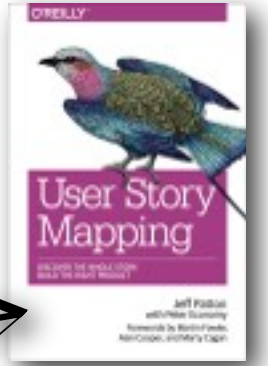
Story Mapping

Don’t Lose the Big Picture

Jeff Patton
jeff@jpattonassociates.com
twitter: @jeffpatton

I wrote this book!



A story map is a simple way to tell a story and break it down into parts

Stories are meant to solve 2 problems

and neither is about writing better requirements

Written requirements don't work the way you think they do

Imagine a simple phone conversation...

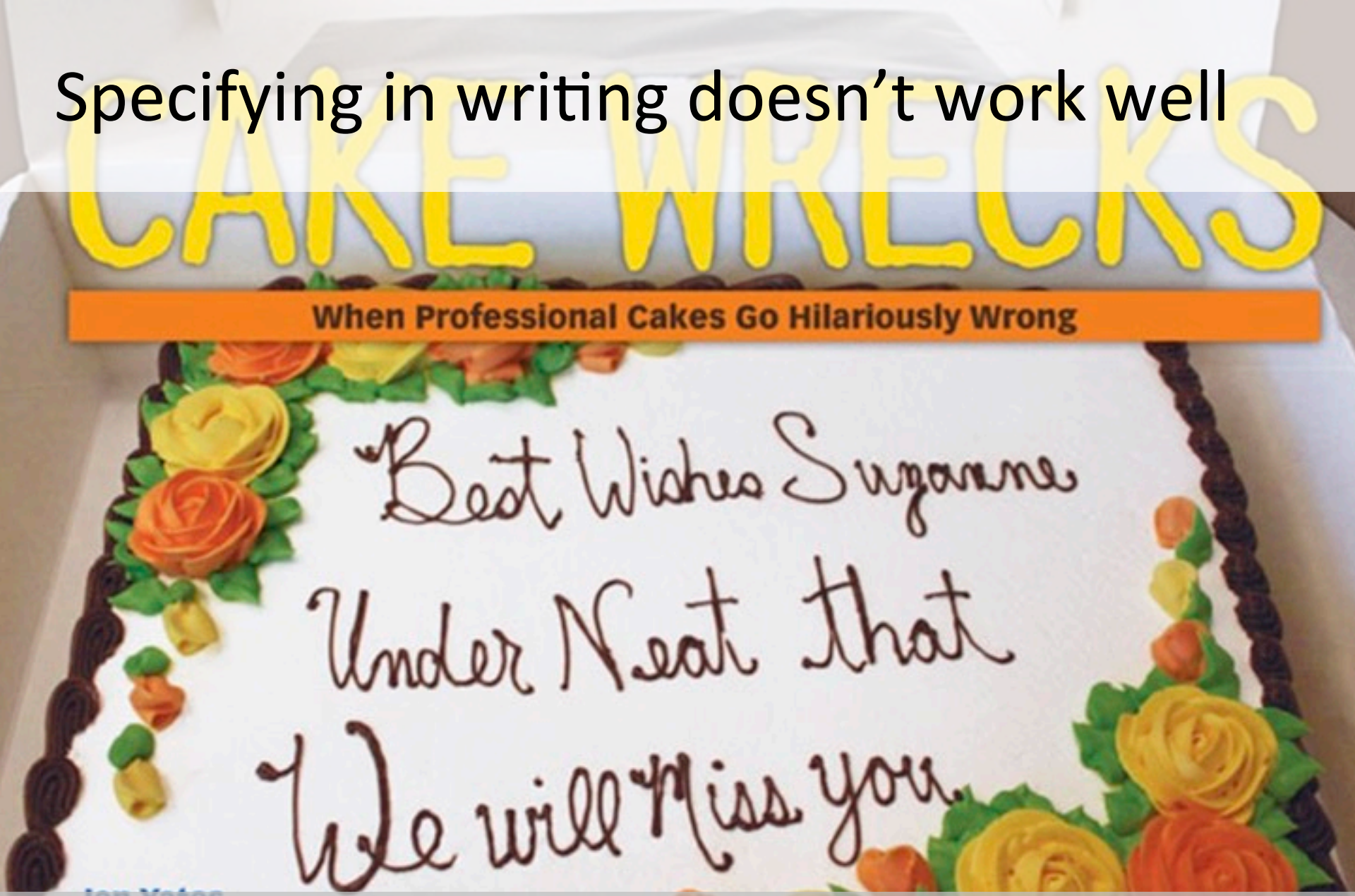
Jen Yates' Cake Wrecks: www.cakewrecks.com

Specifying in writing doesn't work well

http://www.cakewrecks.com/

Cake Wrecks, book by Jen Yates,

Specifying in writing doesn't work well



Jen Yates' Cake Wrecks: www.cakewrecks.com

Specifying in writing doesn't work well

Jen Yates' Cake Wrecks: www.cakewrecks.com

Specifying in writing doesn't work well

Jen Yates' Cake Wrecks: www.cakewrecks.com

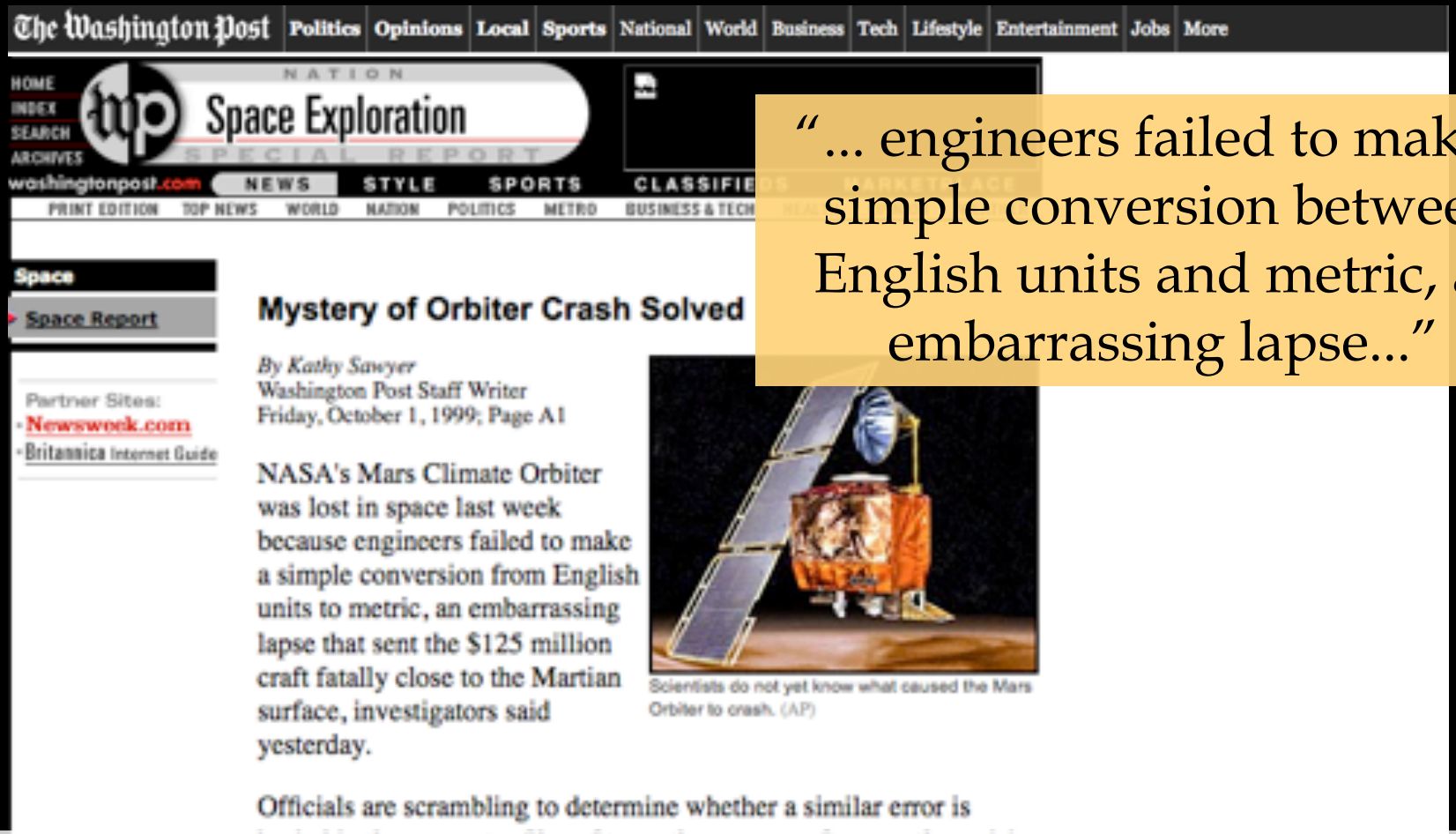
Specifying in writing doesn't work well

Jen Yates' Cake Wrecks: www.cakewrecks.com

Specifying in writing doesn't work well

Jen Yates' Cake Wrecks: www.cakewrecks.com

Specifying in writing doesn't work well



The Washington Post | Politics | Opinions | Local | Sports | National | World | Business | Tech | Lifestyle | Entertainment | Jobs | More

HOME INDEX SEARCH ARCHIVES

NATION

Space Exploration

SPECIAL REPORT

washingtonpost.com NEWS STYLE SPORTS CLASSIFIEDS MARKETPLACE

PRINT EDITION TOP NEWS WORLD NATION POLITICS METRO BUSINESS & TECH

Space

Space Report

Partner Sites:
· Newsweek.com
· Britannica Internet Guide

Mystery of Orbiter Crash Solved

By Kathy Sawyer
Washington Post Staff Writer
Friday, October 1, 1999; Page A1

NASA's Mars Climate Orbiter was lost in space last week because engineers failed to make a simple conversion from English units to metric, an embarrassing lapse that sent the $125 million craft fatally close to the Martian surface, investigators said yesterday.

Officials are scrambling to determine whether a similar error is buried in the computer files of two other spacecraft currently cruising through space: the Mars Polar Lander, scheduled to hit the Martian

It now appears the error had affected the orbiter mission from its

Scientists do not yet know what caused the Mars Orbiter to crash. (AP)

"… engineers failed to make a simple conversion between English units and metric, an embarrassing lapse…"

Sometimes mistakes are less funny

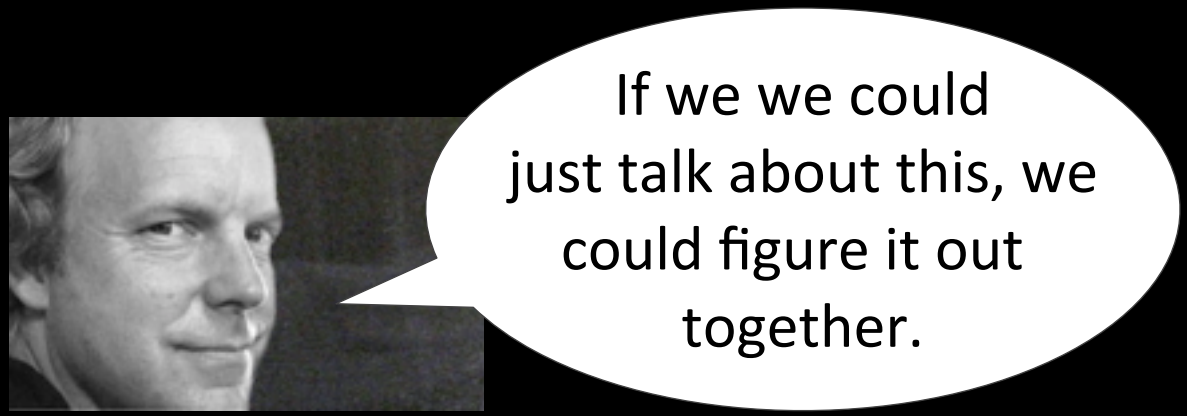
When we share and sign off a document we may believe we understand

Kent has a disruptively simple idea

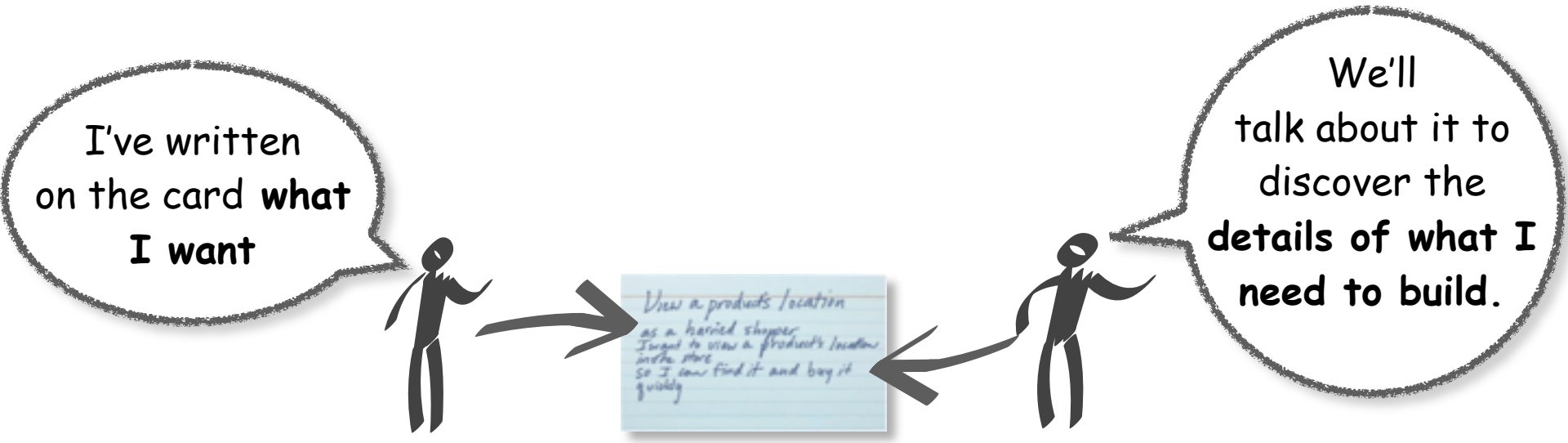
Stop it.

Stop exchanging documents.

Tell me your story.



The original idea of a story was simple: use it to facilitate a conversation



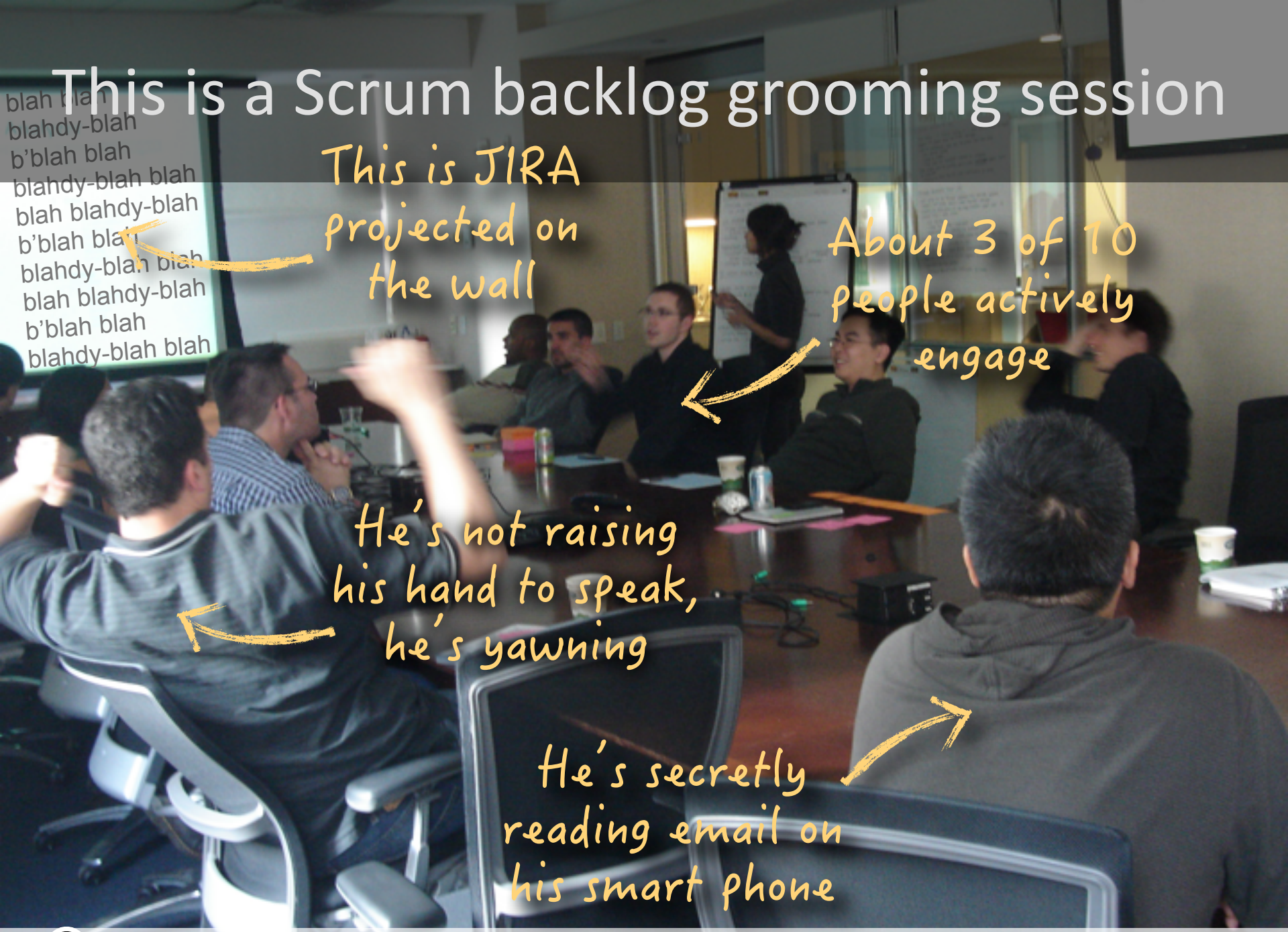
someone who wants something

someone who builds something

Stories get their name from how we use them, not how we write them.

But, we still managed to screw that up

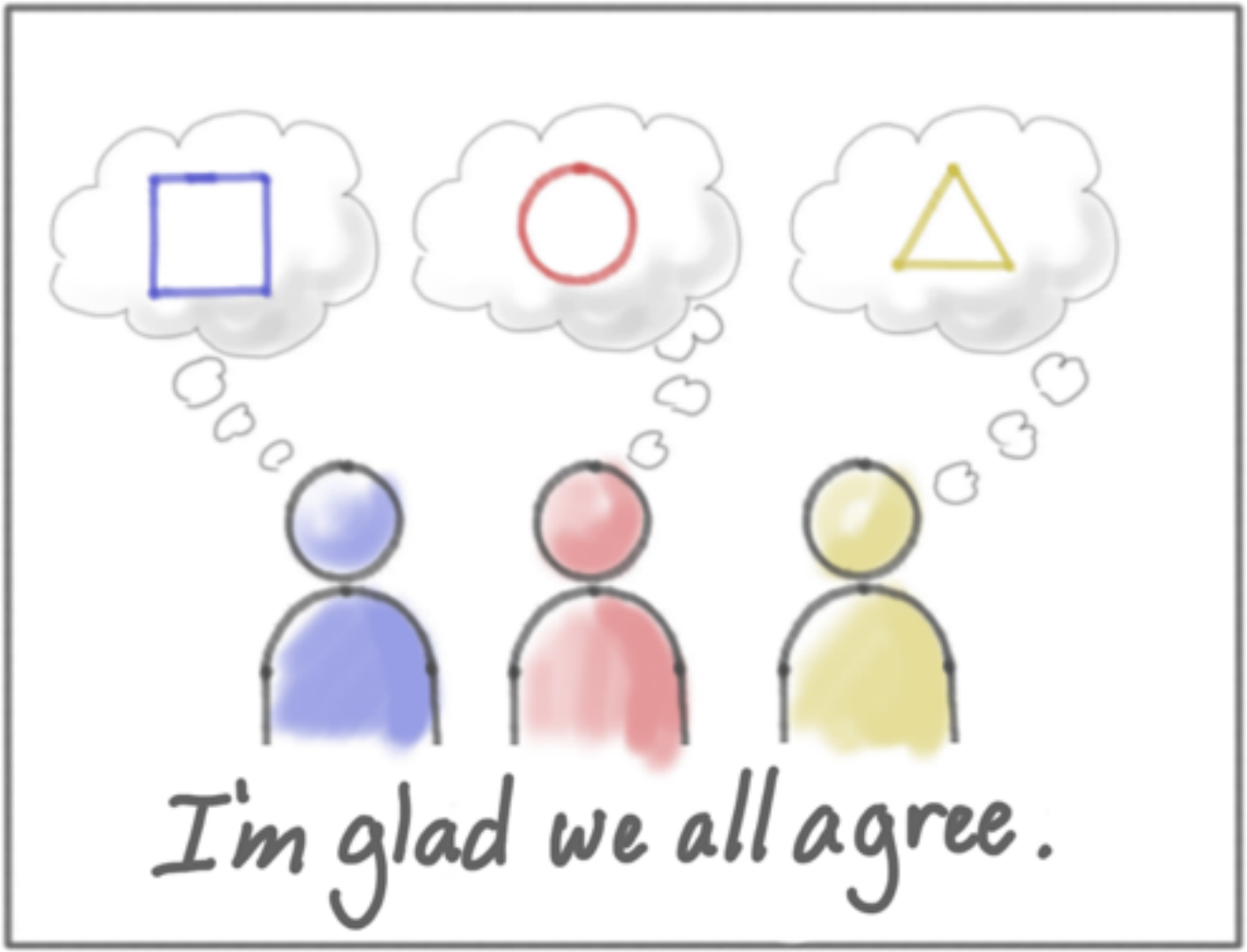
This is a Scrum backlog grooming session



This isn't the kind of conversation Kent had in mind

Something special is going on during an effective conversation

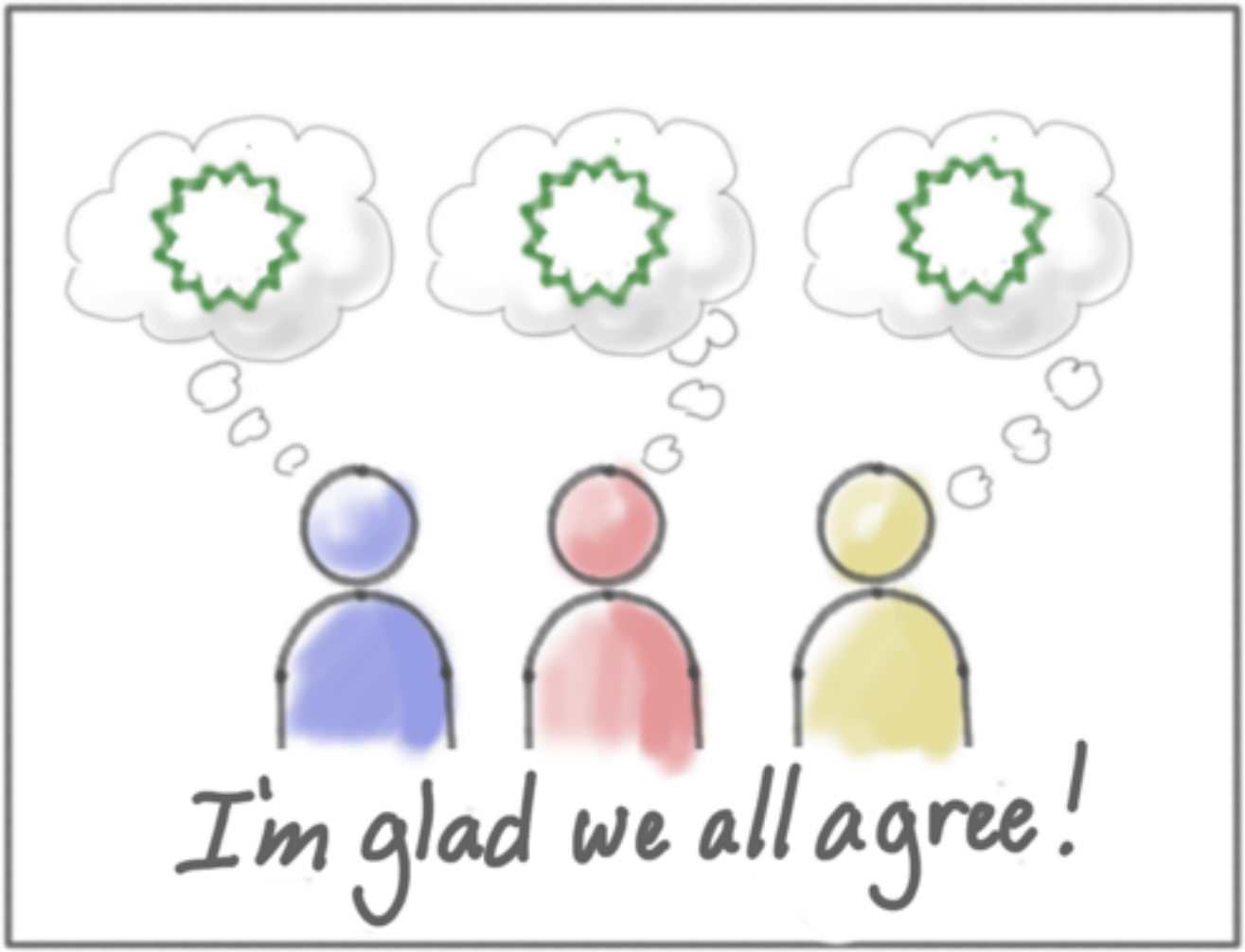
With a shallow discussion, we may all take away something different



