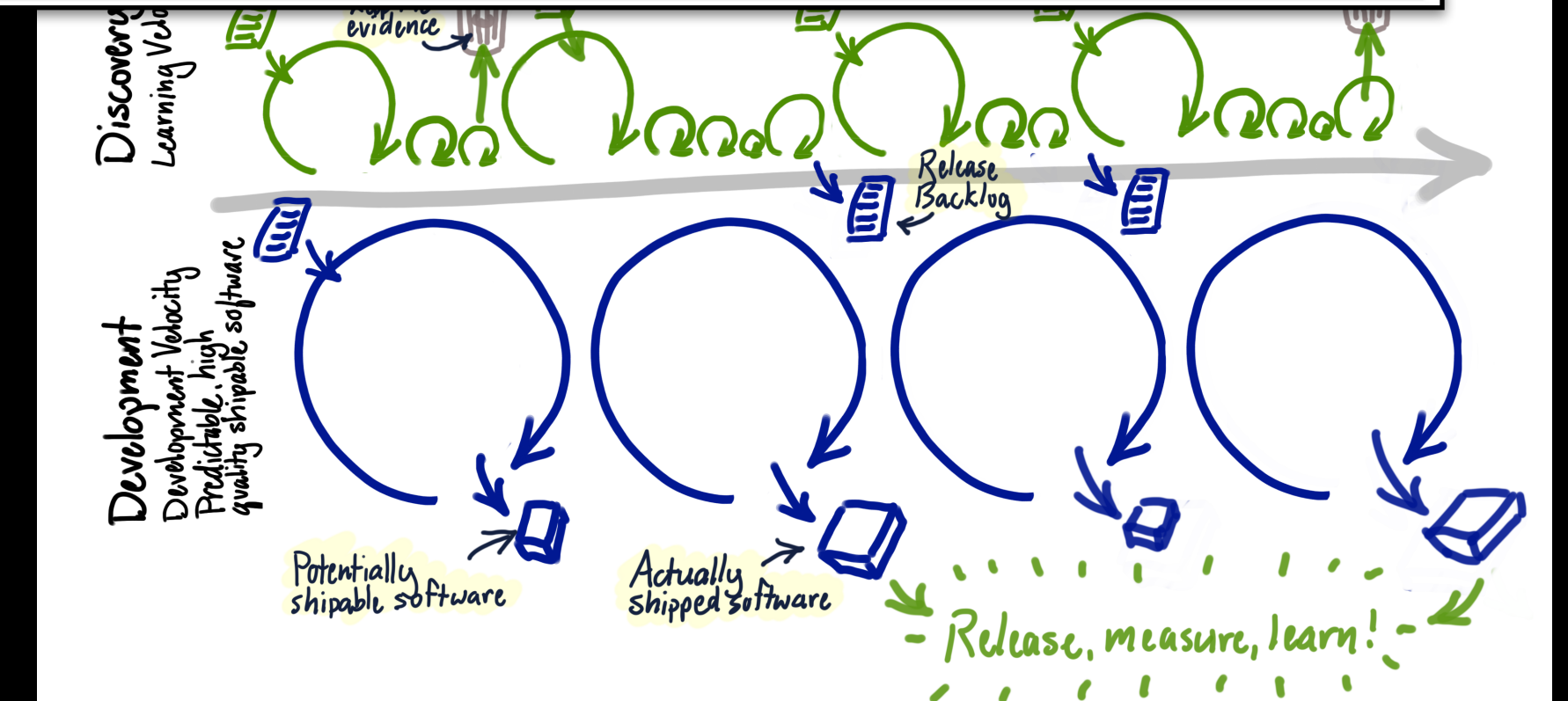