When we **externalize our thinking** with words and pictures, we detect differences

When we combine and refine, we arrive at something better

Afterwards, when we say the same thing, we actually mean it

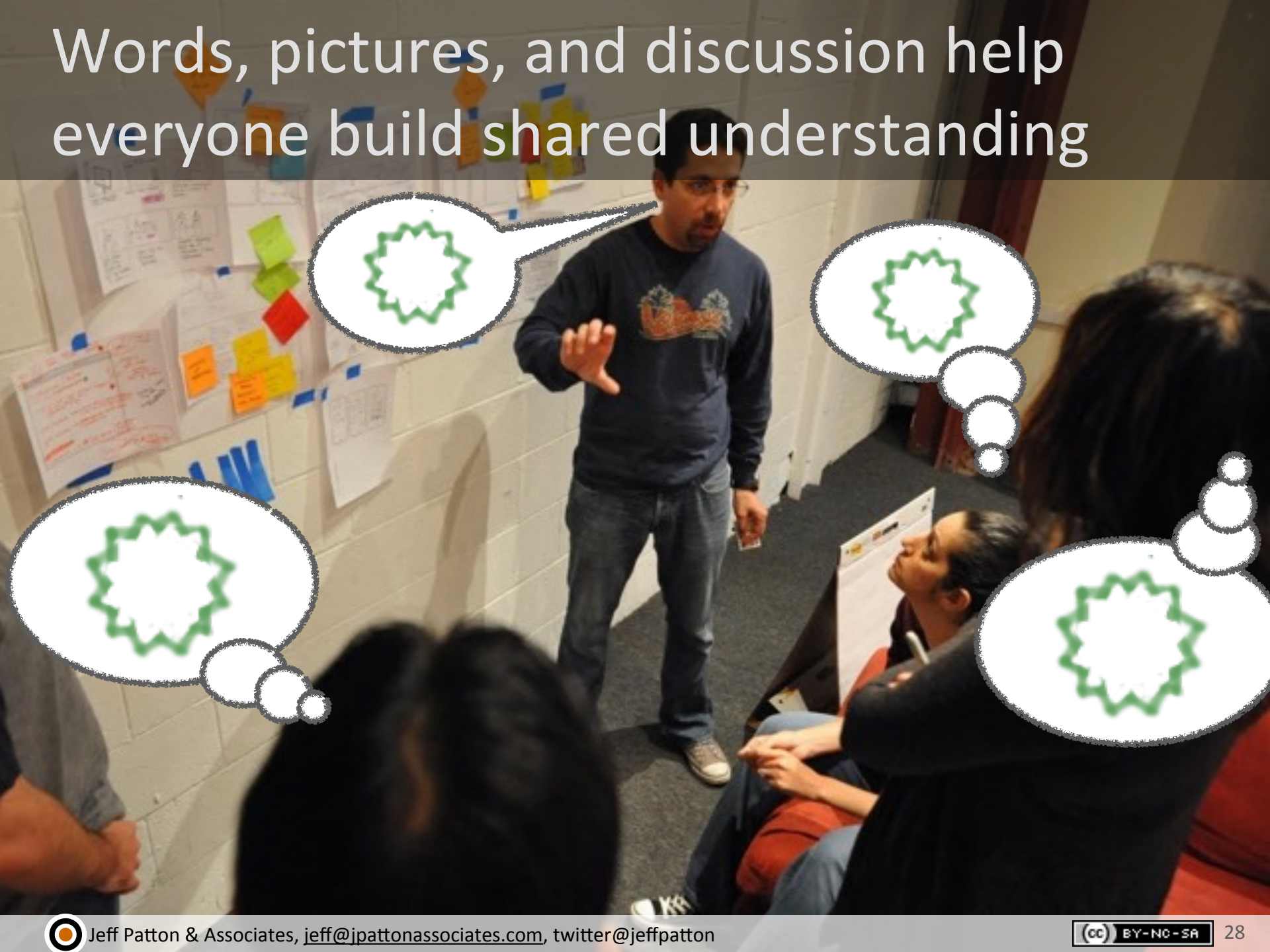


Shared understanding and **alignment** are the objectives of collaborative work

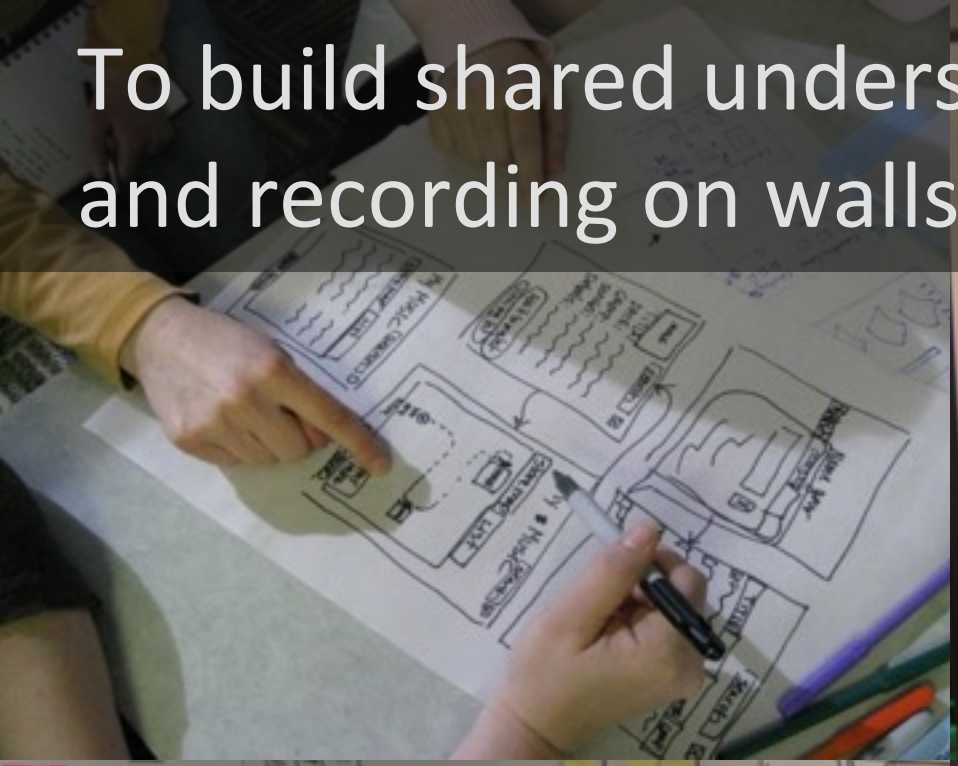
✳ Credit for this illustration goes to ThoughtWorks' Luke Barret. Jeff Patton drew these illustrations based on Luke's. Luke doesn't recall where he first saw this cartoon.

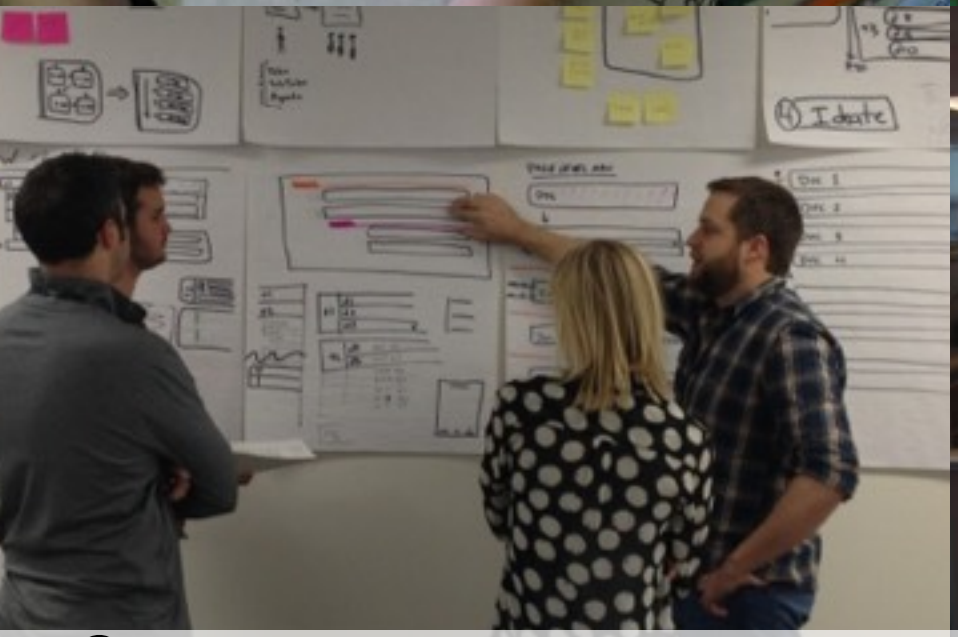
Shared documents aren't shared understanding

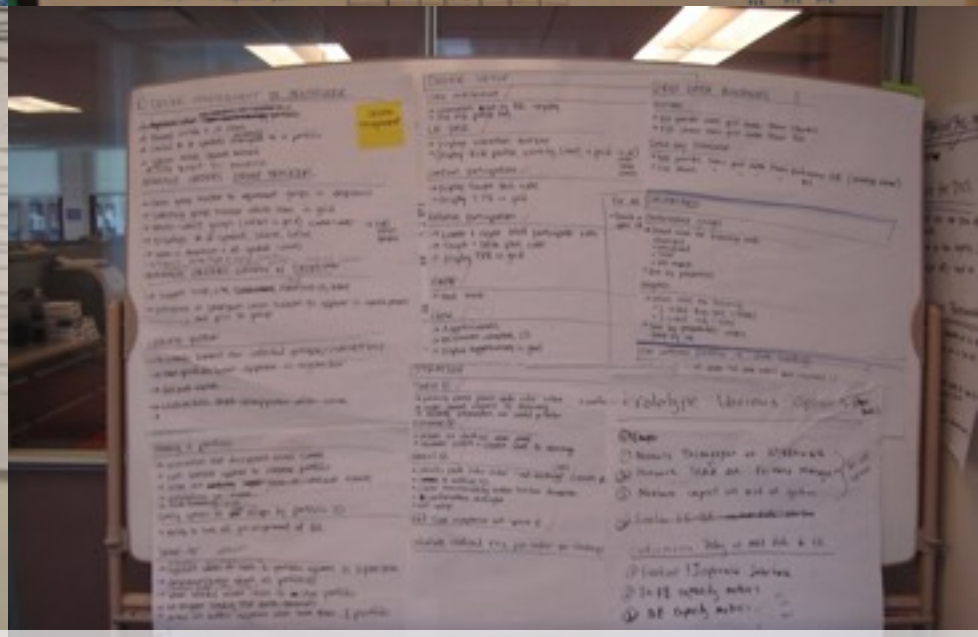
Words, pictures, and discussion help everyone build shared understanding



To build shared understanding, use sketching and recording on walls and whiteboards





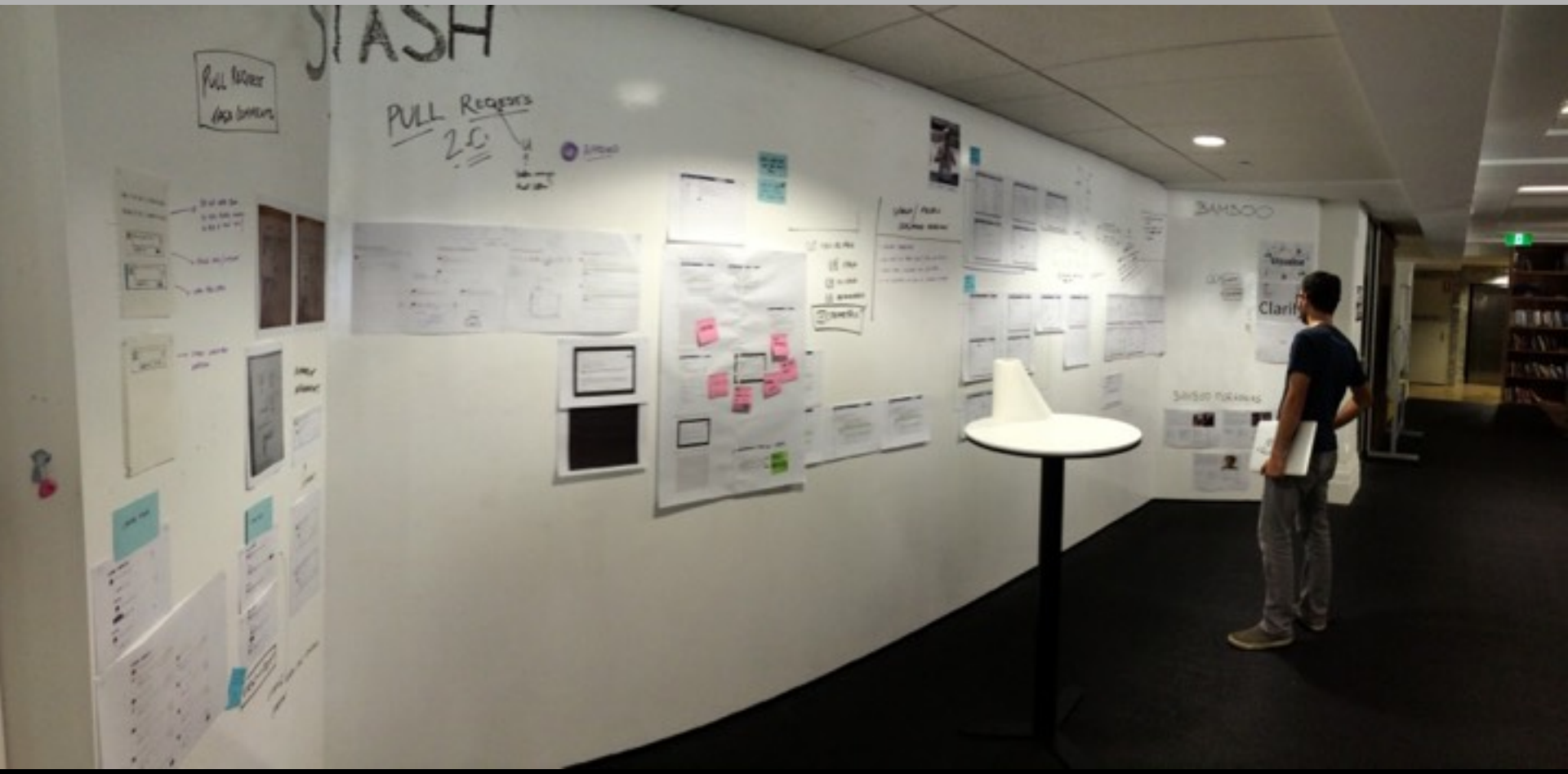


Use words and pictures

sketch,
tell stories,
write down
facts & decisions

Shared Understanding and collaboration at Atlassian

Shared Understanding and collaboration at Atlassian



A different kind of stand-up meeting at Atlassian

What you record during conversations works like a vacation photo

Looking at it helps you remember details that aren't in the photo

What you record during conversations works like a vacation photo

Looking at it helps you remember details that aren't in the photo

Effective story conversations

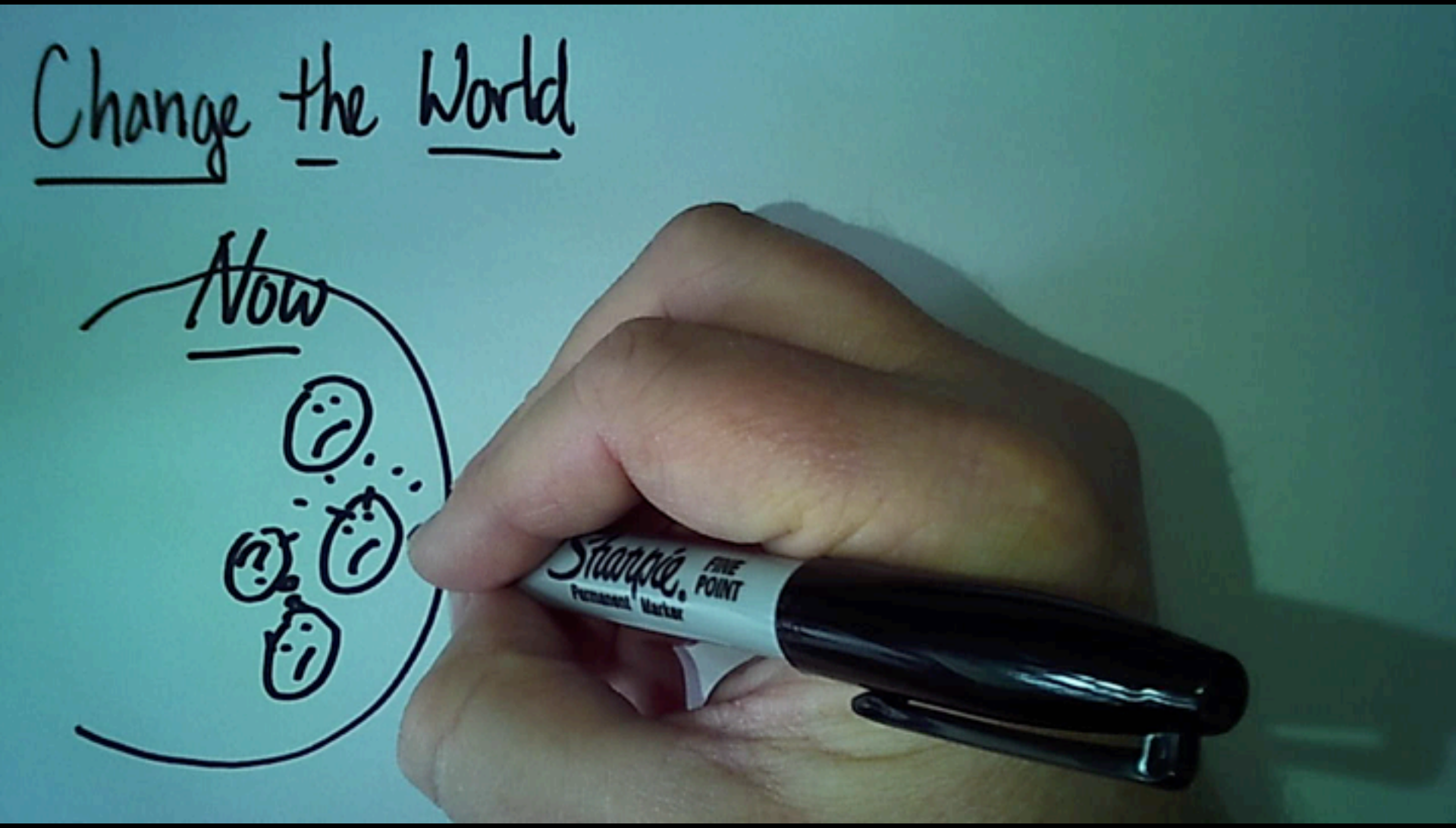
build shared understanding

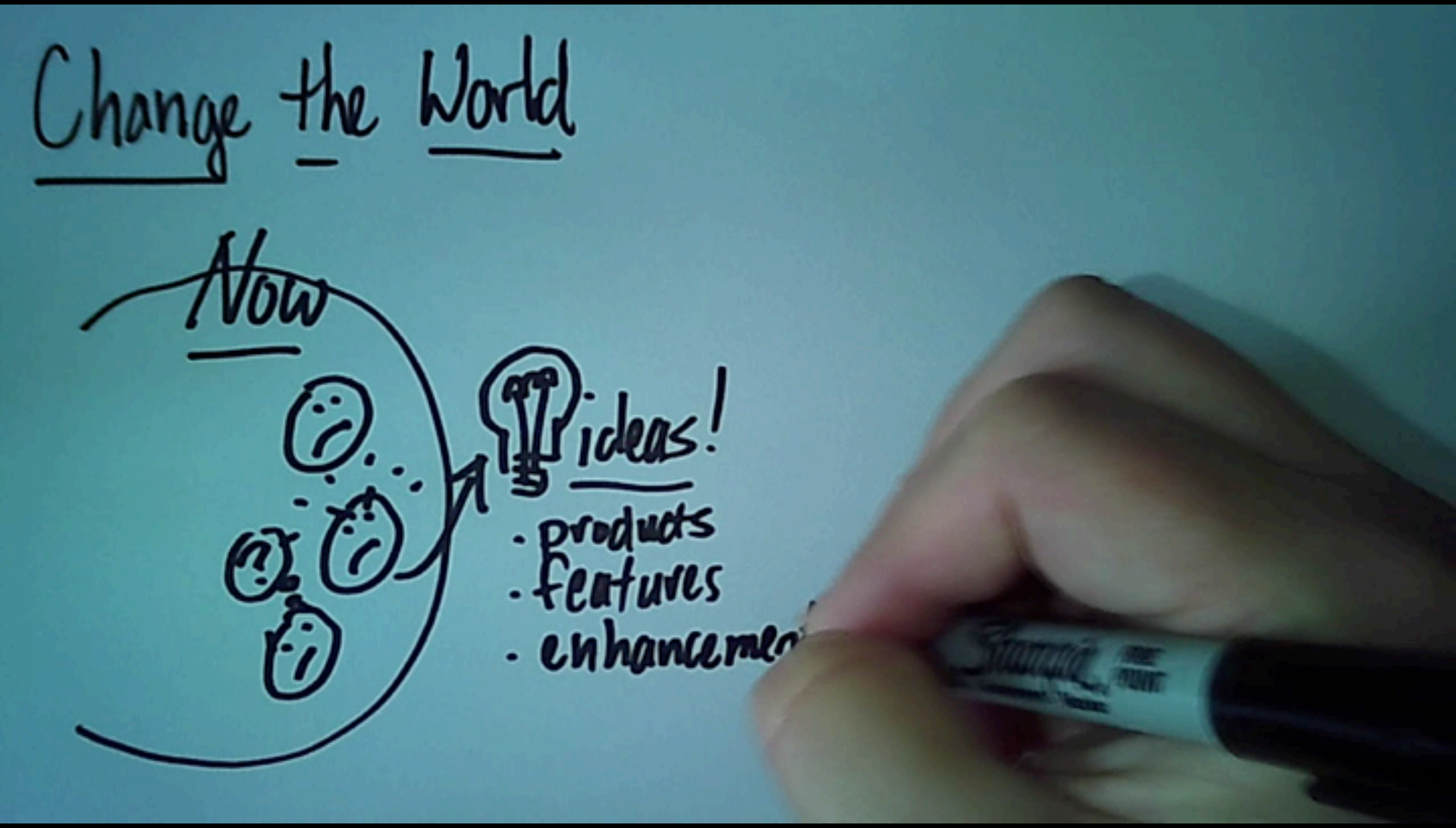
The best **documents** use words and pictures to **help recall our conversations**, they don't replace conversations

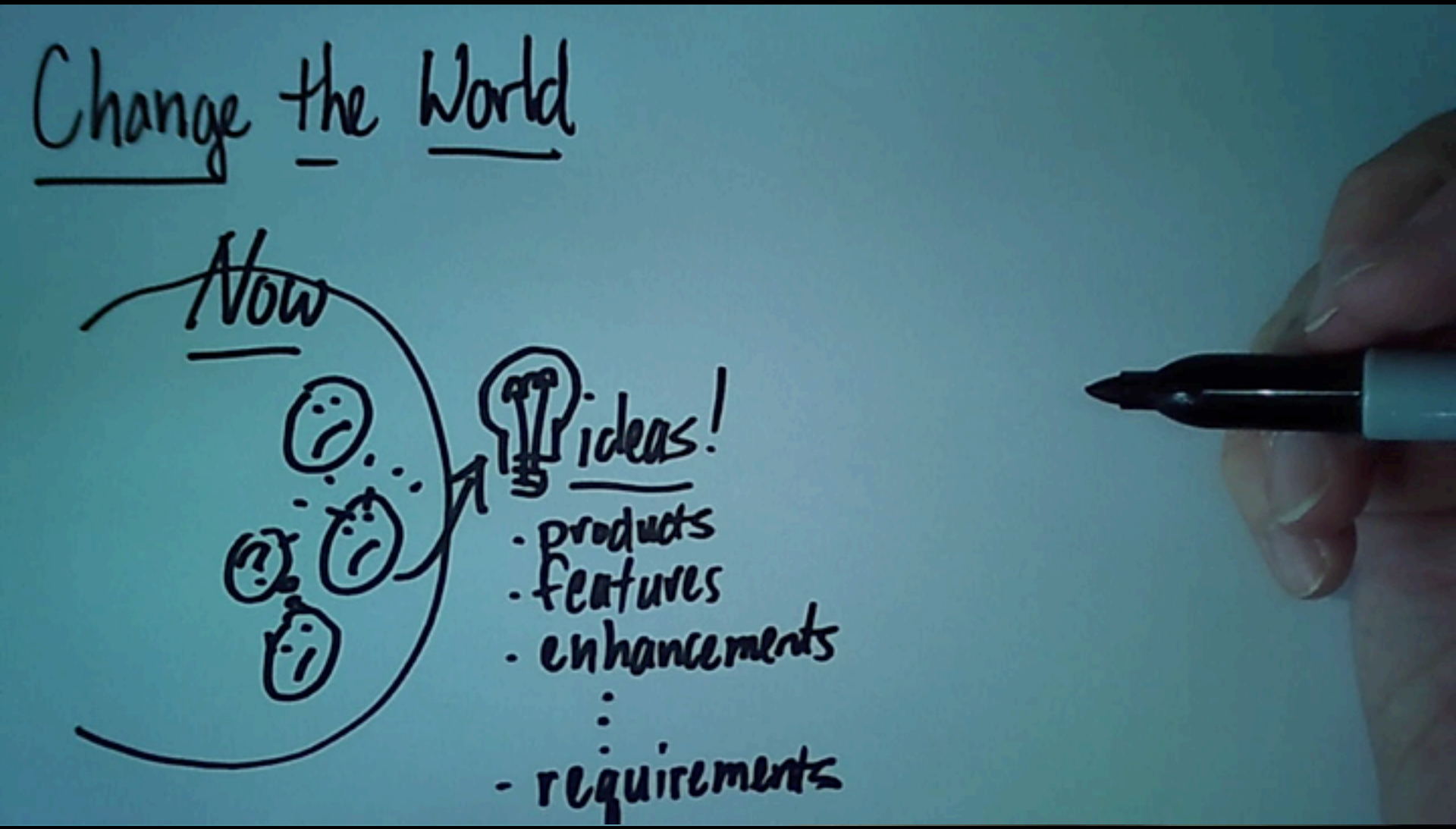
Stories are meant to solve 2 problems:

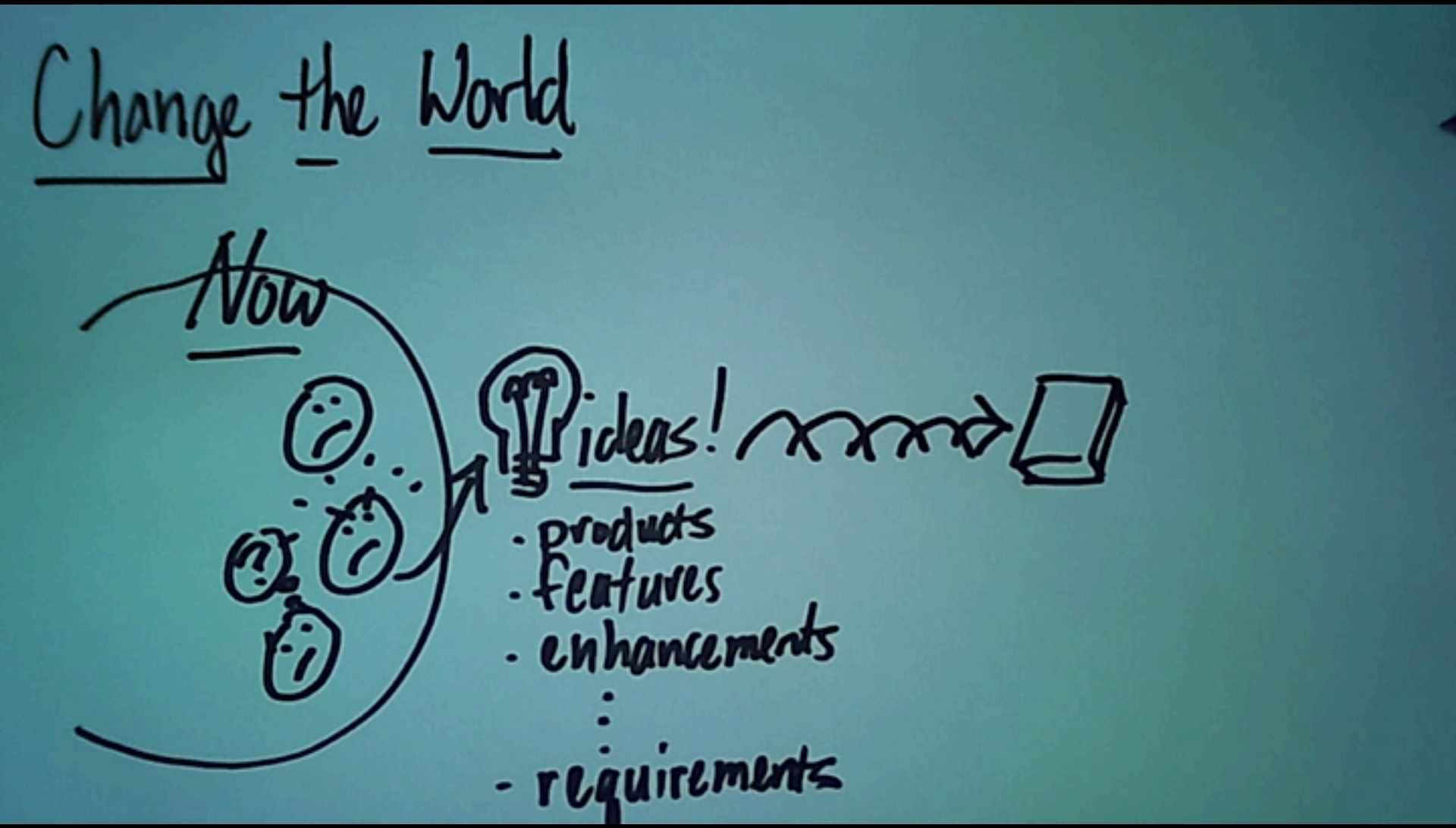
1. Building shared-understanding through story telling

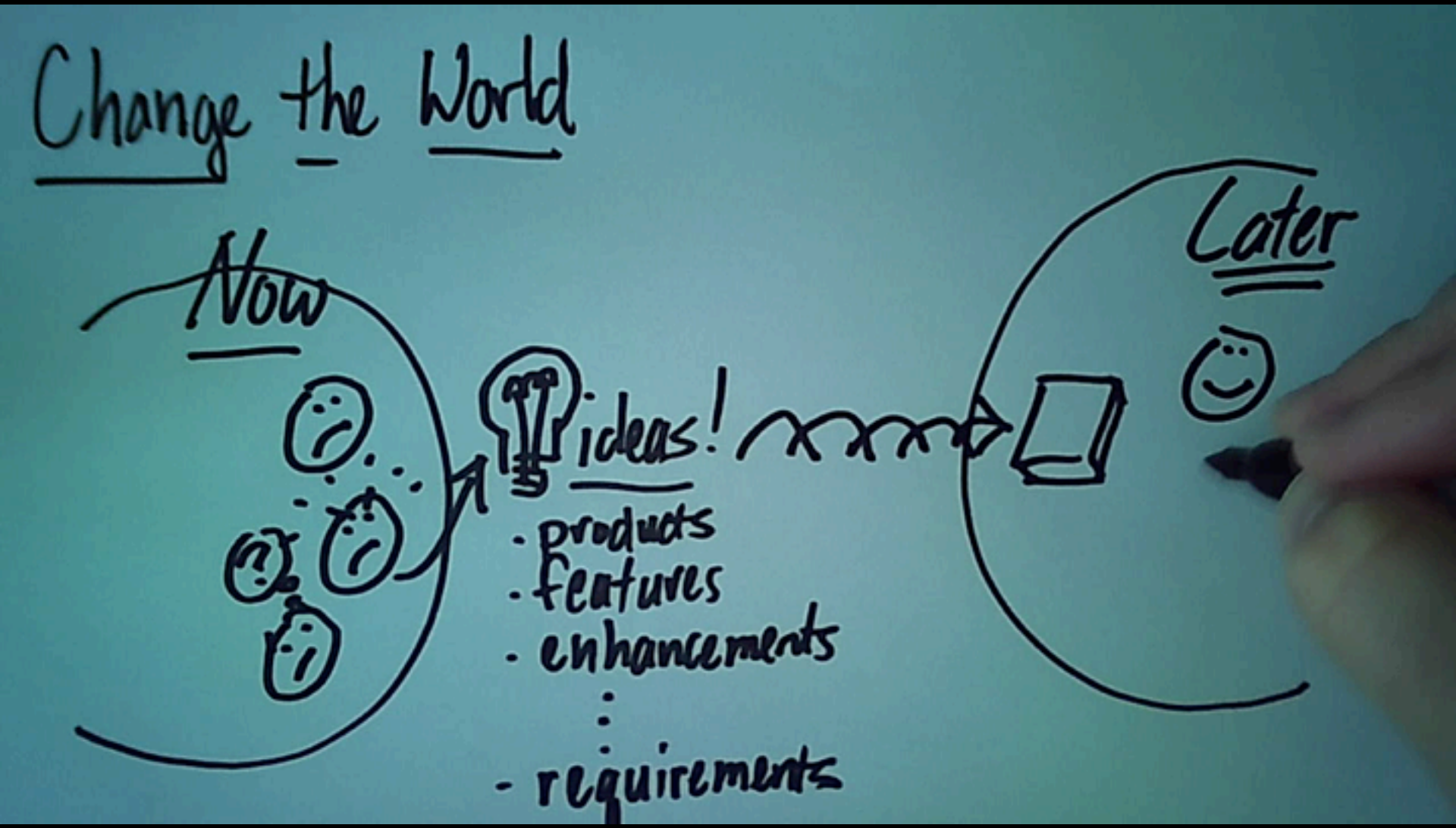
Your job isn't really to build software

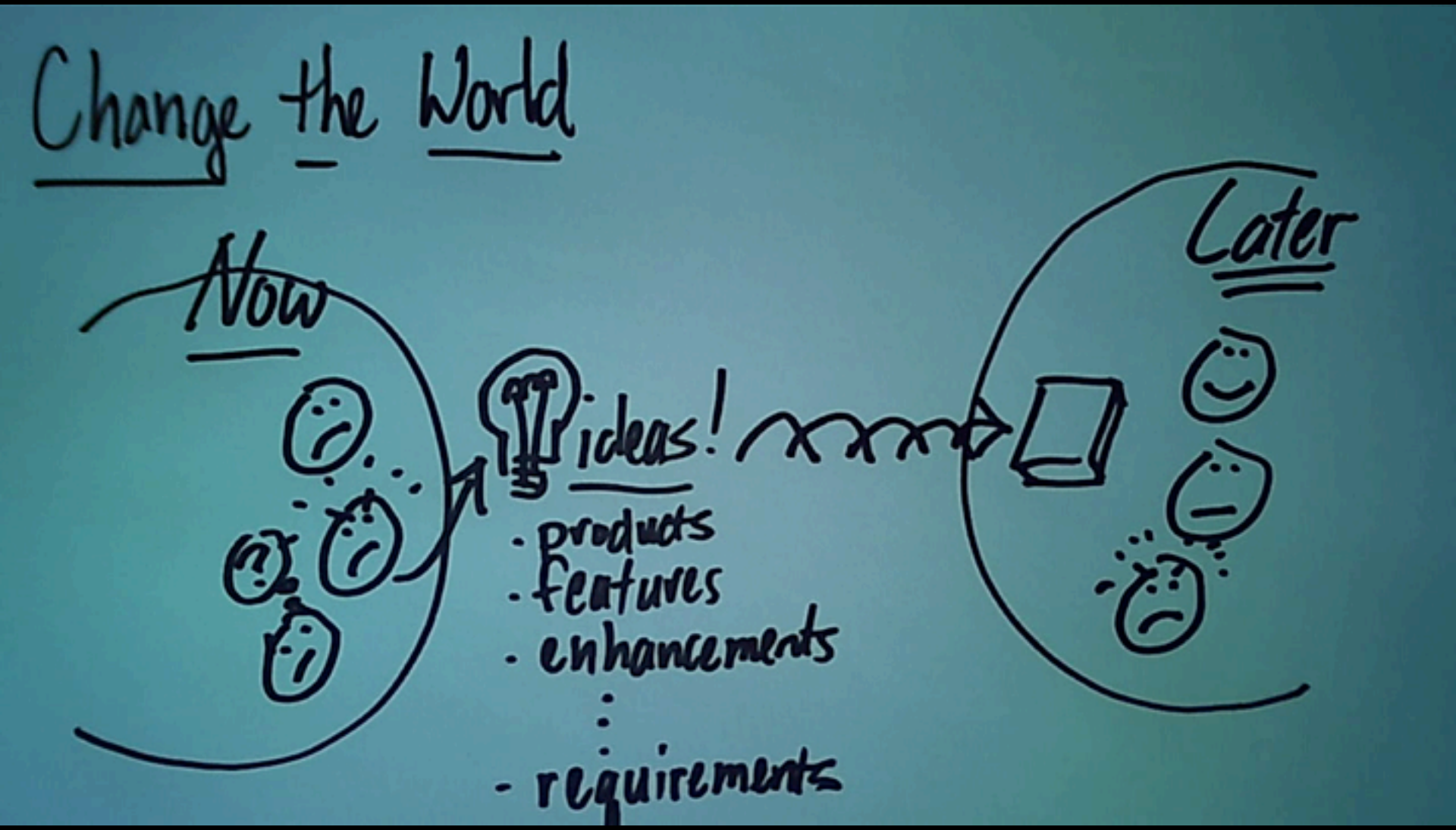


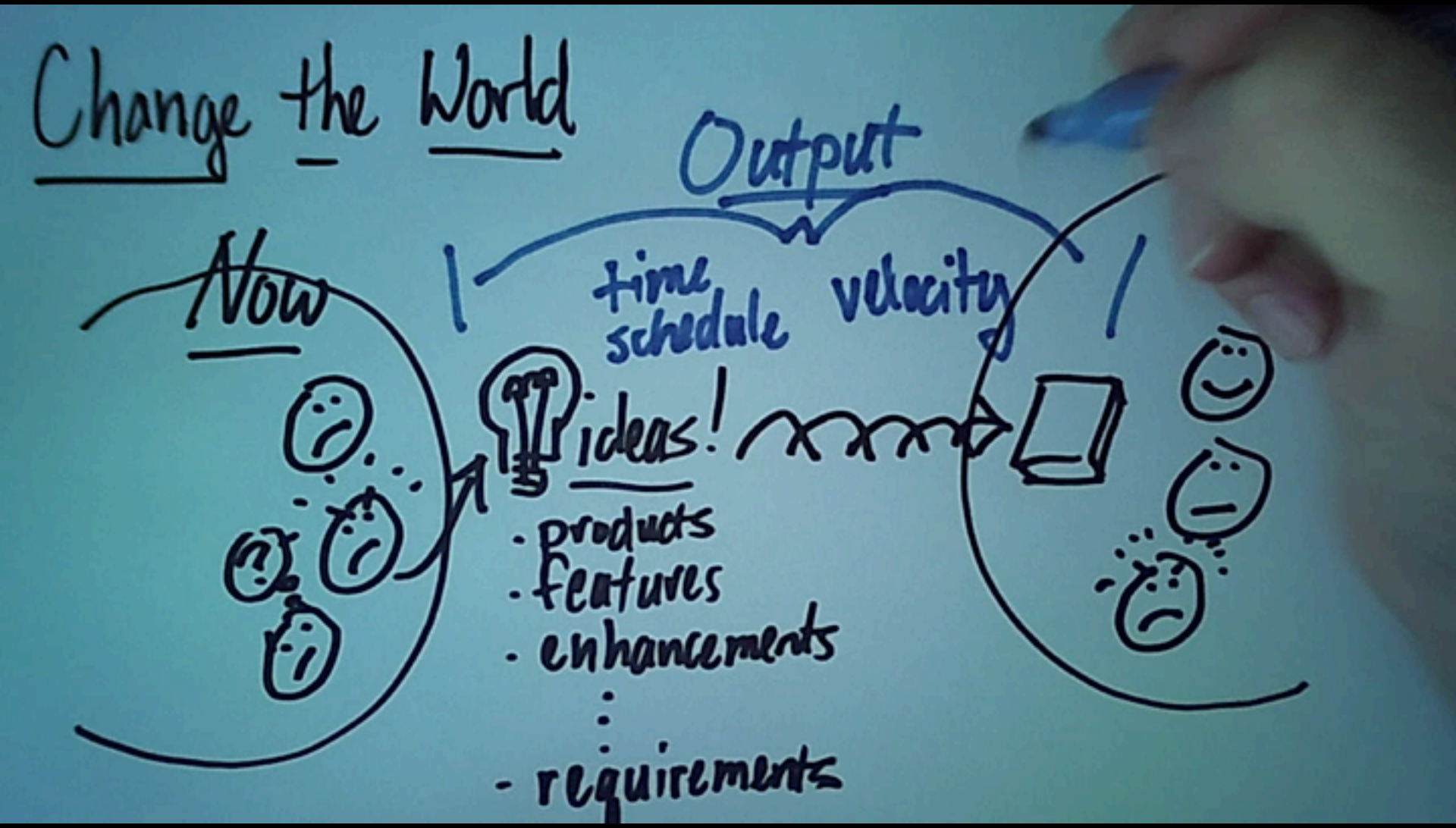


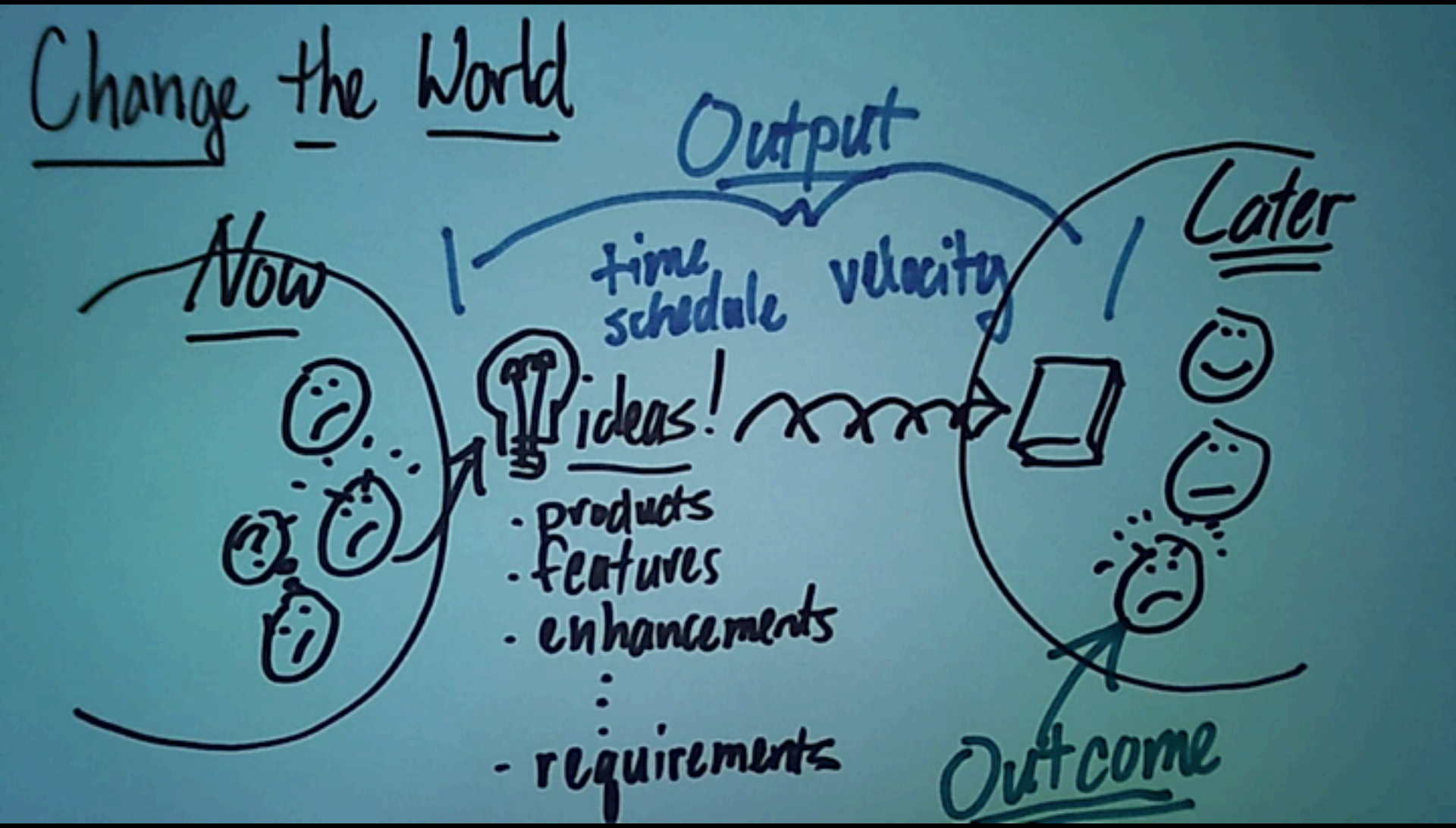

Change the World
Now
ideas!
- products
- features
- enhancements
:
- requirements

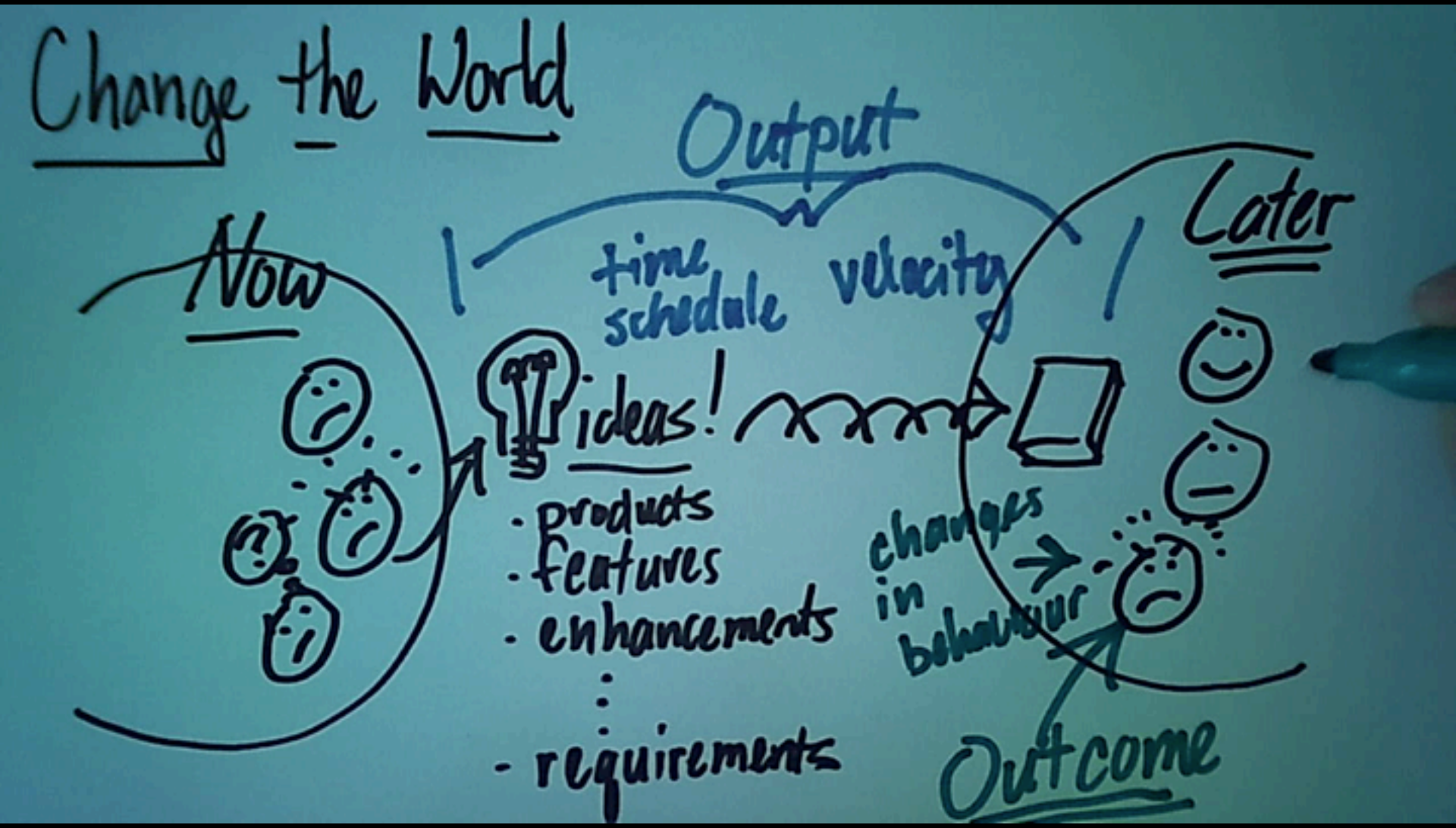

Change the World
Now
ideas!
- products
- features
- enhancements
:
- requirements