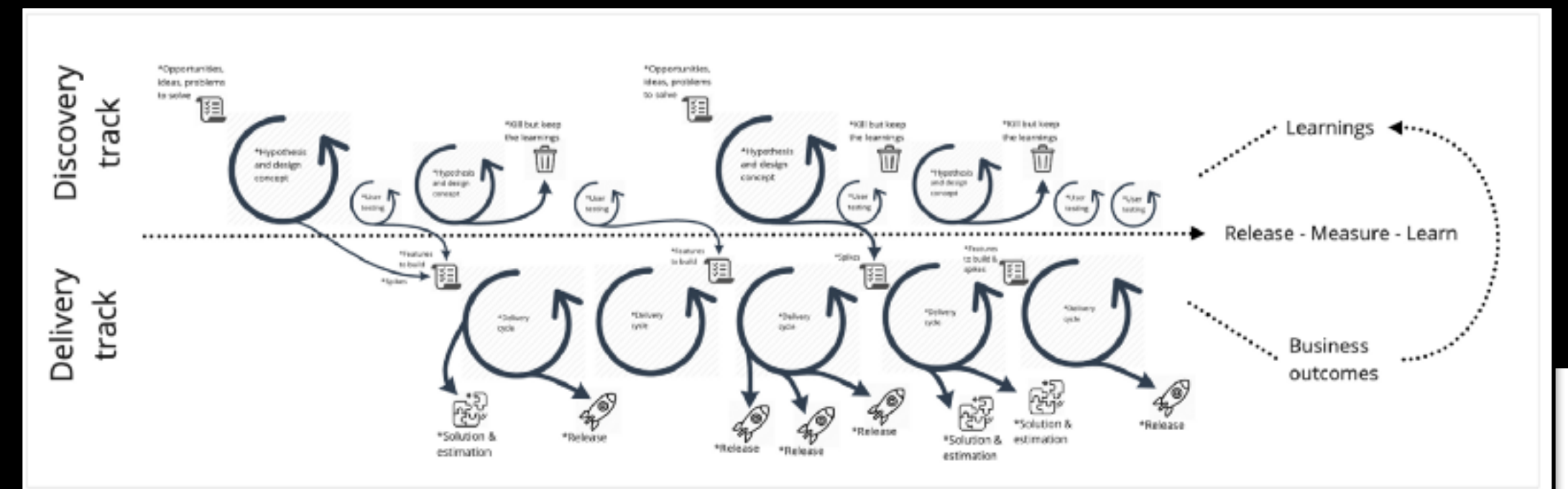
...Leads to Dual-Track Agile