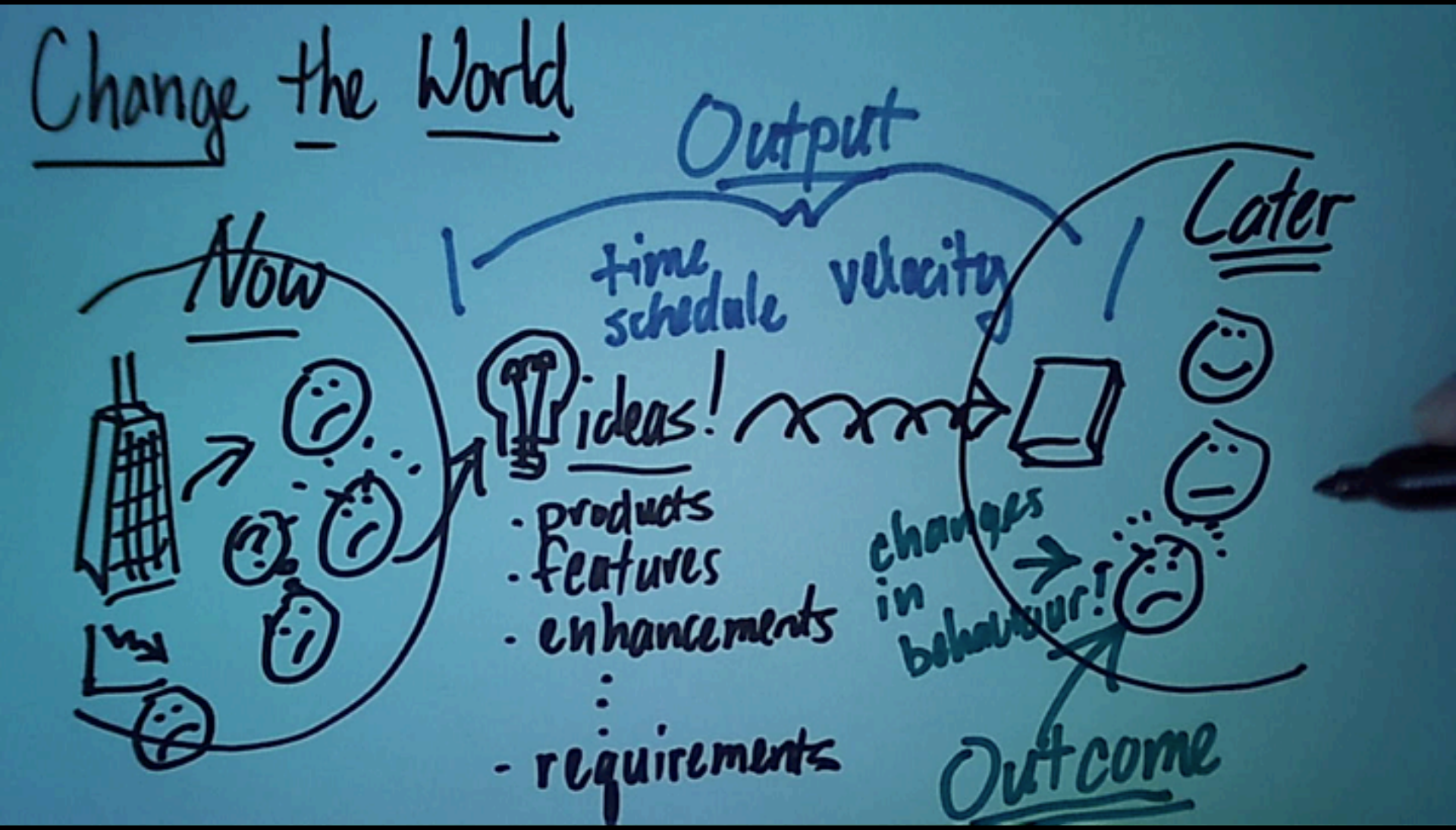

Change the World
Now
Later
ideas!
- products
- features
- enhancements
:
- requirements

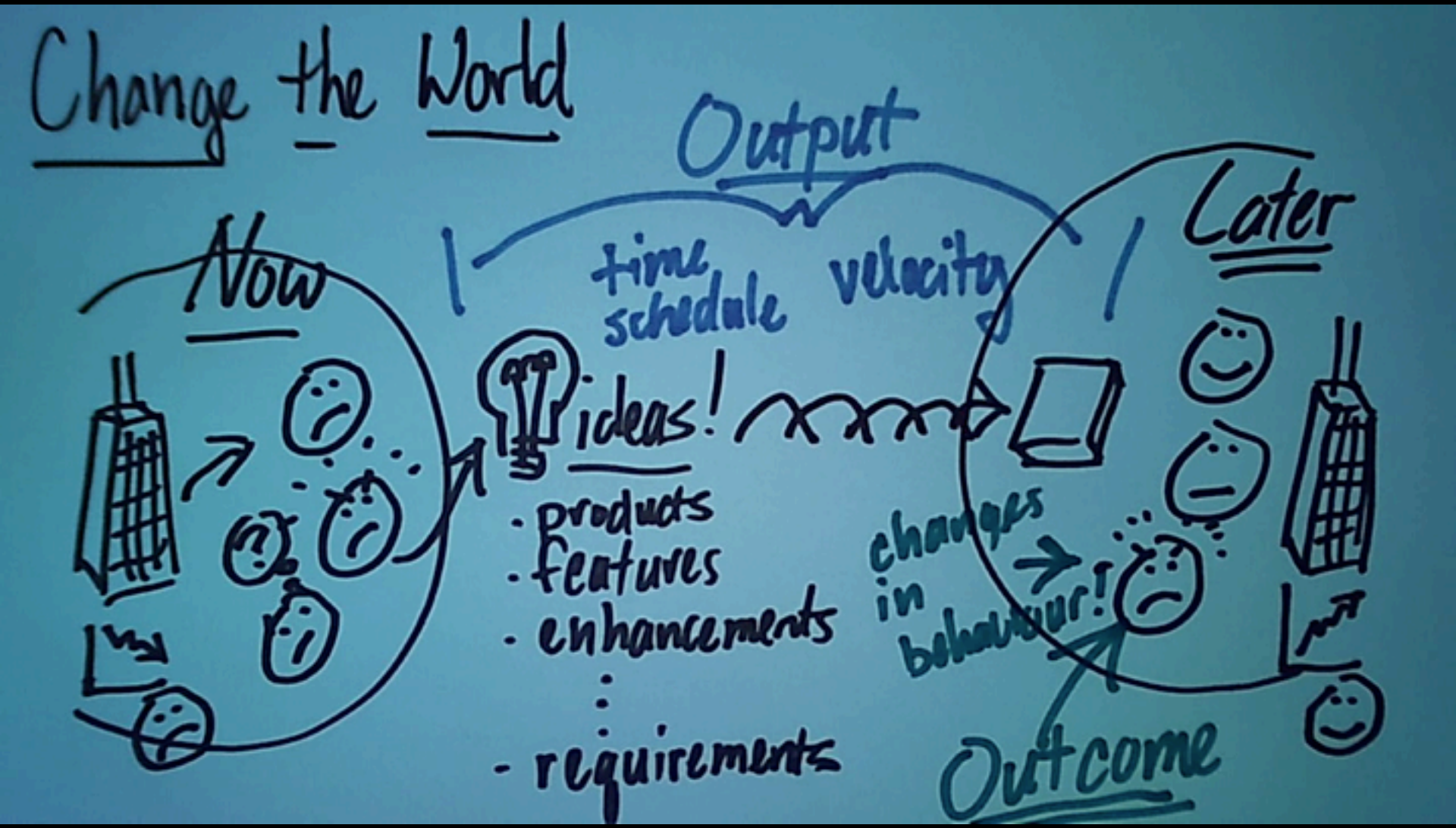

Change the World
Now
Later
ideas!
- products
- features
- enhancements
:
- requirements

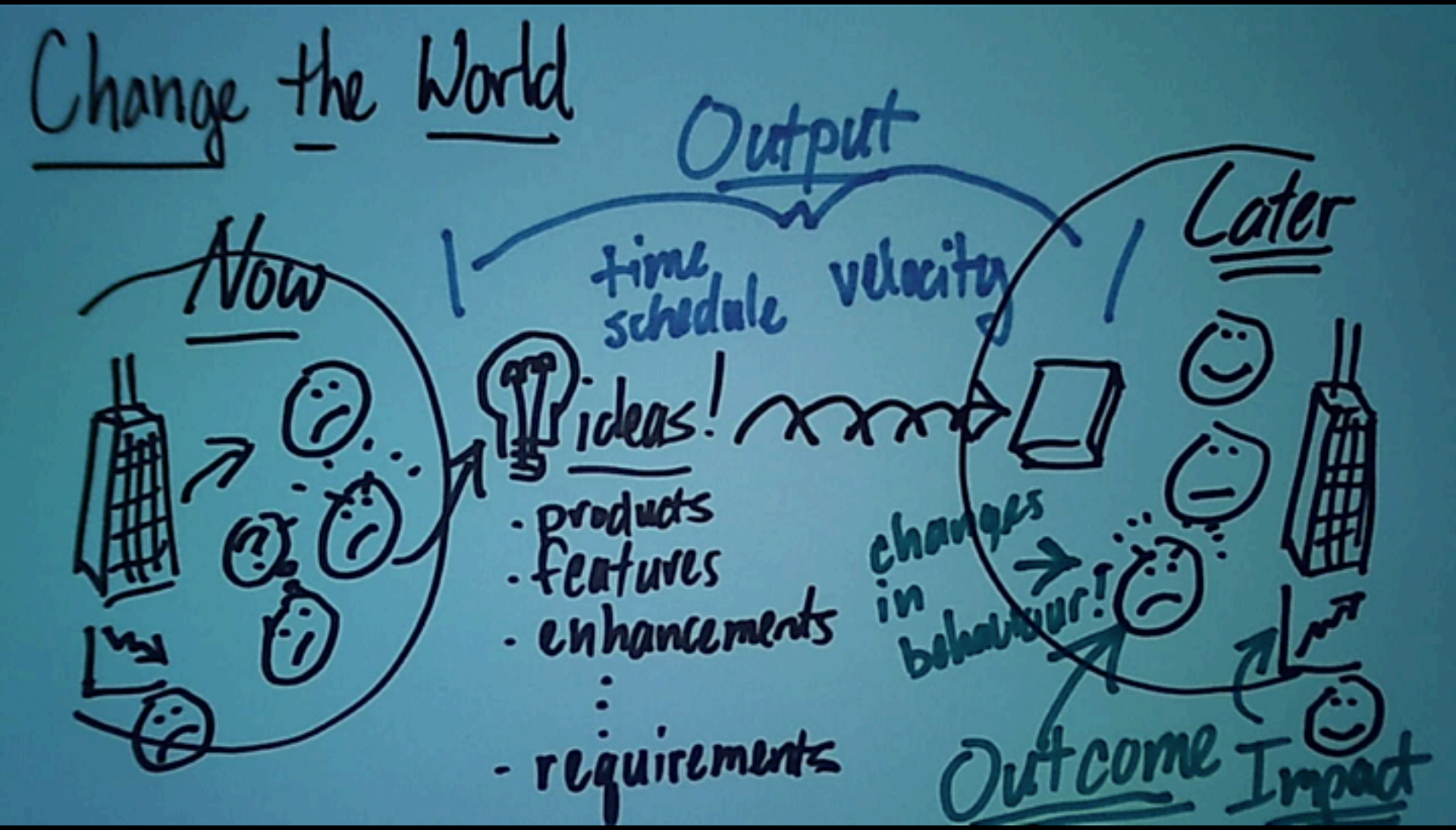

Change the World
Output
Now
time
schedule
velocity
ideas!
- products
- features
- enhancements
:
- requirements

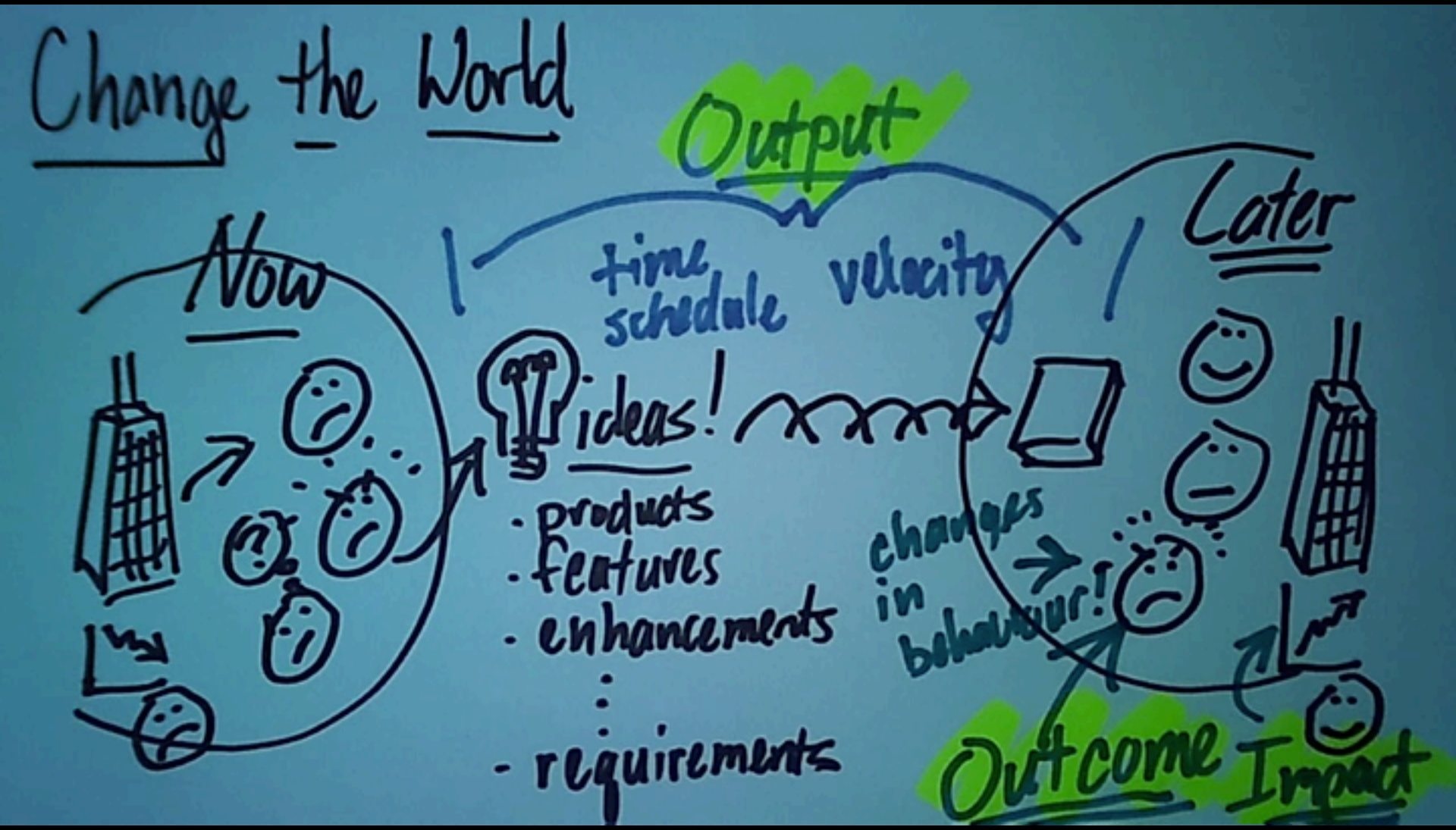

Change the World
Output
Now
time
schedule
velocity
Later
ideas!
- products
- features
- enhancements
:
- requirements
Outcome

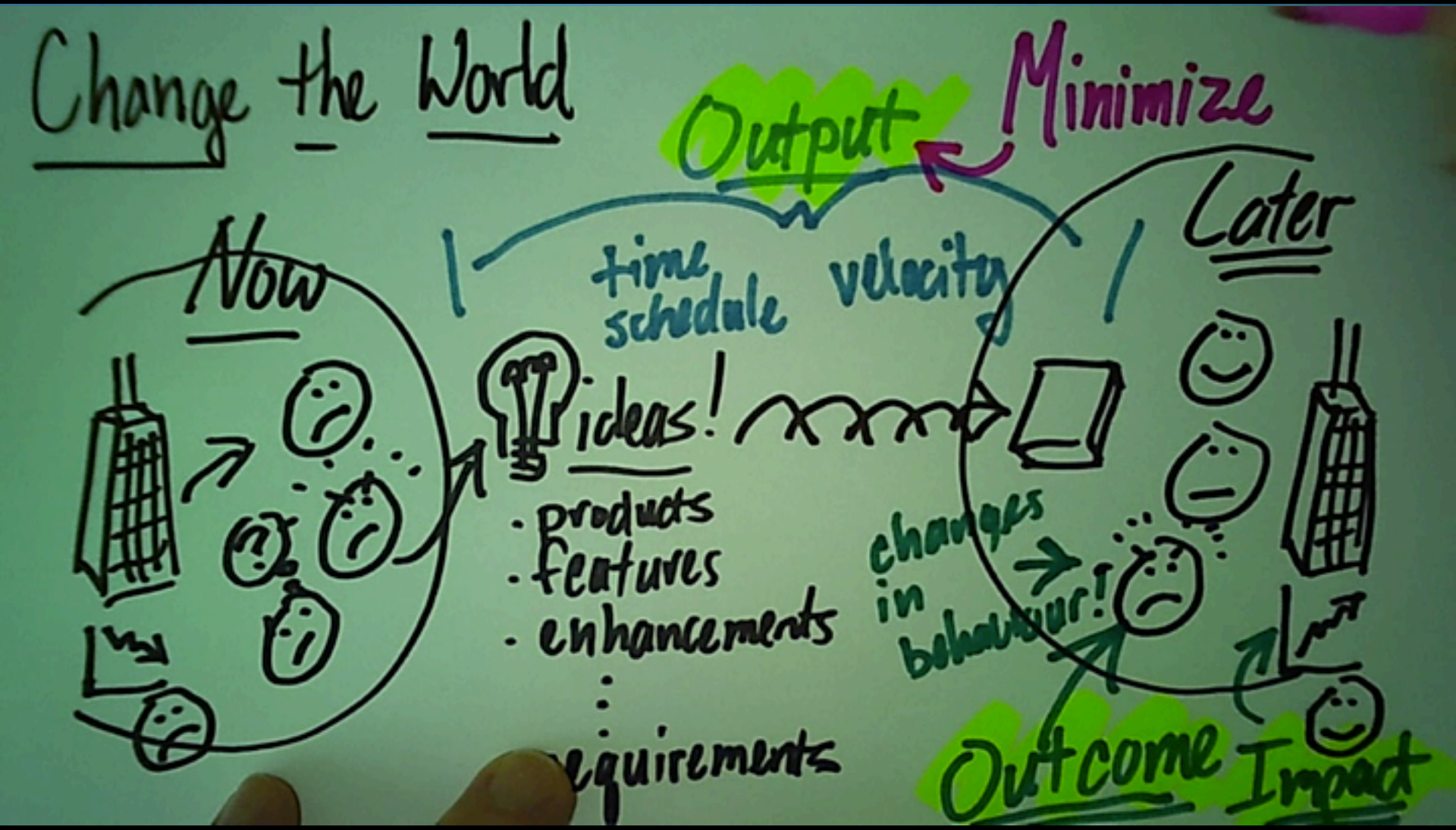

Change the World
Output
Now
time
schedule
velocity
Later
ideas!
- products
- features
- enhancements
- requirements
changes
in
behaviour
Outcome

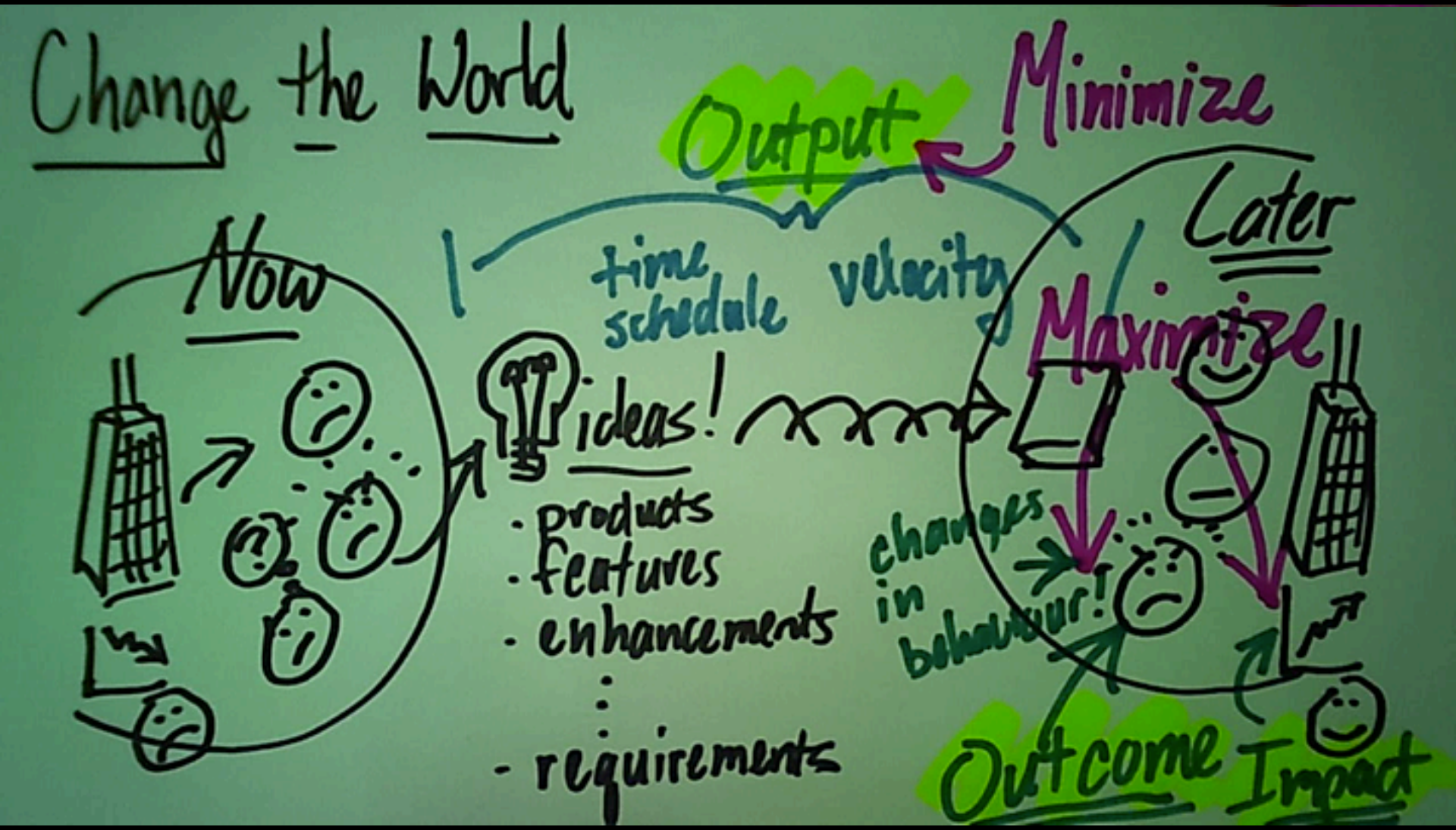

Change the World
Output
Now
time
schedule
velocity
Later
ideas!
- products
- features
- enhancements
- requirements
changes in behaviour!
Outcome

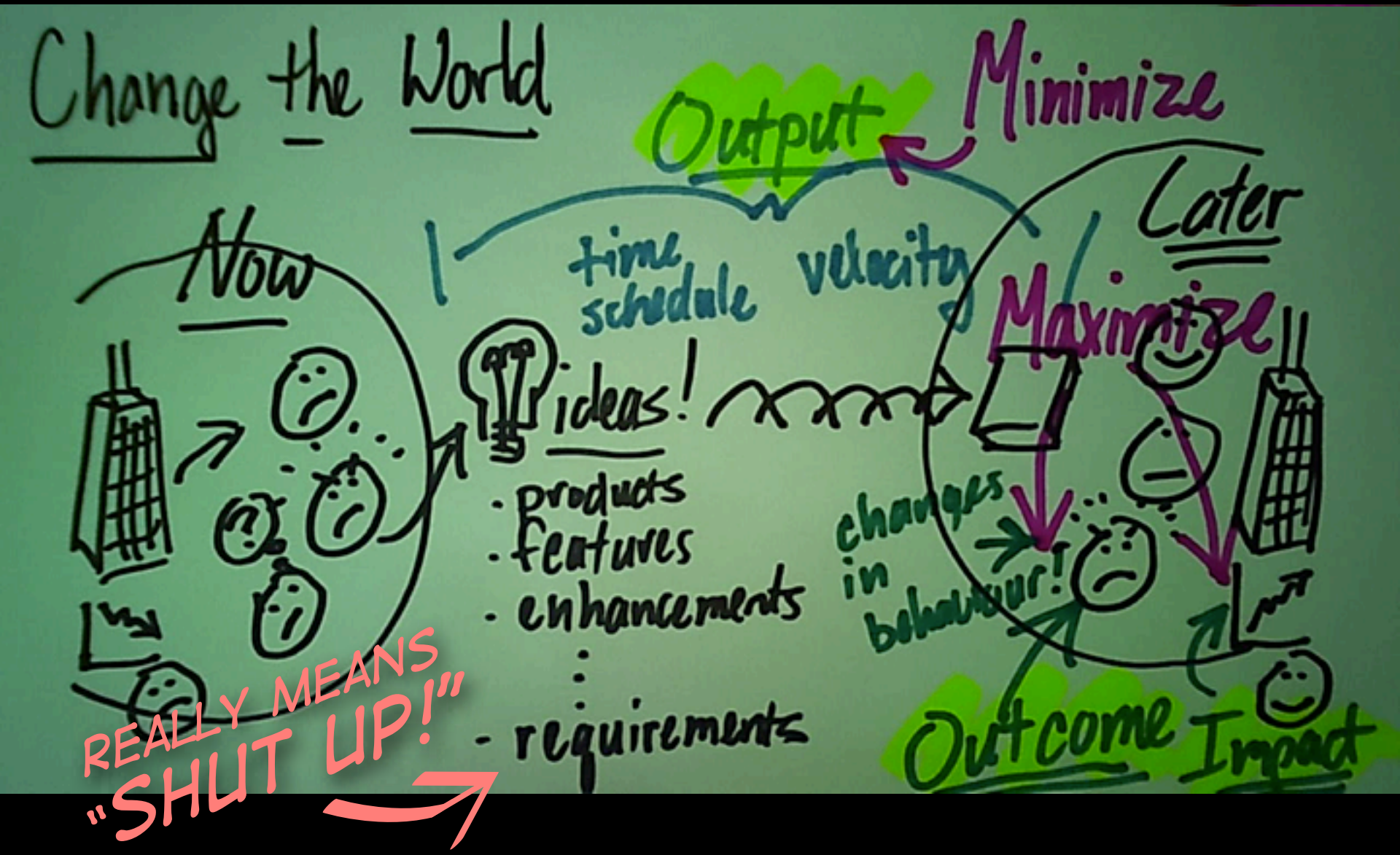

Change the World
Output
Now
time
schedule
velocity
Later
ideas!
- products
- features
- enhancements
:
- requirements
changes
in
behaviour!
Outcome


Change the World
Output
Now
time
schedule
velocity
Later
ideas!
- products
- features
- enhancements
- requirements
changes
in
behaviour!
Outcome
Impact


Change the World
Output
Now
time
schedule
velocity
Later
ideas!
- products
- features
- enhancements
- requirements
changes in behaviour!
Outcome
Impact


Change the World
Output
Minimize
Now
time
schedule
velocity
Later
Ideas!
- products
- features
- enhancements
changes
in
behaviour!
Outcome
Impact


Change the World
Output
Minimize
Now
time
schedule
velocity
Later
Maximize
ideas!
- products
- features
- enhancements
- requirements
changes in behaviour!
Outcome
Impact


Change the World
Output
Minimize
Now
Later
time
schedule
velocity
Maximize
ideas!
- products
- features
- enhancements
:
- requirements
changes in behaviour!
Outcome
Impact
REALLY MEANS
"SHUT UP!"

Stories are an antidote to "requirements"

Software development has been steered wrong by the word 'requirement,' defined in the dictionary as "something mandatory or obligatory."

The word carries a connotation of absolutism and permanence, inhibitors to embracing change. And the word 'requirement' is just plain wrong.

Stories

Plan using units of
the traffic with the same
users to dial frequently used numbers.
to estimate the development effort necessary to implement it.

Software development has been steered wrong by the word "requirement", defined in the dictionary as "something mandatory or obligatory." The word carries a connotation of absolutism and permanence, inhibitors to embracing change. And the word "requirement" is just plain wrong. Out of one thousand pages of "requirements", if you deploy a system with the right 20% or 10% or even 5%, you will likely

Kent suggested we talk about what happens when things come out

Talk about who does what, and why

What I was thinking
of was the way users sometimes
tell stories about the cool new things the
software they use does:

"I type in the zip code and it automatically fills in the city and state without me having to touch a button!"

I think that was the example that triggered the idea.
If you can tell stories about what the software does
and **generate energy and interest and a vision in your listener's mind**, then why not tell
stories before the software does it?

Focus discussion and collaboration around who will use the product and what they'll do later, after delivery

You'll have to think things through

This is a cake for a baby shower

Jen Yates' Cake Wrecks: www.cakewrecks.com

I don't think they thought this through…

This is a cake for a baby shower

Jen Yates' Cake Wrecks: www.cakewrecks.com

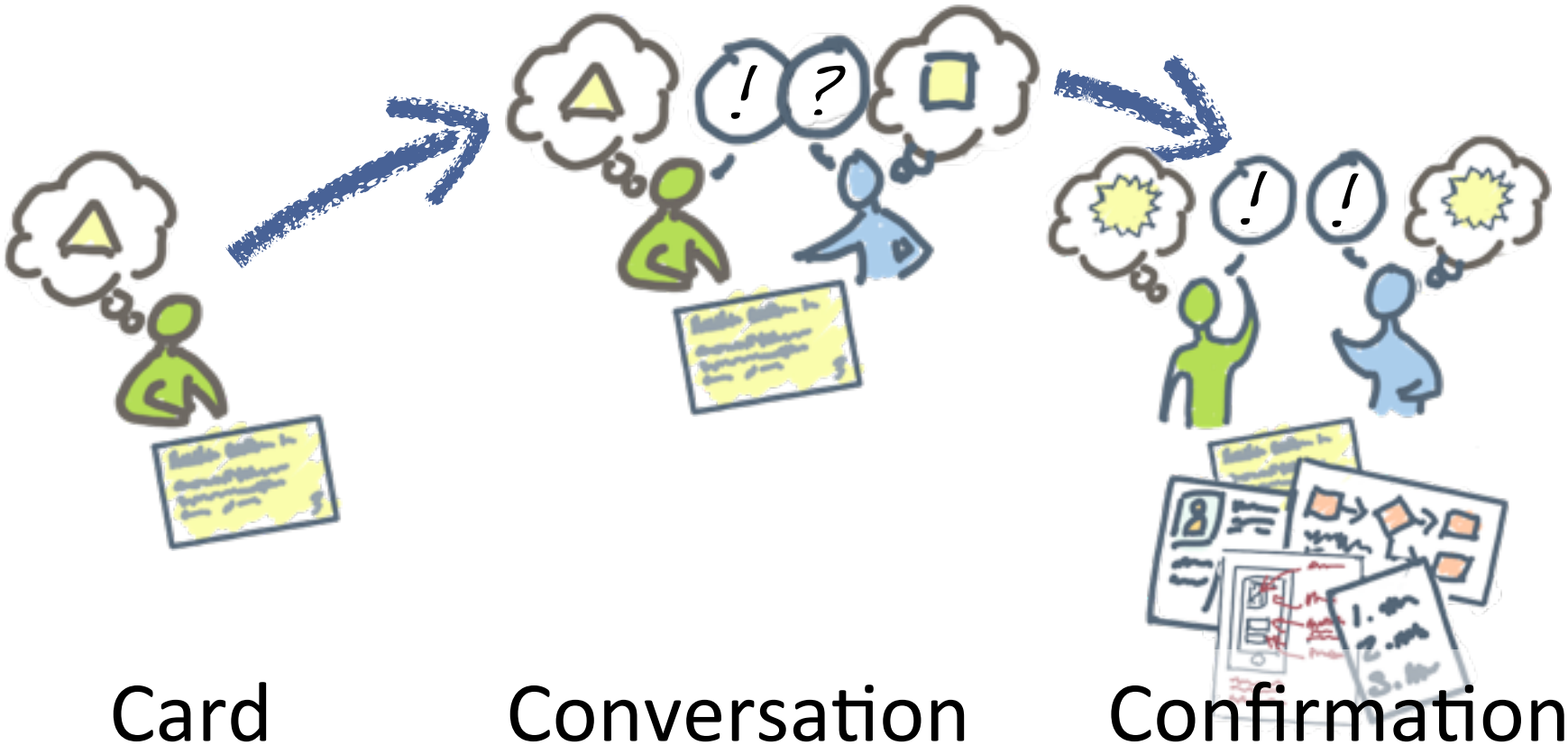
Stories are meant to solve 2 problems:

1. Building shared-understanding through story telling

2. Minimizing output, maximizing outcome and impact

Stories aren't a different way to write requirements, they're a different way to work

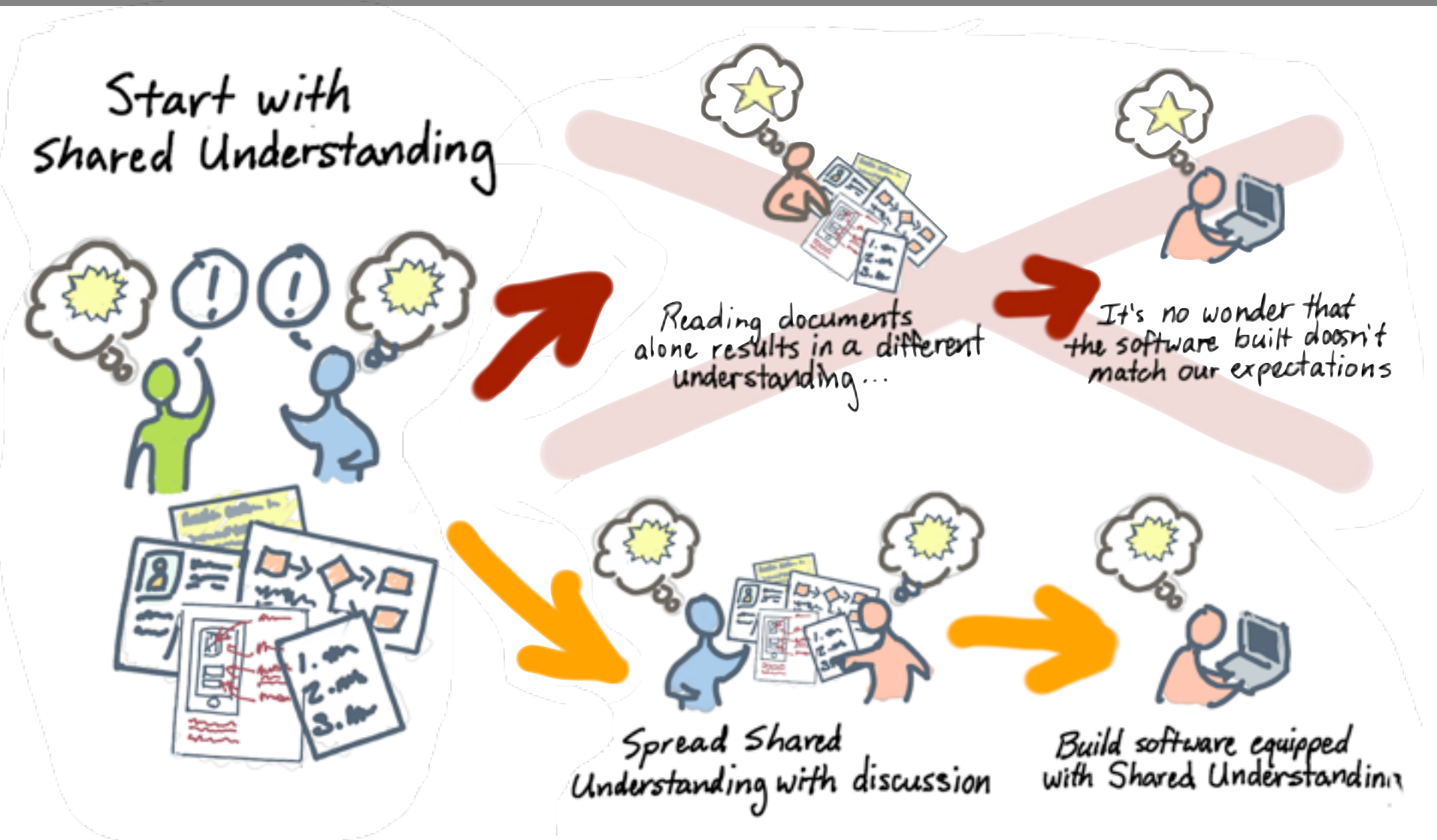
Stories have a simple lifecycle



Card Conversation Confirmation

** Ron Jeffries coined the 3 C's in Extreme Programming Installed*

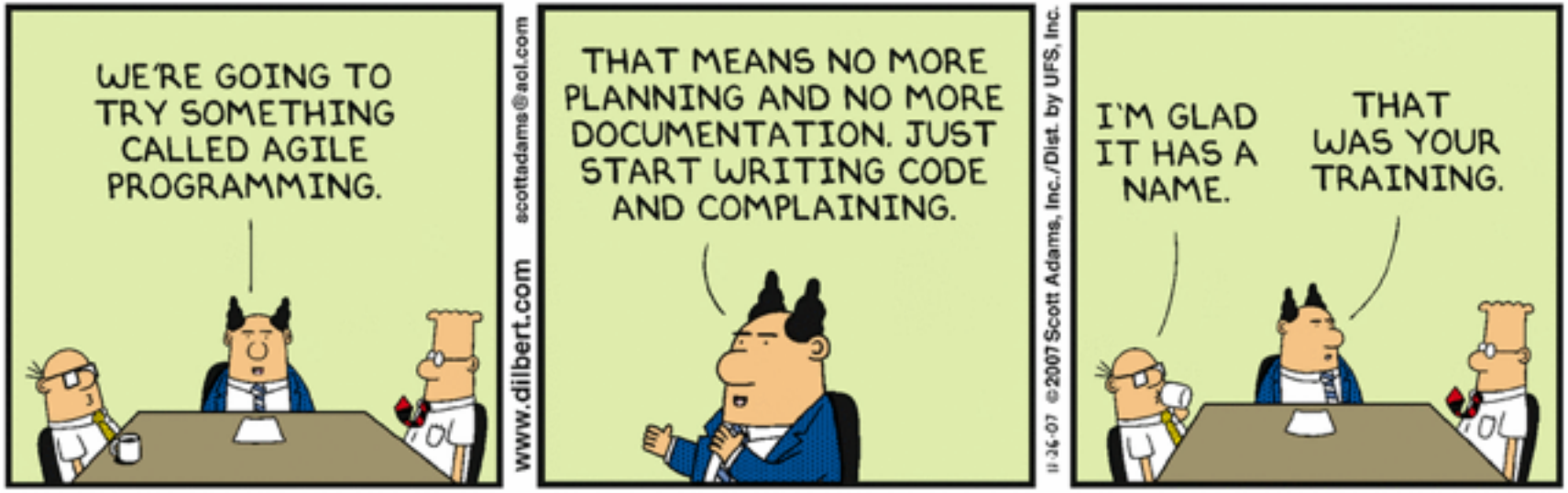
If you replace a conversation with a document, you've stopped using stories



If you replace a conversation with a document, you've stopped using stories

© 2003 United Feature Syndicate http://dilbert.com/strip/2003-09-25

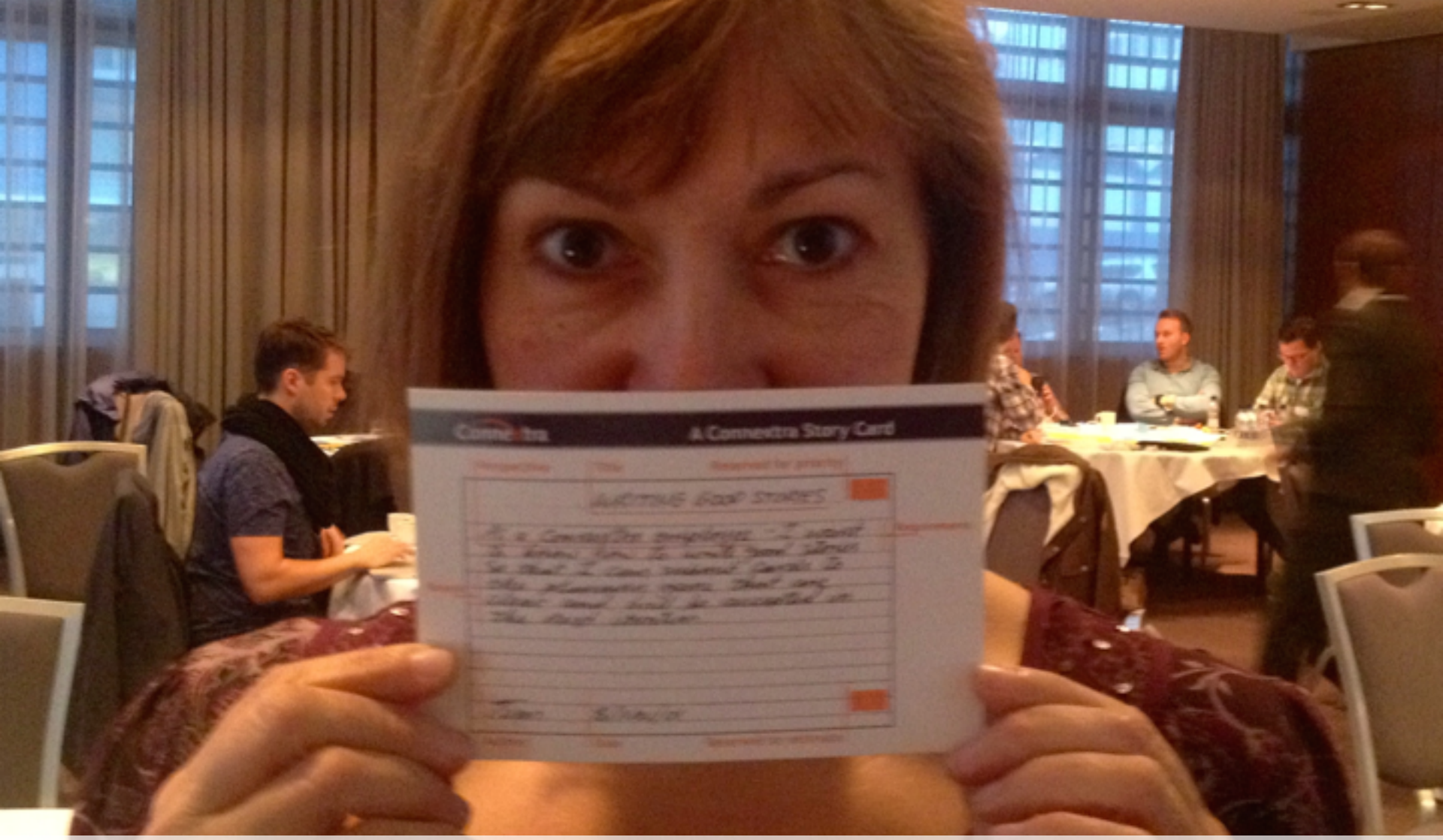
If you replace a conversation with a document, you've stopped using stories



© 2007 Scott Adams, Inc. http://dilbert.com/strip/2007-11-26

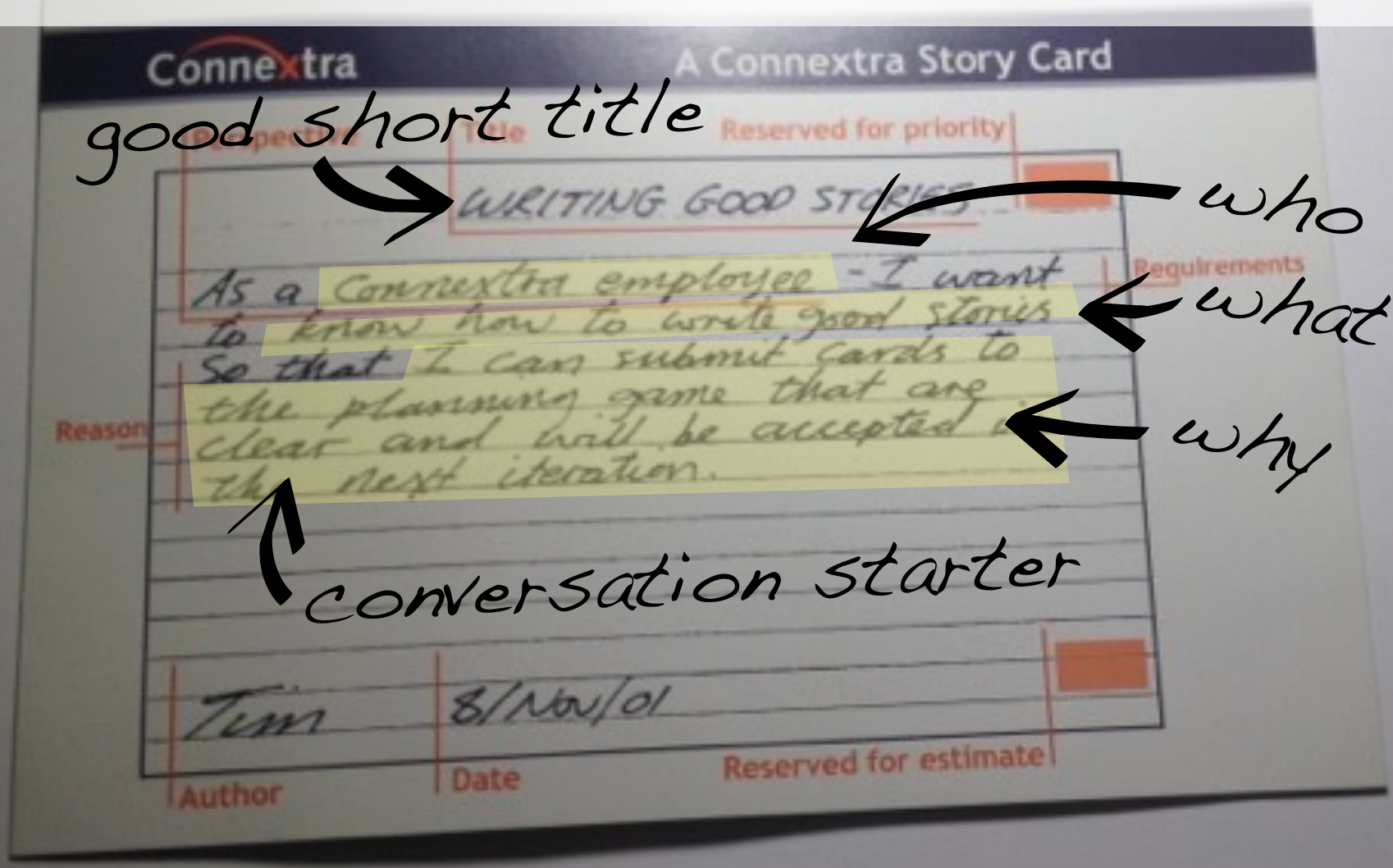
A template for starting good conversation

and stopping useless conversations



Say “Hi” to Rachel

Rachel and her team at Connextra created a clever conversation starter



Really discussing the who's, what's, and why's goes beyond the simple template

Who are the users and what benefit do they get?