What it should be

This is what the PO sees



A set of research and design deliverables that line up perfectly to delivery, whose LoE can be negotiated



Recursive and situational, where we validate problems, concepts, and delivery

Why is this?

...And Leads to Separated Concerns



One group is trying to generate “the best experience” while another group is trying to “ship as quickly as possible.”

This ends up being mediated by the PO, and in my experience, tends to come back to Output thinking.

Tomorrow



“Good decisions come from experience.
Experience comes from bad decisions.”

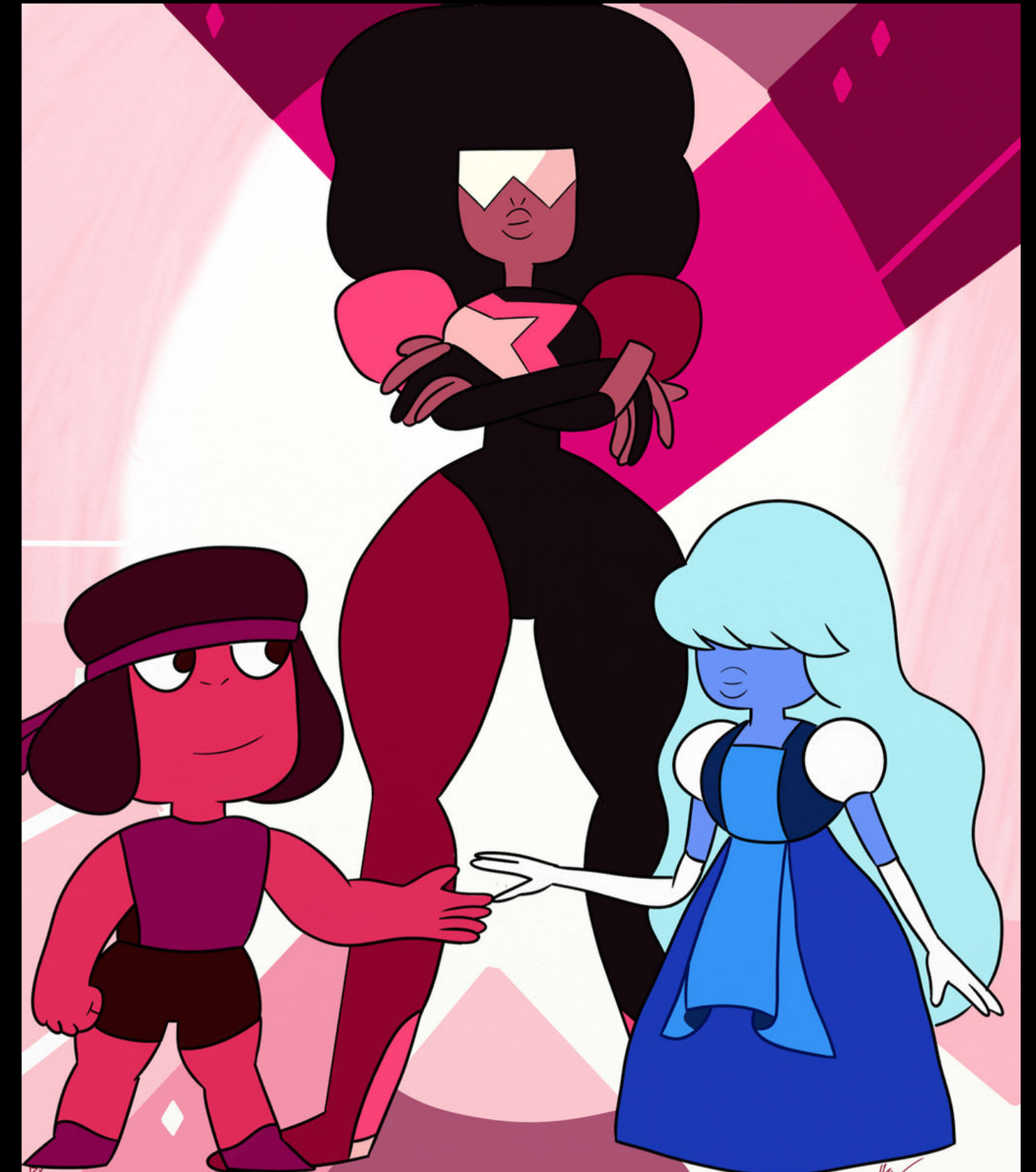
Where we go



- A more holistic approach to design and Agile
- A more ethical approach to design

Holism

- Needs analysis at every level - Hand off Problems or Challenges, not tasks
- Awareness that problems, concepts, and solutions are prone to change and updating
- “Right Problem, Right Solution, Done Right*” - these interests remain open throughout the process
- The right approach - to process, team topology, etc - is the approach that maximizes shared understanding and collaborative decision-making throughout the process.
- This is a **culture change** as much as it is a process change, if not more



*Jen Cardello

<https://medium.com/fidelity-design/how-a-product-design-framework-guides-ux-research-de0e371384e9>

Ethics

- It was 20 years into my career that ethics even surfaced onto my radar (thank you, Mike Monteiro)
- The only real ethics topic that ever got addressed was accessibility, which is a narrow section of inclusivity.

Move fast and break...



...people

An extra word about design ethics

- Don't be afraid of being a "first conversation"
- Don't be afraid to ask uncomfortable questions about the impacts of your design decisions outside of immediate business concerns.
- Build common vocabulary - we do this by writing, responding, and gathering up our conversations into a lexicon.