- Discuss many possible users.
- Discuss stakeholders and others that may not directly use the software.
- Discuss elements of the system, or services

What will users do in the future using your software?

- Discuss what users will do with the feature or changes being
- Discuss what the system will do, especially if we're talking about a backend service

Why should your organization build the software?

- Discuss who benefits - it may not be the user
- Discuss why users would benefit
- Discuss why the organization would benefit

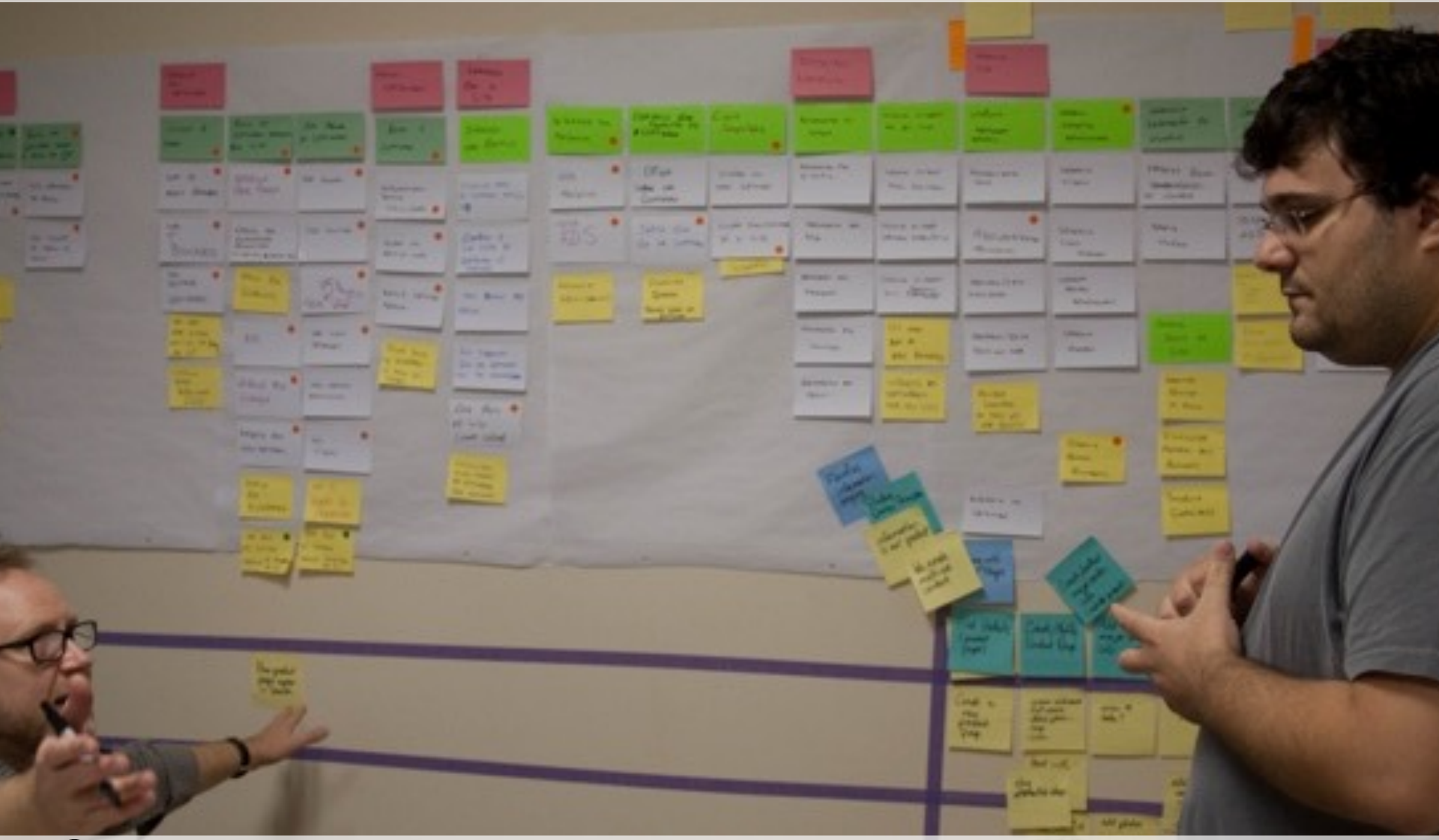
Keep what's on the card, or in the backlog simple

Use story cards to reference all the information discussed during conversations

A better way to build a backlog

Build story maps in small collaborative groups

Use the map for continuous discussion



Use story maps to understand your whole product or feature's experience

Use mapping to break down big stories without losing the big picture

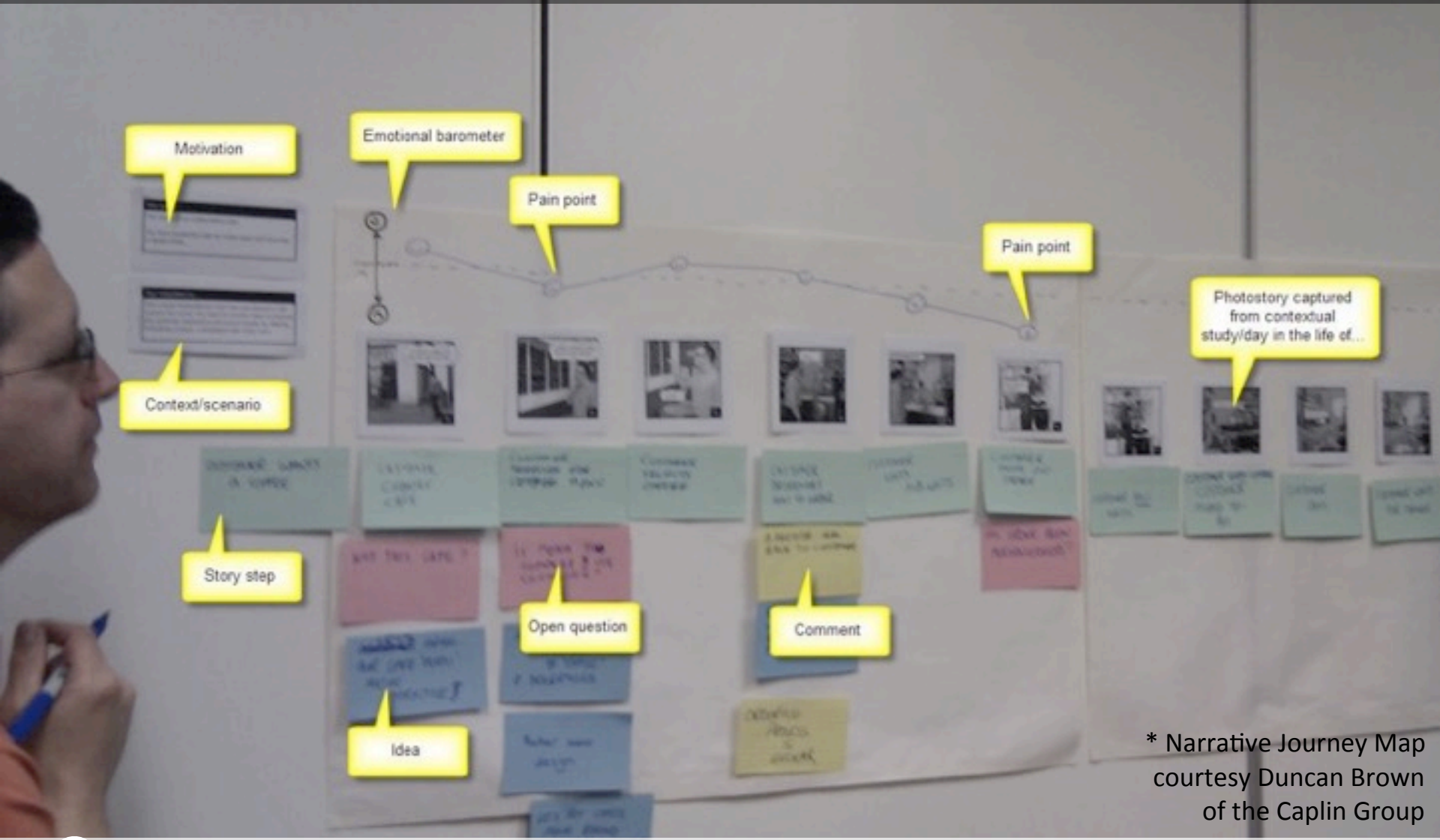
A better way to understand users and their challenges

Discussions drive out more details, validate, and build shared understanding

Journey maps describe the world today

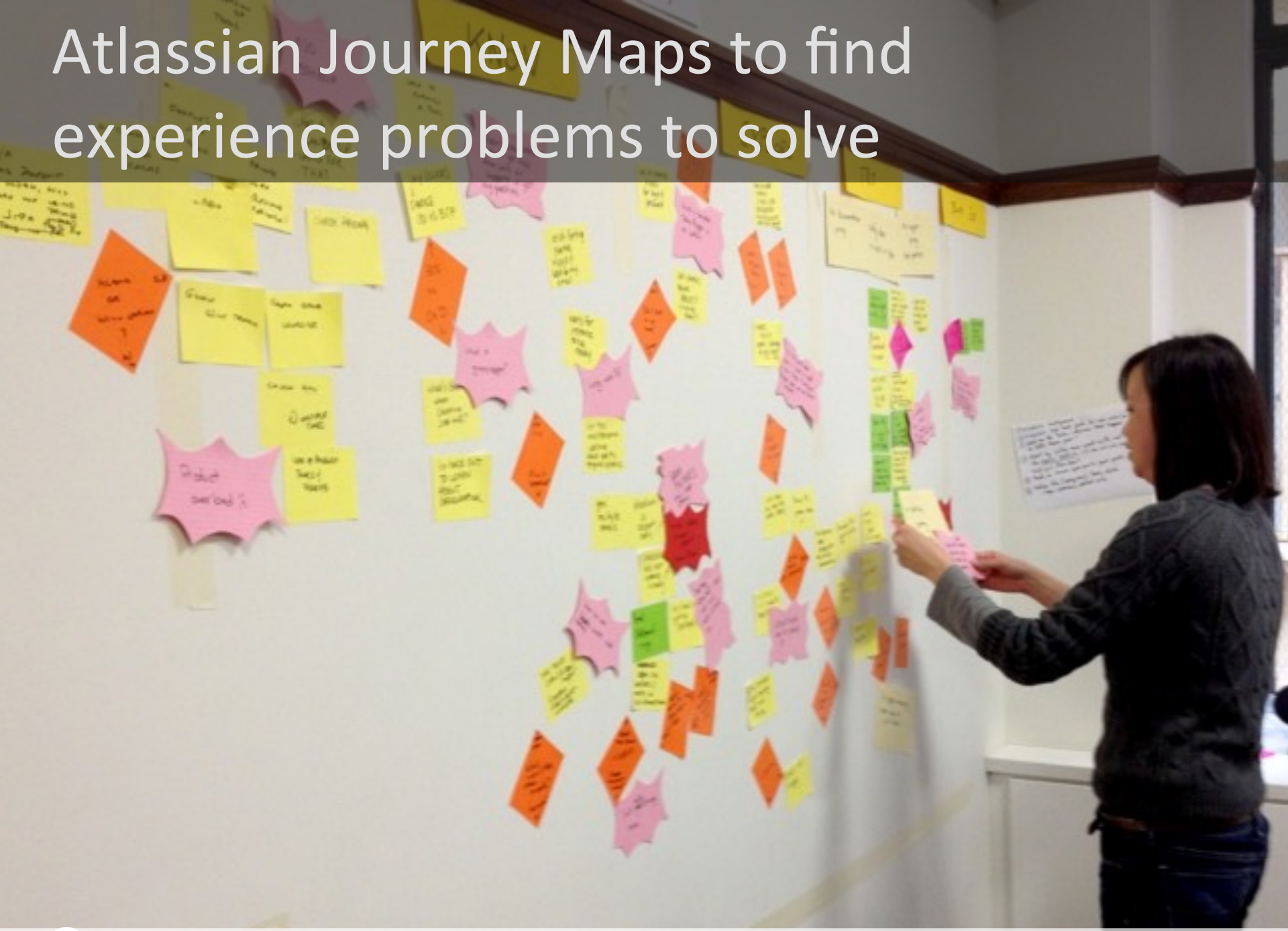
* Narrative Journey Map courtesy Duncan Brown of the Caplin Group

Journey maps describe the world today



* Narrative Journey Map courtesy Duncan Brown of the Caplin Group

Atlassian Journey Maps to find experience problems to solve

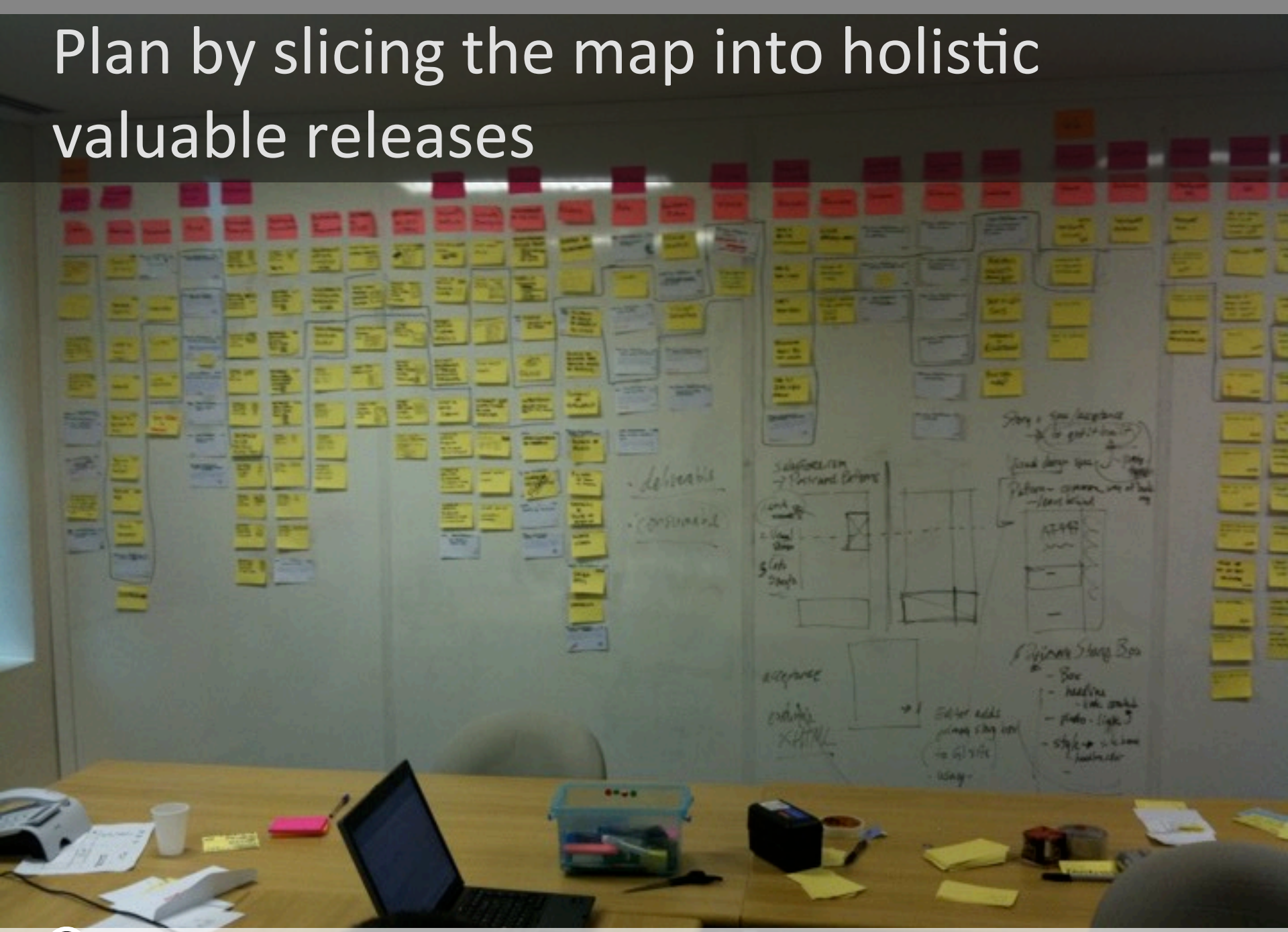


Atlassian Journey Maps to find experience problems to solve

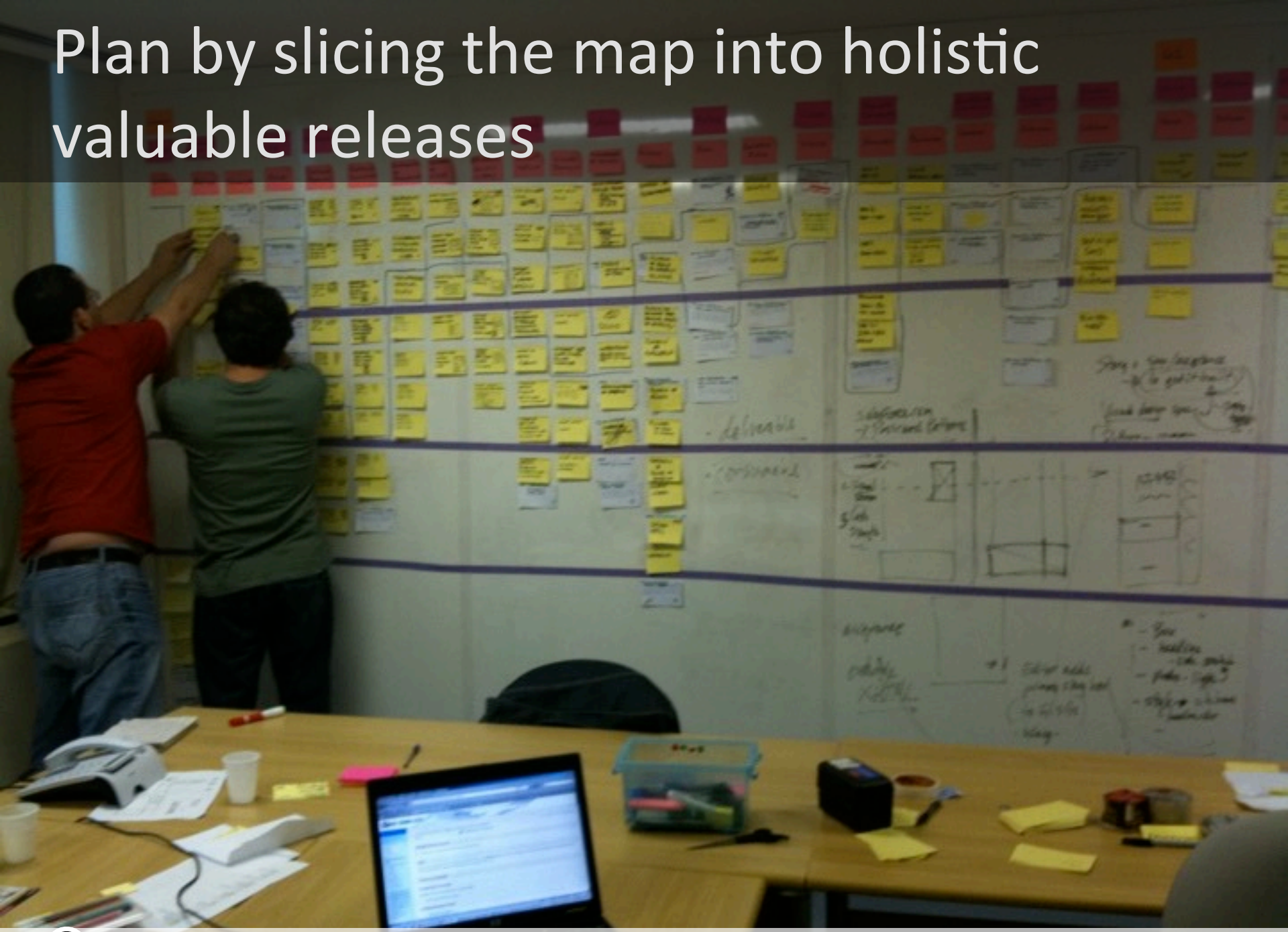
Use maps to understand the way people work today, and identify options for improvement

A better way to find smaller viable releases

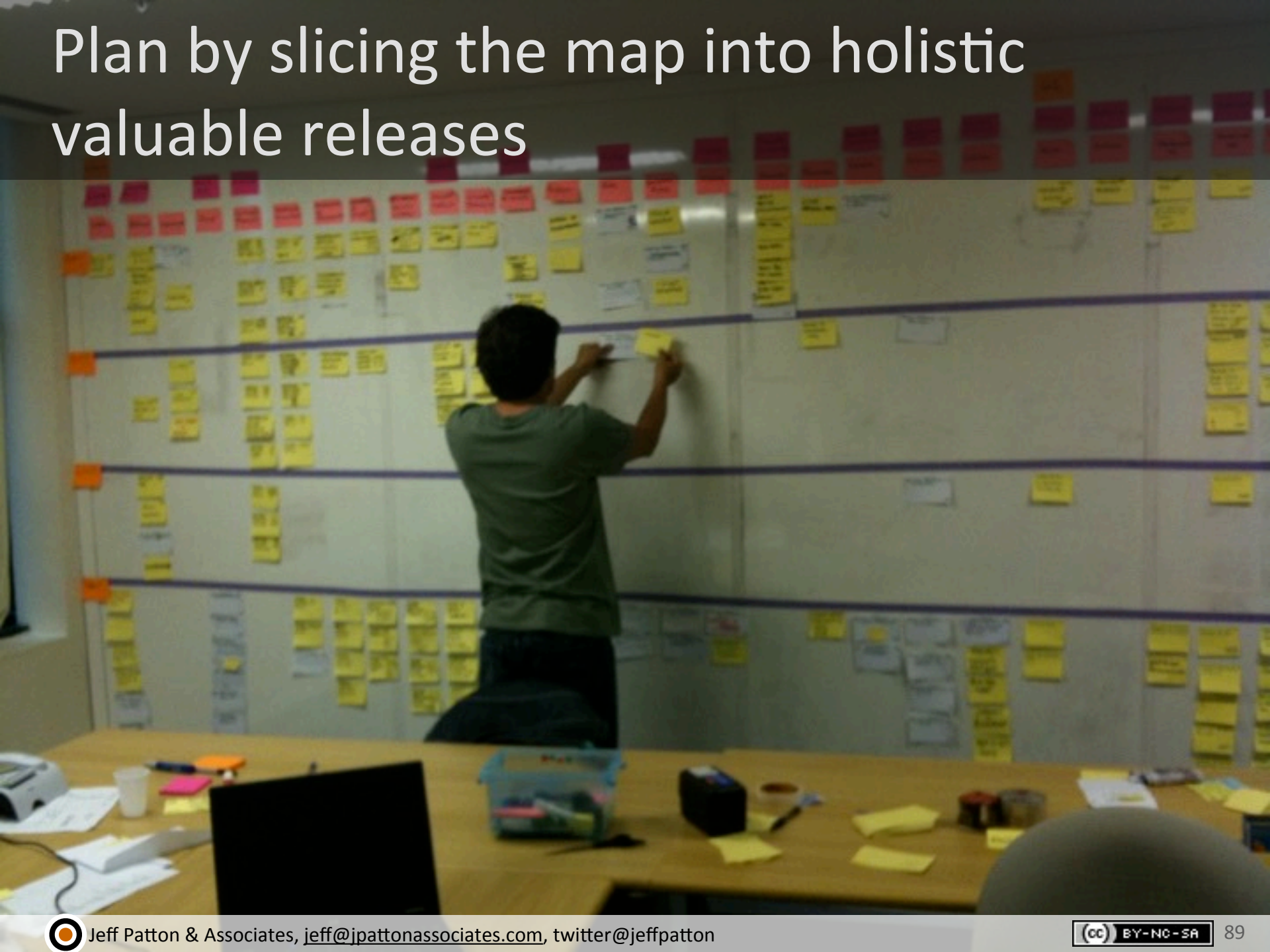
Plan by slicing the map into holistic valuable releases