And Back to Today



Product Level HCD + Agile

Ingredients for success

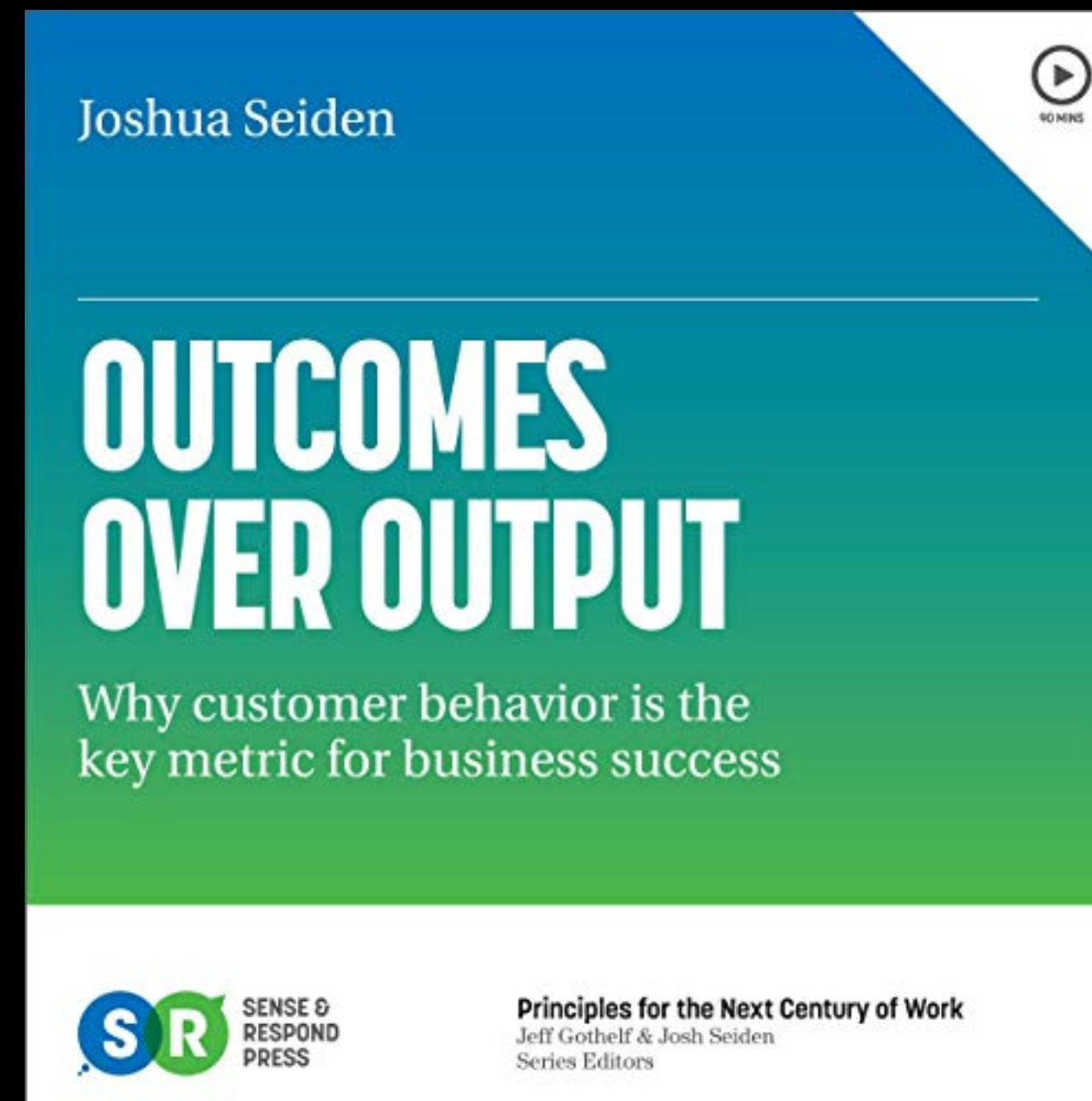
- Engaged Product Owner
- Designer-Team integration
- Access to users
- Synthesis time for discovery and test findings
- **Metrics Tied to Design Success**
- **Genuine permission to learn/fail**

Conversations that work

At a product level

Alignment and vision

- ‘What problem are we trying to solve?’
- ‘Right Problem, Right Solution, Done Right’
- ‘What are the stakes?’
- ‘Sometimes the *most* expensive way to find out whether something works is to build it and see.



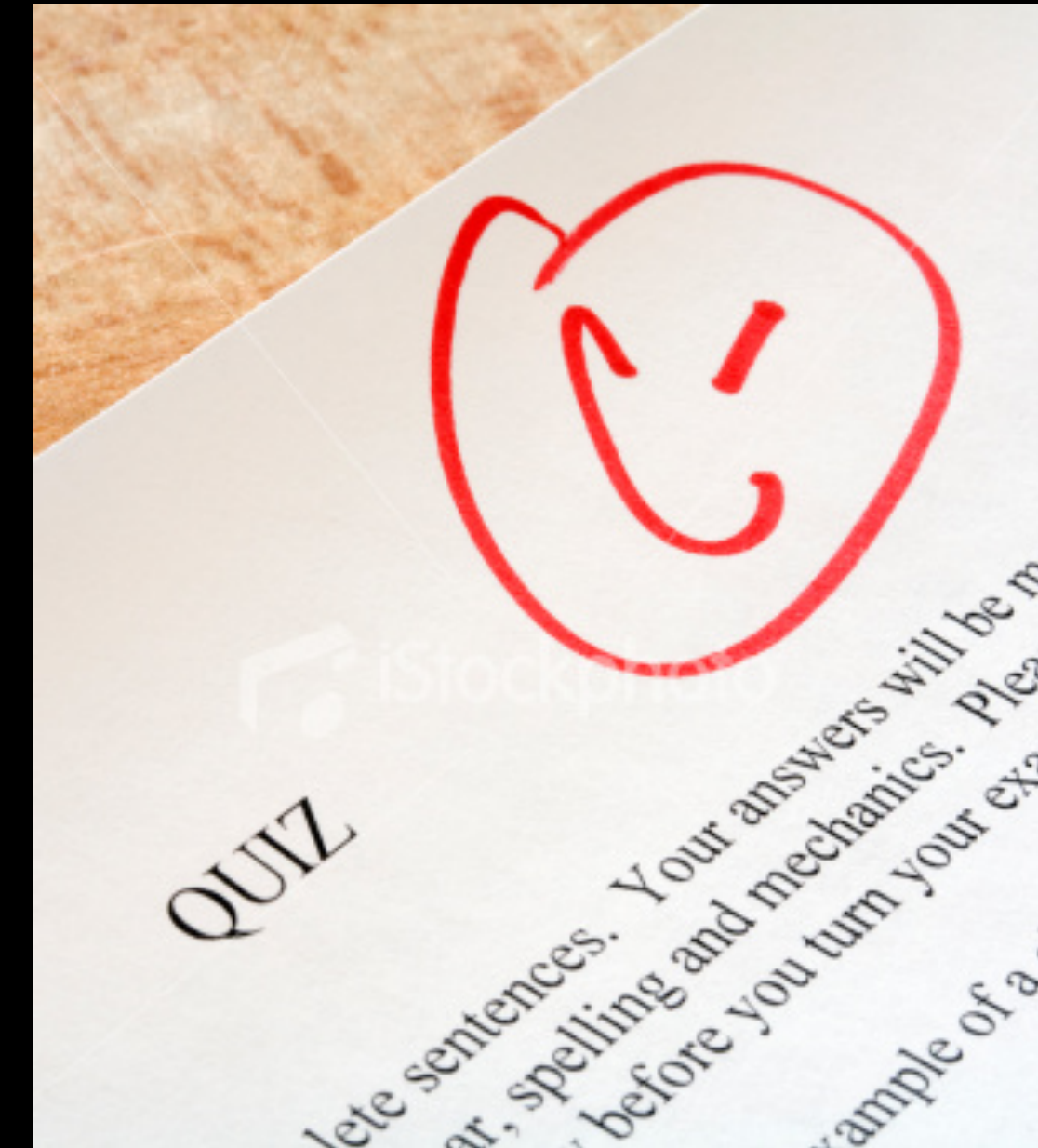
Get your product owners / business stakeholder to step out of delivery and project their thinking into outcomes.

Conversations that work

At a product level

Delivery Hurdles

- ‘How will we know we did a good job?’
- ‘Positions vs Interests’
- Activities vs Artifacts
- Moving from methodologies to purpose



Start speaking the same language about outcomes, trade-offs, and risks.

Purpose - Experience Elements

- Awareness
- Arrival
- Wayfinding
- Orientation
- Decision Support
- Action
- Expectation Setting

At a program/org level

- Needs analysis and orchestrated hand-offs
- A research roadmap
- A focus on outcomes (and not output) from top to bottom - with measurement
- Audit/Review projects against *Ingredients* and *Stage of Entry*

The goal is to make a shift from “champion-driven” design to “outcome-driven” design.

Intake / Needs Analysis and hand-offs

- Not just at “Discovery/Experiment” Steps
- Can be a workshop or a huddle
- HCD-style shareout when work moves from team-to-team
- The questions of “Right Problem, Right Solution, Done Right” become the purview of every successive team

As soon as one team tasks another team to deliver its ideas instead of taking over a challenge, you are beginning to reproduce waterfall thinking.