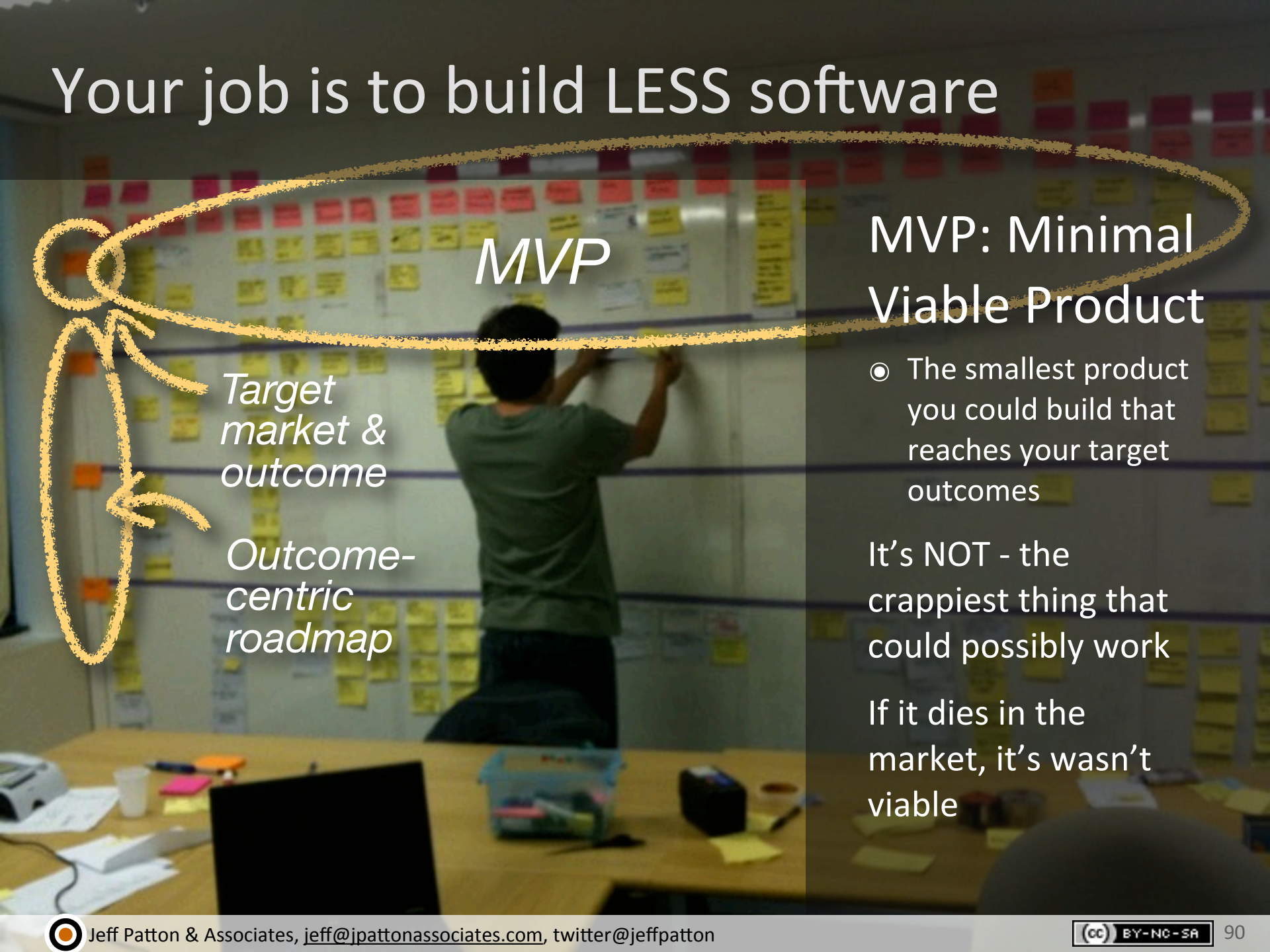
Plan by slicing the map into holistic valuable releases



Plan by slicing the map into holistic valuable releases



Your job is to build LESS software



MVP: Minimal Viable Product

- The smallest product you could build that reaches your target outcomes

It's NOT - the crappiest thing that could possibly work

If it dies in the market, it's wasn't viable

Use maps to focus release strategy on specific users and outcomes

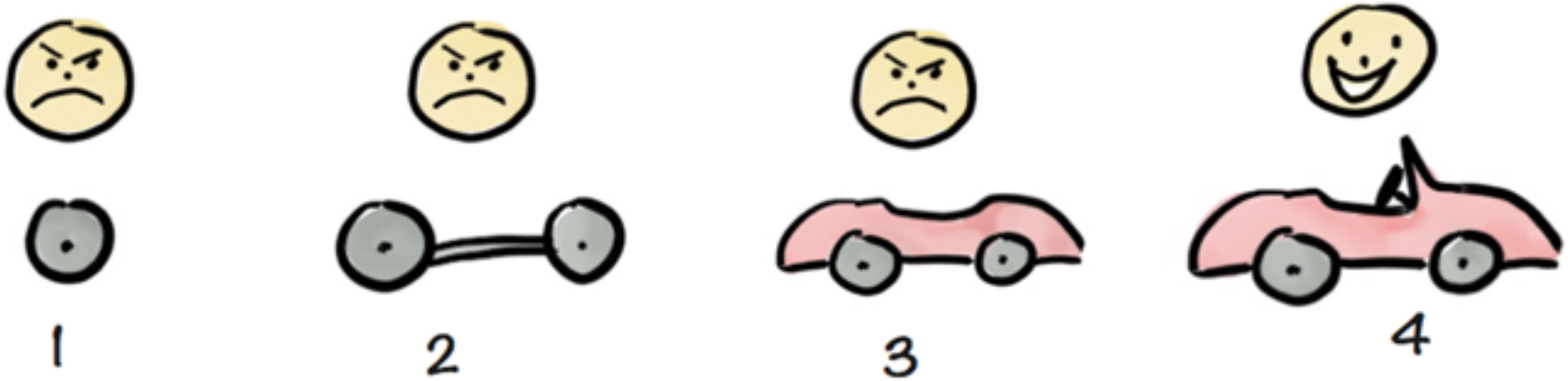
But, how do you know if you're hypothesis is correct?

You don’t

A better way to test your hypothesis

Delivering your hypothetical solution a piece at a time delays learning

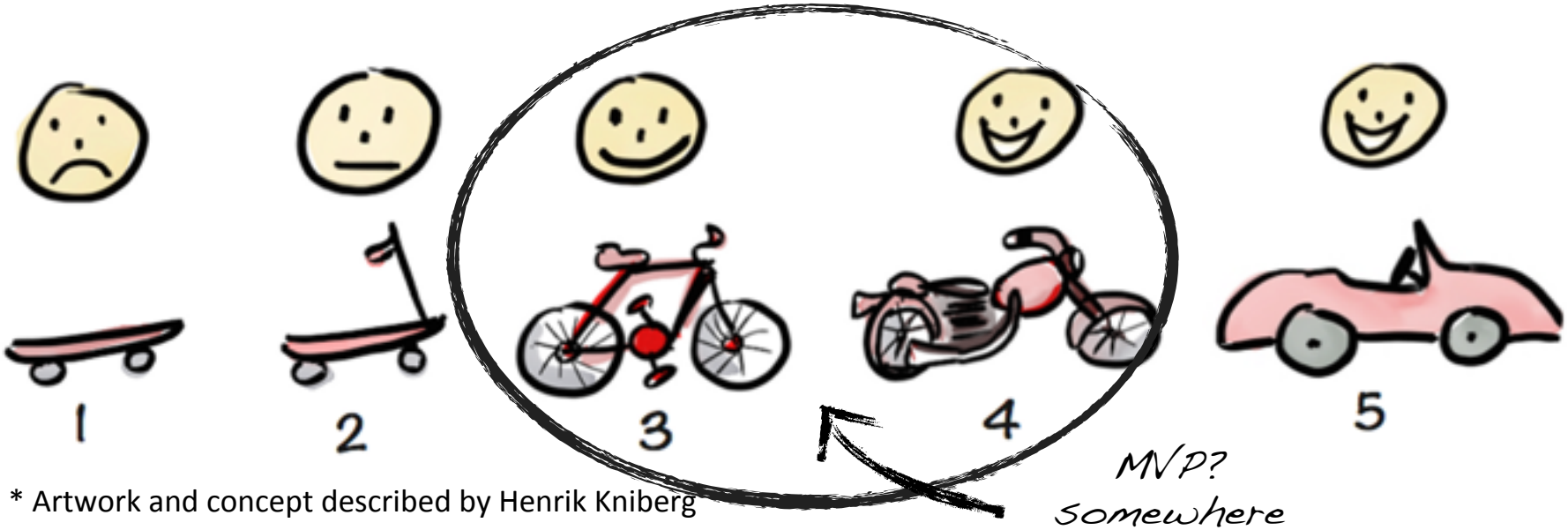
Hypothesis:



* Artwork and concept described by Henrik Kniberg

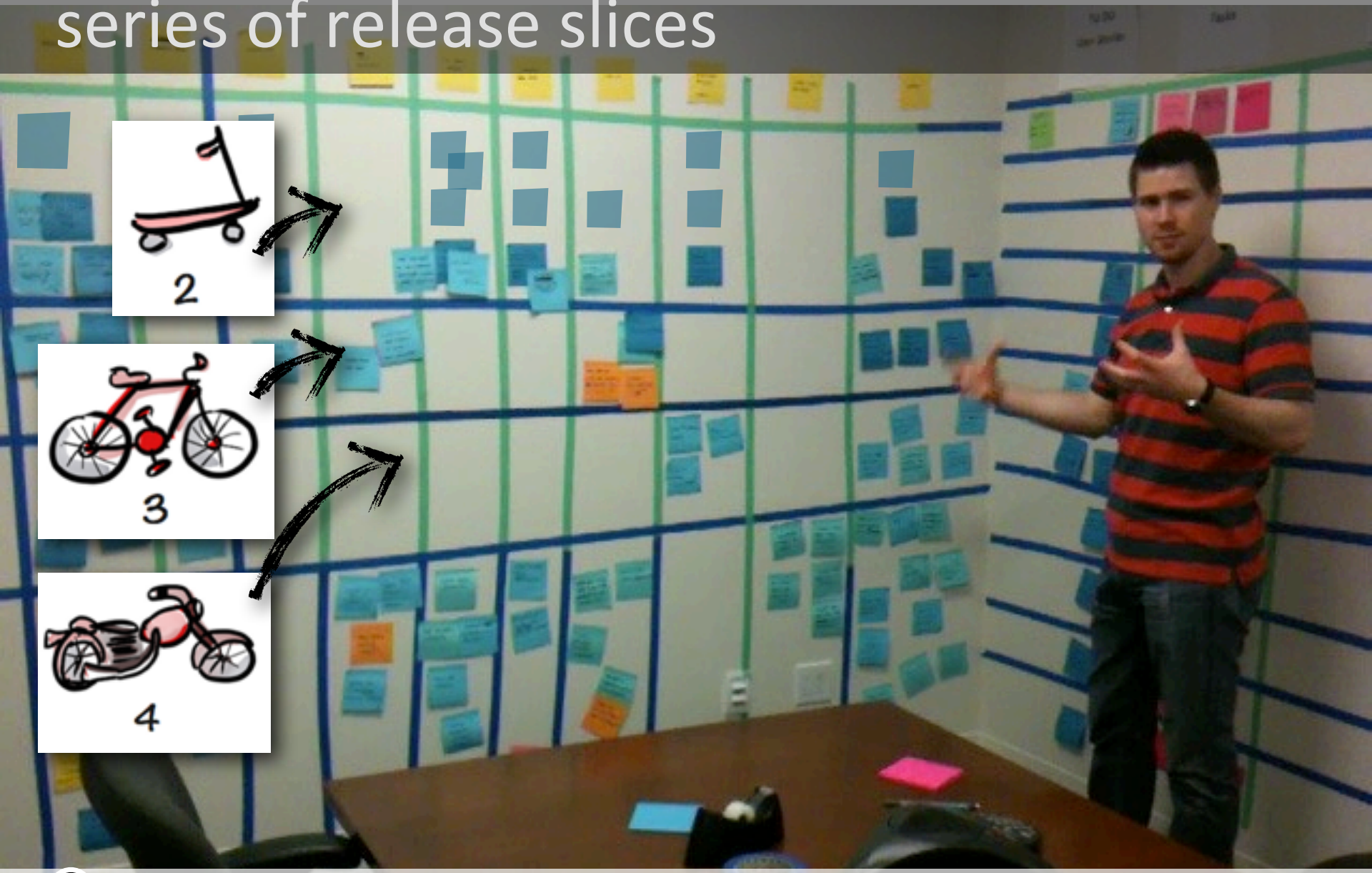
Deliver minimum viable product **tests** to a smaller audience to find what's really viable

Hypothesis:



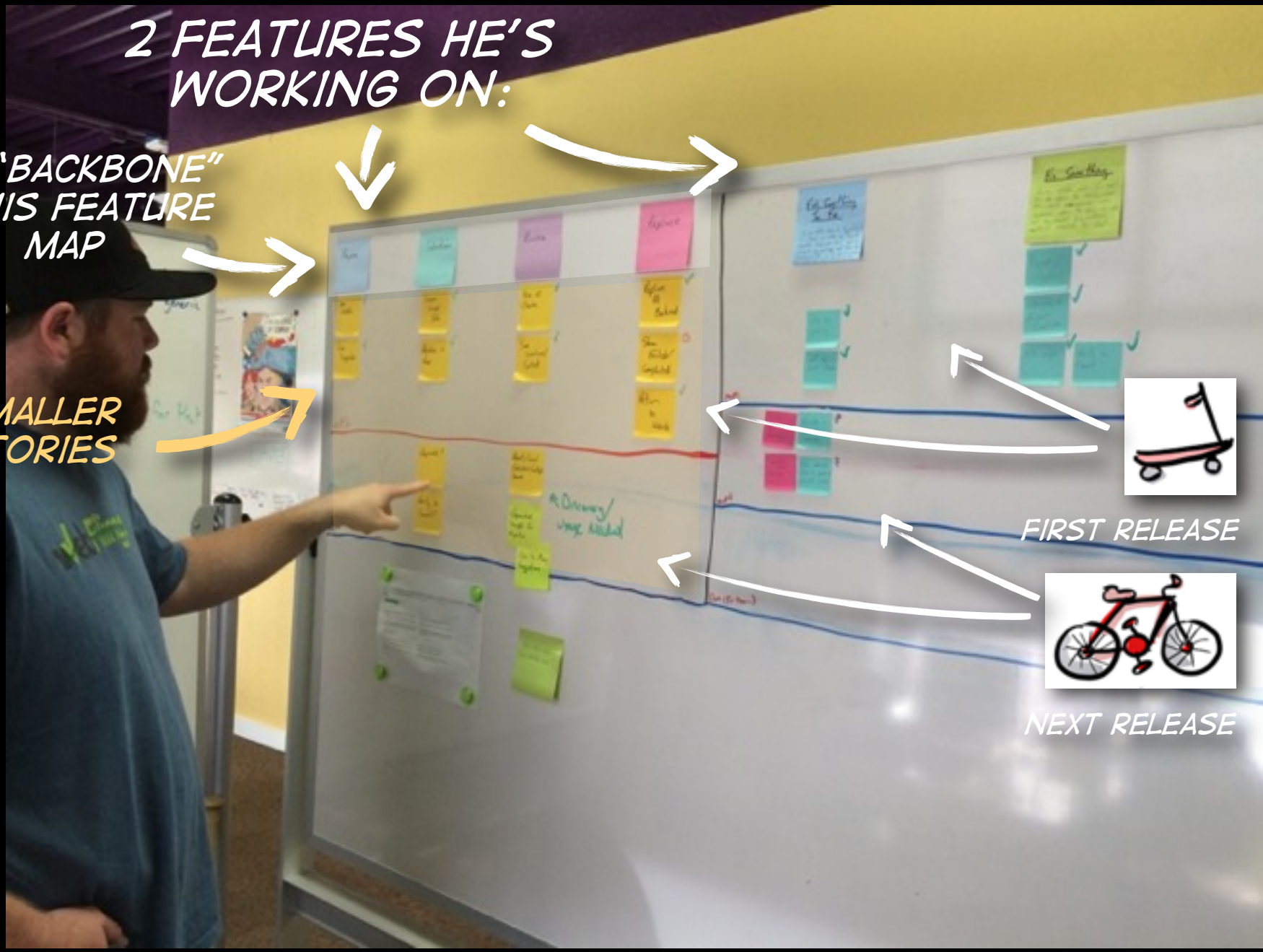
* Artwork and concept described by Henrik Kniberg

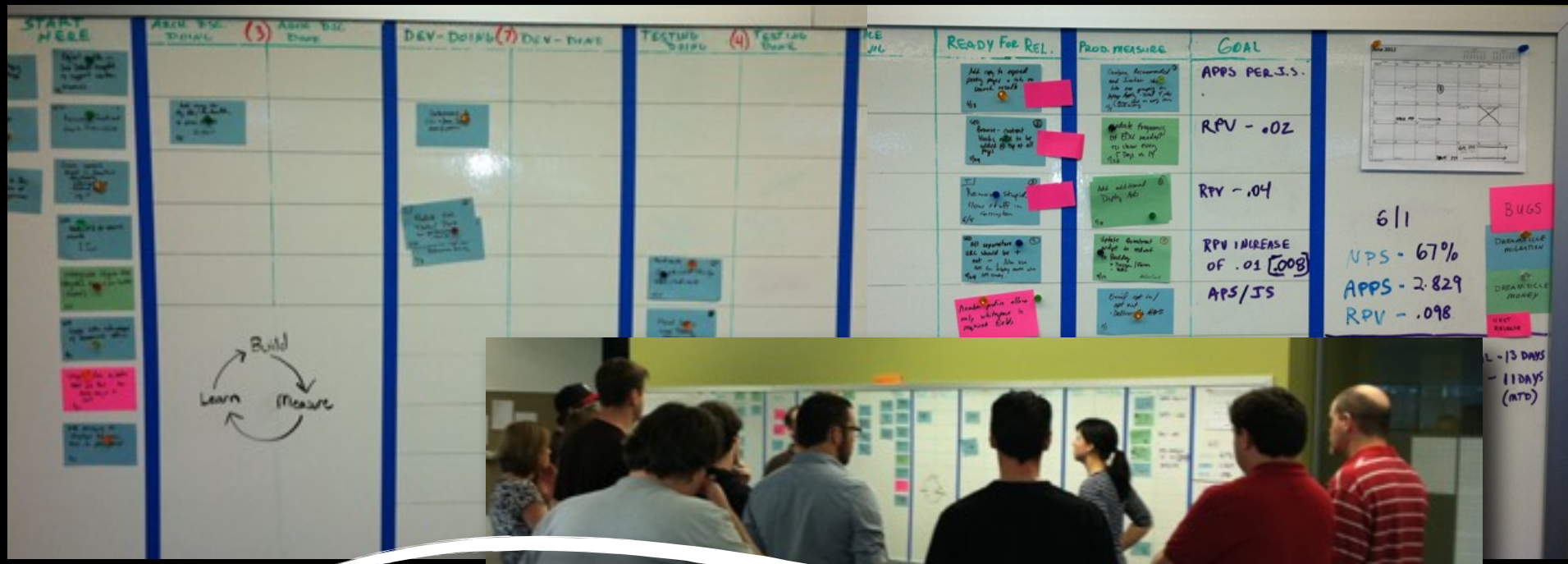
Eric has organized his backlog into a series of release slices

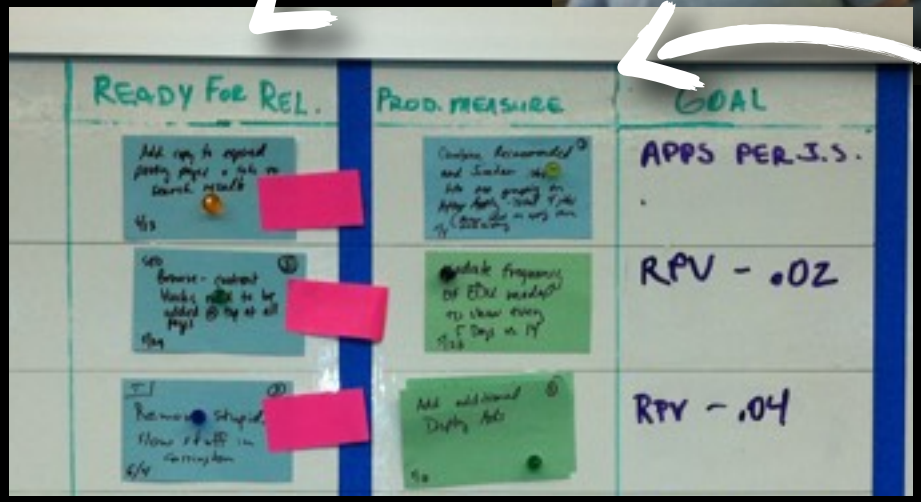


Jade releases to learn

Jade Robbins, Product Manager at Workiva


2 FEATURES HE'S WORKING ON:
THE "BACKBONE" OF HIS FEATURE MAP
SMALLER STORIES
FIRST RELEASE
NEXT RELEASE





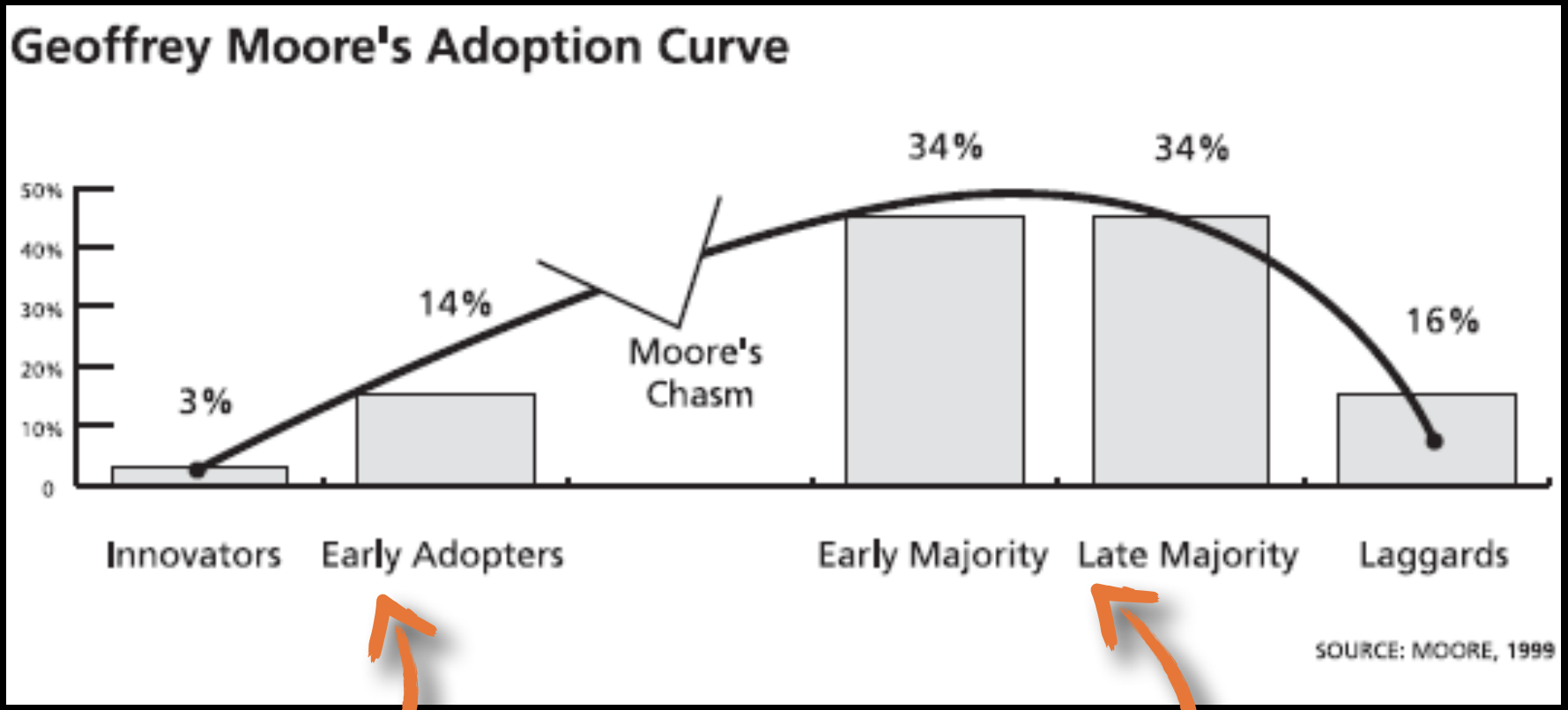
EXPLICIT RELEASE STEP

EXPLICIT MEASURE STEP & METRICS

NOTHING LEAVES THEIR BOARD UNTIL THERE'S BEEN A DISCUSSION ON WHAT THEY'VE LEARNED

Snag-a-Job's task board photo courtesy of David Bittenbender

All your users aren't guinea pigs: Use customer development partners



RELEASE EXPERIMENTS TO DEVELOPMENT PARTNERS

WHEN YOU'RE CONFIDENT IT'S VIABLE, RELEASE GENERALLY

Use maps to focus release strategy on specific learning outcomes

A safer way to deliver on time

Use maps to focus development strategy on reducing delivery risk

“accurate estimate” is an oxymoron

To release benefit on a schedule we'll need to budget, and leverage incremental and iterative thinking

(What's the difference?)