Research Roadmap

- Research time and support becomes responsive to program-level needs.
- Stakes drive the decisions
- A formal ceremony has a forcing function of breaking the HCD/UX silos - better information sharing, better collaboration

The goal is to make a shift from “champion-driven” design to “outcome-driven” design.

Define success and measure it

- “Culture eats strategy for breakfast” - this is where strategy is going to lose to culture if the organization lets it. (You see this with orgs trying to adopt OKRs, too).
- A focus on outcomes (and not output) from top to bottom - **with outcome measurement**

Audit/Review Projects

- Which of the ingredients for design success did the team have in place?
- Which design phases did the team have access to?
- Were there changes along the way to either of these things, and what happened?

Audit Examples

Case Study: Client Experience A

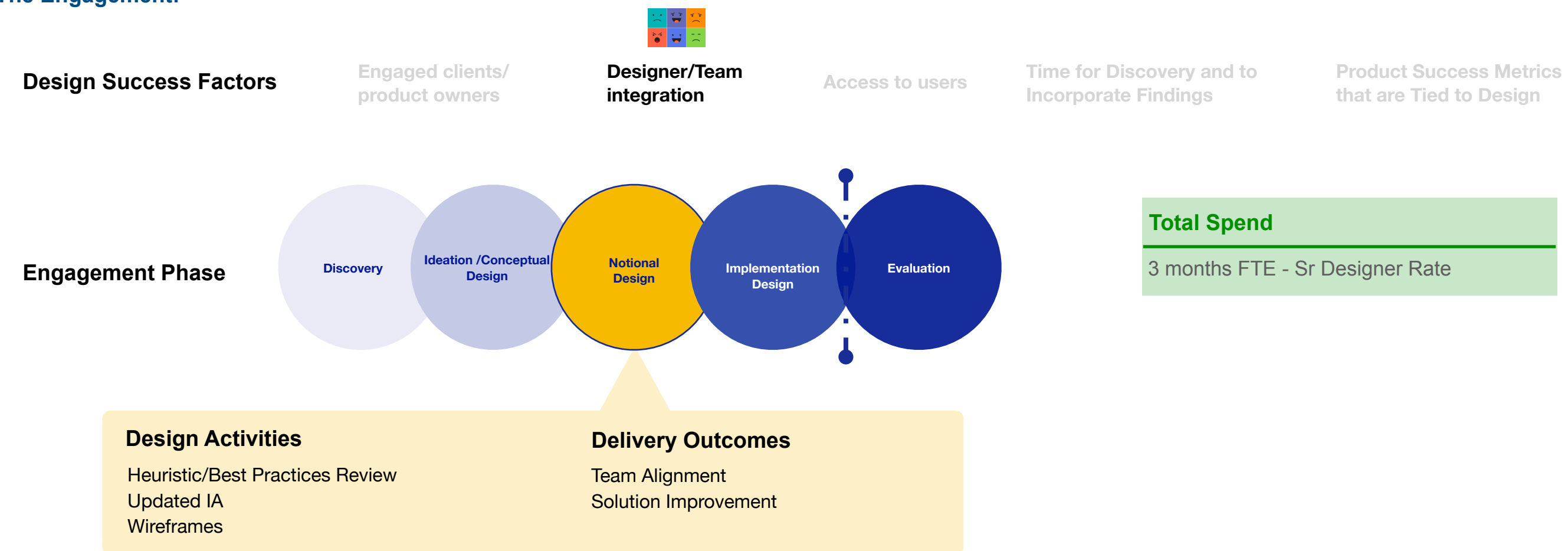
The product:

Phasellus vel augue vitae nisl tempor vestibulum. Nulla vitae augue tincidunt, cursus nisi sit amet, dignissim enim. Proin velit orci, venenatis eget dictum vel, mattis vitae arcu. Pellentesque id quam sit amet sapien maximus ullamcorper. Fusce faucibus nec felis vitae rutrum.

The challenge:

Pellentesque quis tincidunt purus, eu consequat nisi. Integer a dolor bibendum, sodales ante id, viverra metus. Donec ornare dui nisi, eu blandit enim cursus id. Mauris porttitor at erat sed maximus. Maecenas eget ipsum id neque mattis volutpat.

The Engagement:



The Outcome:

Taking an 'Expedience First' approach, we documented our assumptions, updated the information architecture (presentation and findability of information) of the site while keeping as light a touch as possible on the technical architecture and on the established HHS design system.

While our immediate client contact was happy with the direction and progress, we did not get buy-in from their upper leadership, who were anticipating a more visual approach to the redesign, and were vocal in their displeasure.

Note: While the client ended this phase unhappy, **they did arrive back at many of our initial suggestions following a period of greater project involvement and user feedback.**

Audit Examples

Case Study 2: Client Experience B

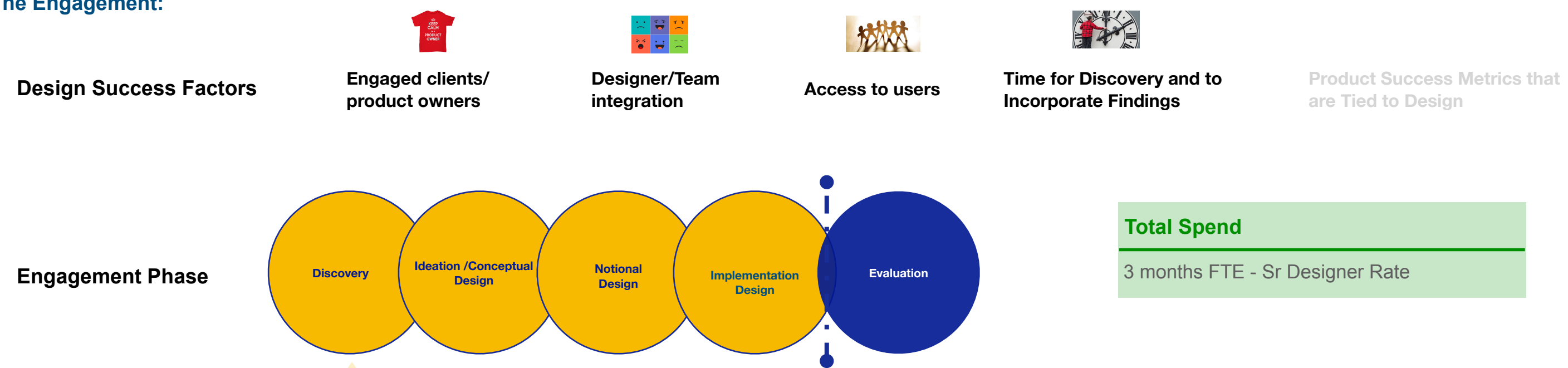
The product:

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas imperdiet vestibulum lacus, at vehicula nisl blandit id. Nam id tempus enim, eget aliquet orci. Quisque euismod purus ut ipsum convallis feugiat.

The challenge:

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Maecenas imperdiet vestibulum lacus, at vehicula nisl blandit id. Nam id tempus enim, eget aliquet orci. Quisque euismod purus ut ipsum convallis feugiat.

The Engagement:



Design Activities

Creative Design Definition (Discovery)

- Audience and Goals
- Inspiration

Visual Design Ideation

- Initial Design Concepts
- Design Gallery/Voting
- Refinement

User Testing (Notional Design)

- Impression Testing

Implementation Design

- Design Reviews

Delivery Outcomes

- Problem Definition
- Solution Ideation
- Solution Validation
- Solution Improvement
- Team Alignment
- Client Buy-In

The Outcome:

With the same team and a similar timeframe*, we were able to translate greater client engagement, a willingness to spend some time in Discovery and problem definition tasks, and user availability into a first release that represented a bigger forward step for the product, and had much more buy-in from the client and their leadership.

**Our second 3 months at least benefited from the fact that the team was more familiar with one another, and that we had already done some documentation-based product discovery. Our velocity for the second three months far outpaced our velocity in the first three months. If we had started from ground zero, my hunch is that this may have taken 3-4 extra weeks.*

Thank You TorChi!

Let me know what your experience has been with Agile and HCD

Discuss design ethics and where we go from here

brad@humancenteredagile.com