“incrementing” builds a bit at a time

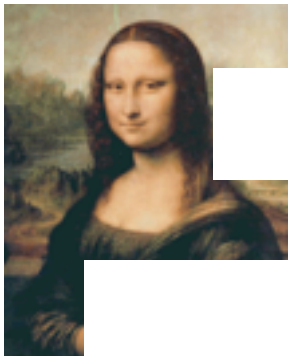
Incrementing calls for a fully formed idea.

And, doing it on time requires dead accurate estimation.

1

2

3



4

5

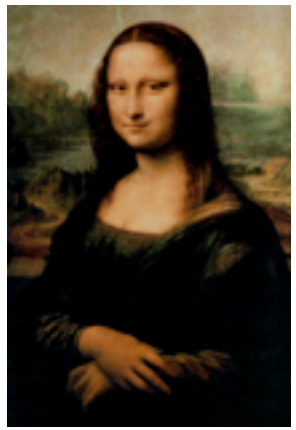
“iterating” and “incrementing” builds a rough version, validates it, then slowly builds up quality

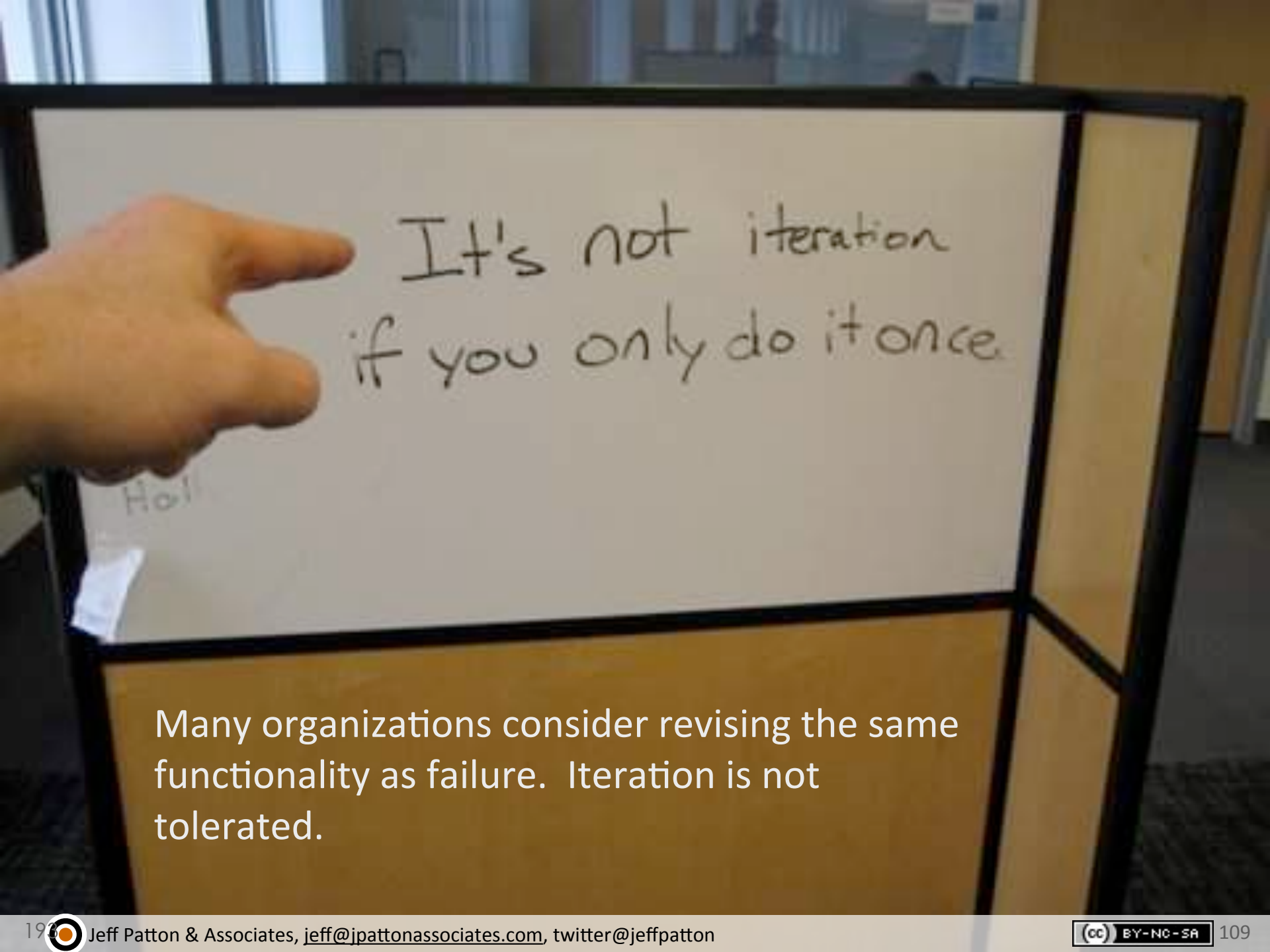
A more iterative allows you to move from vague idea to realization making course corrections as you go.

1 2 3 4 5







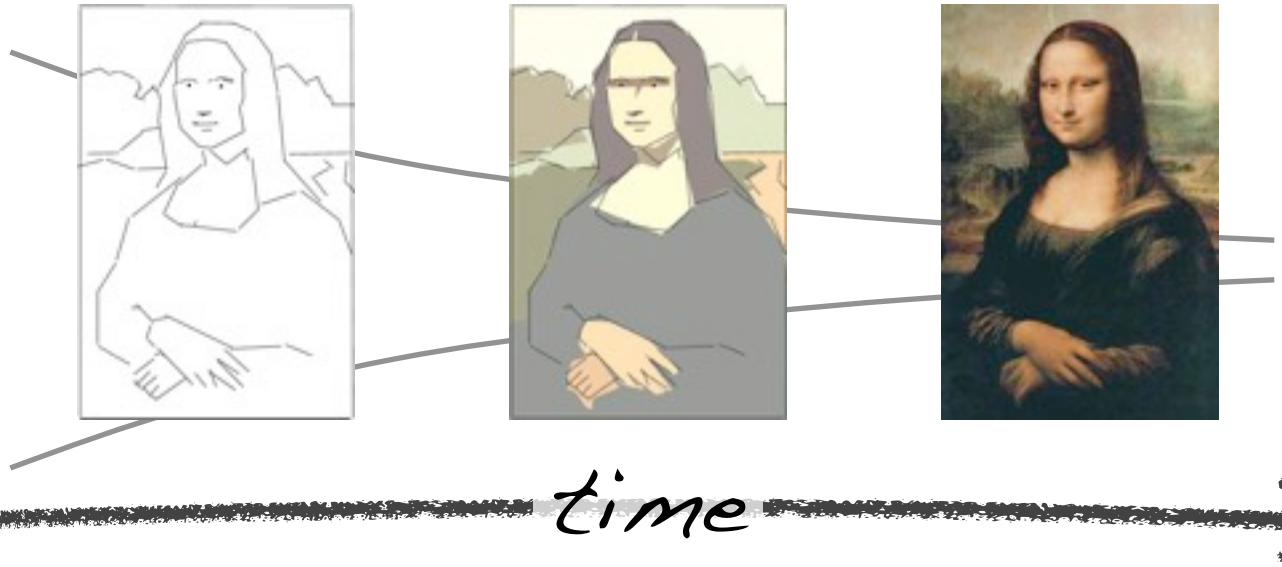


Many organizations consider revising the same functionality as failure. Iteration is not tolerated.

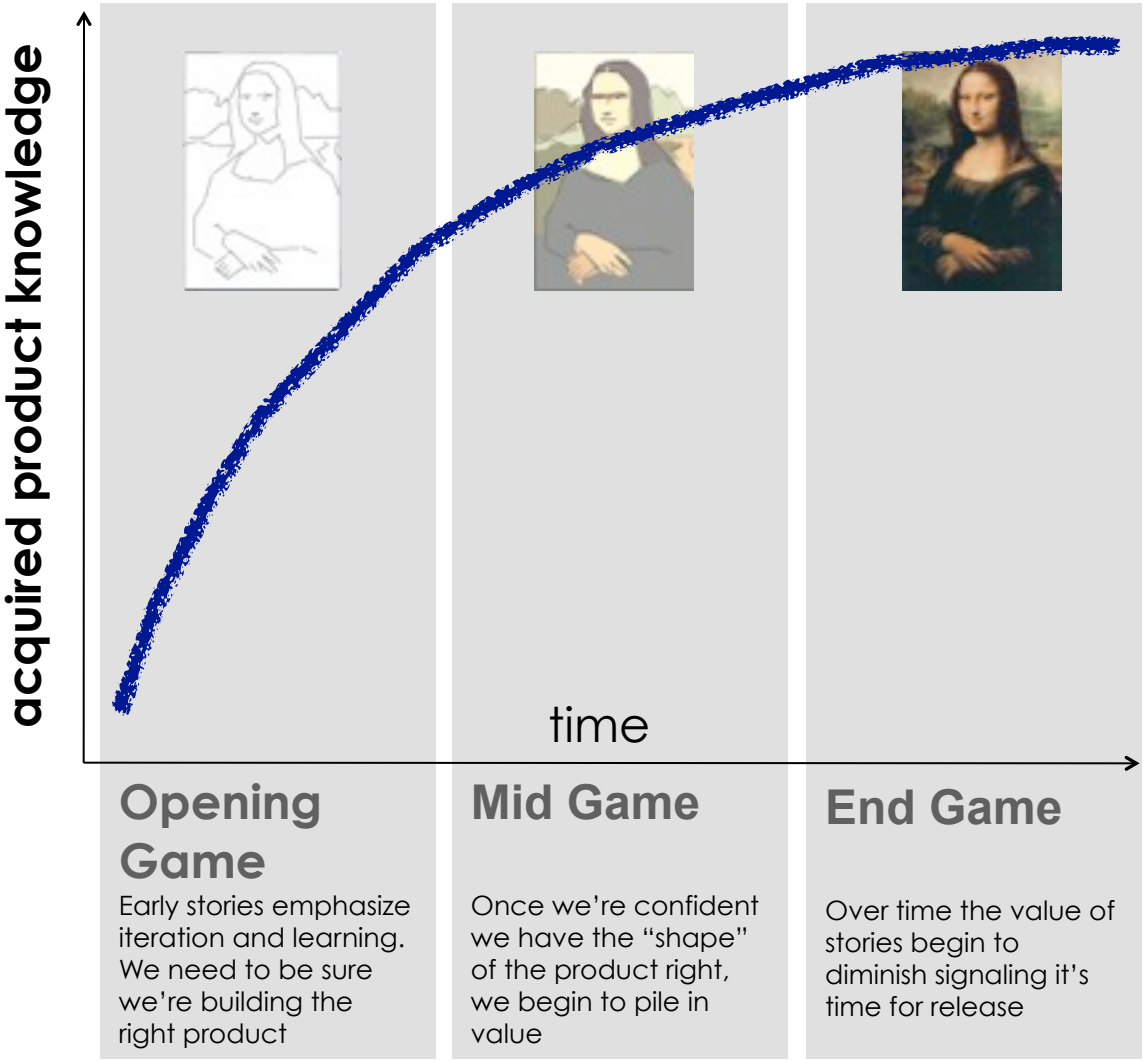
Work like an artist to envision and build the product holistically

"Art is never finished, only abandoned."

-Leonardo DaVinci



Organize work to maximize learning

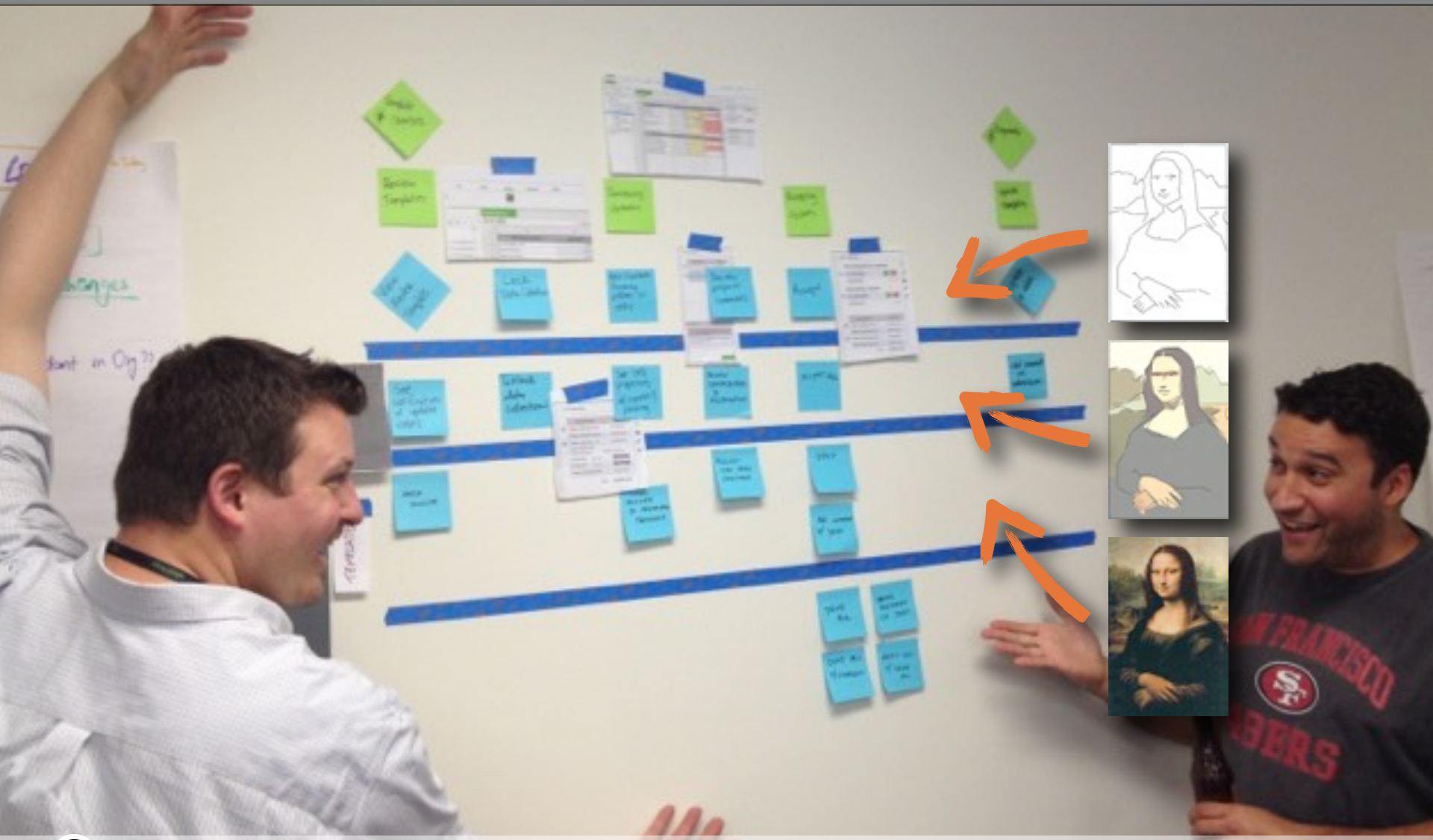


The inverse of risk is knowledge

Learning earlier about delivery risks helps us finish on time

Alistair Cockburn refers to cutting the small "polishing" stories as "trimming the tail."

Use a story map to slice out a delivery strategy



Product Owners must understand the delivery strategy that leads to a finished product

Sculpture at various stages of completion, Musée d'Orsay, Paris

Build up software iteratively and incrementally to release the highest quality possible on time

1. Change the way you work: tell stories, don't just write them

2. Use simple visualizations to anchor the stories you tell

3. Map the whole story to find the parts that matter most

4. Think things through: minimize output, maximize outcome and impact

5. Build minimum viable product tests to find what's minimum and viable in the market

6. Build your product up iteratively and incrementally

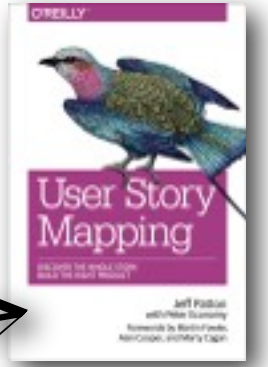
Effective stories help everyone work towards product success

Story Mapping

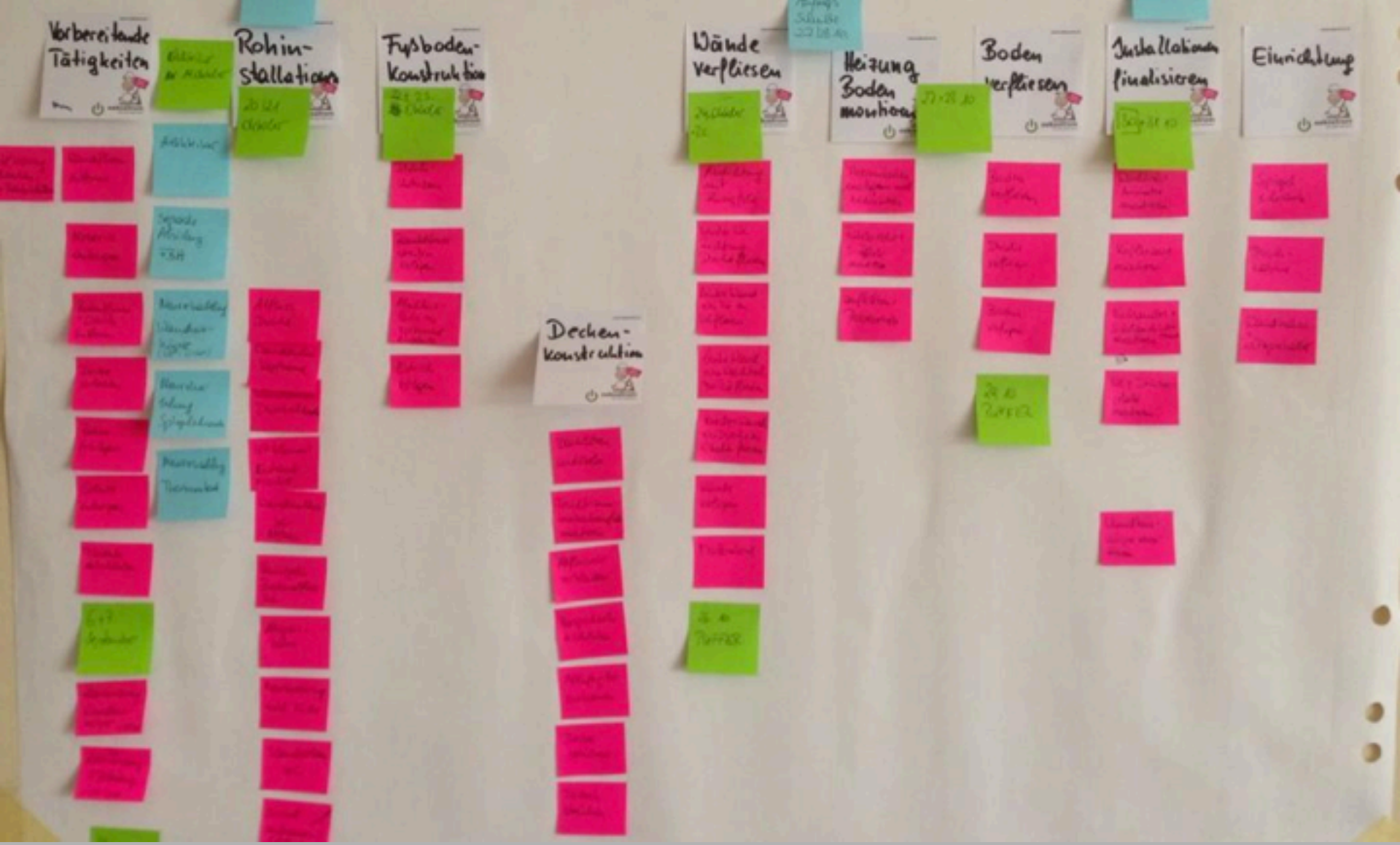
Don't Lose the Big Picture

Jeff Patton
jeff@jpattonassociates.com
twitter: @jeffpatton

I wrote this book!

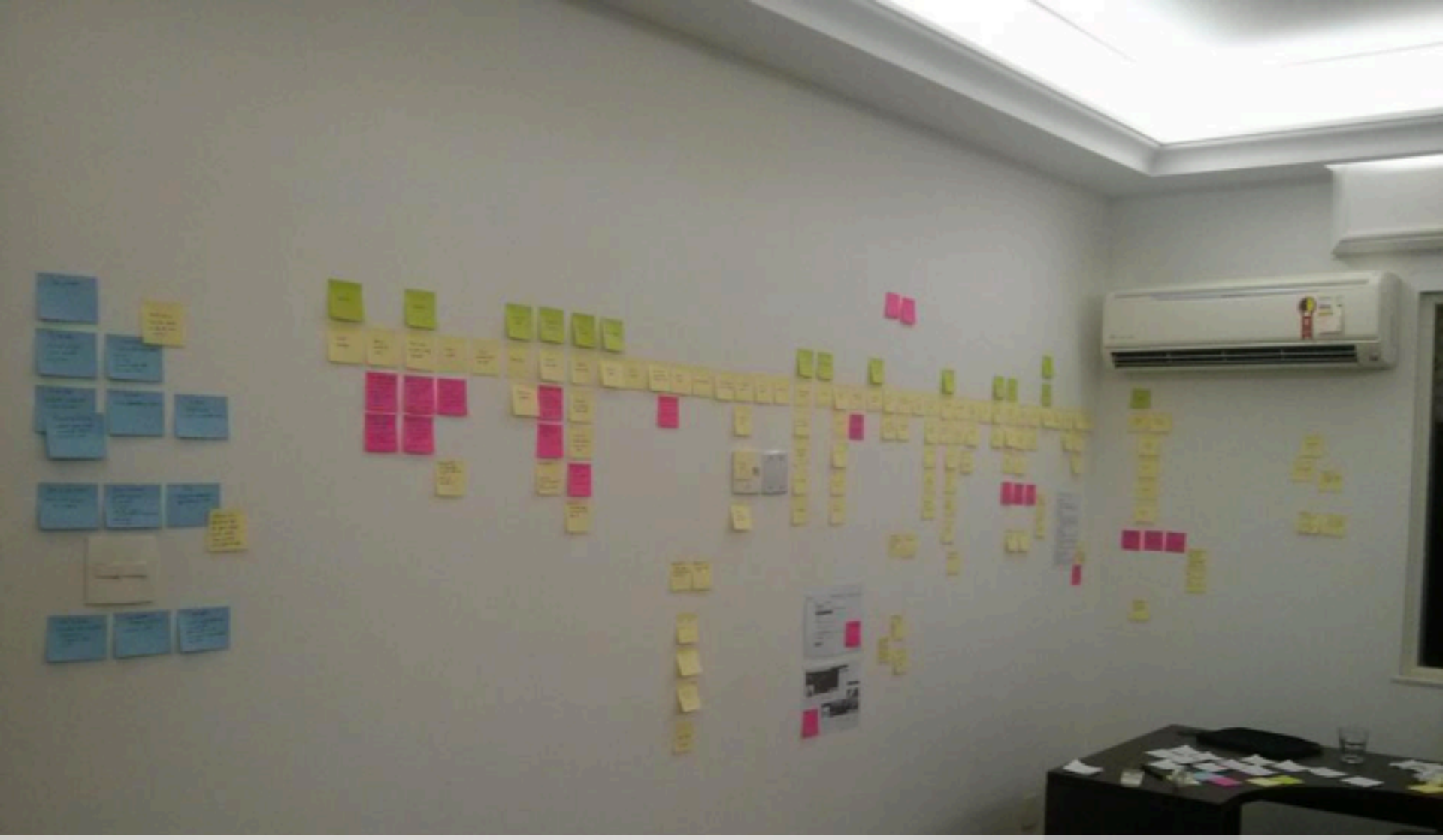


Ubiquity



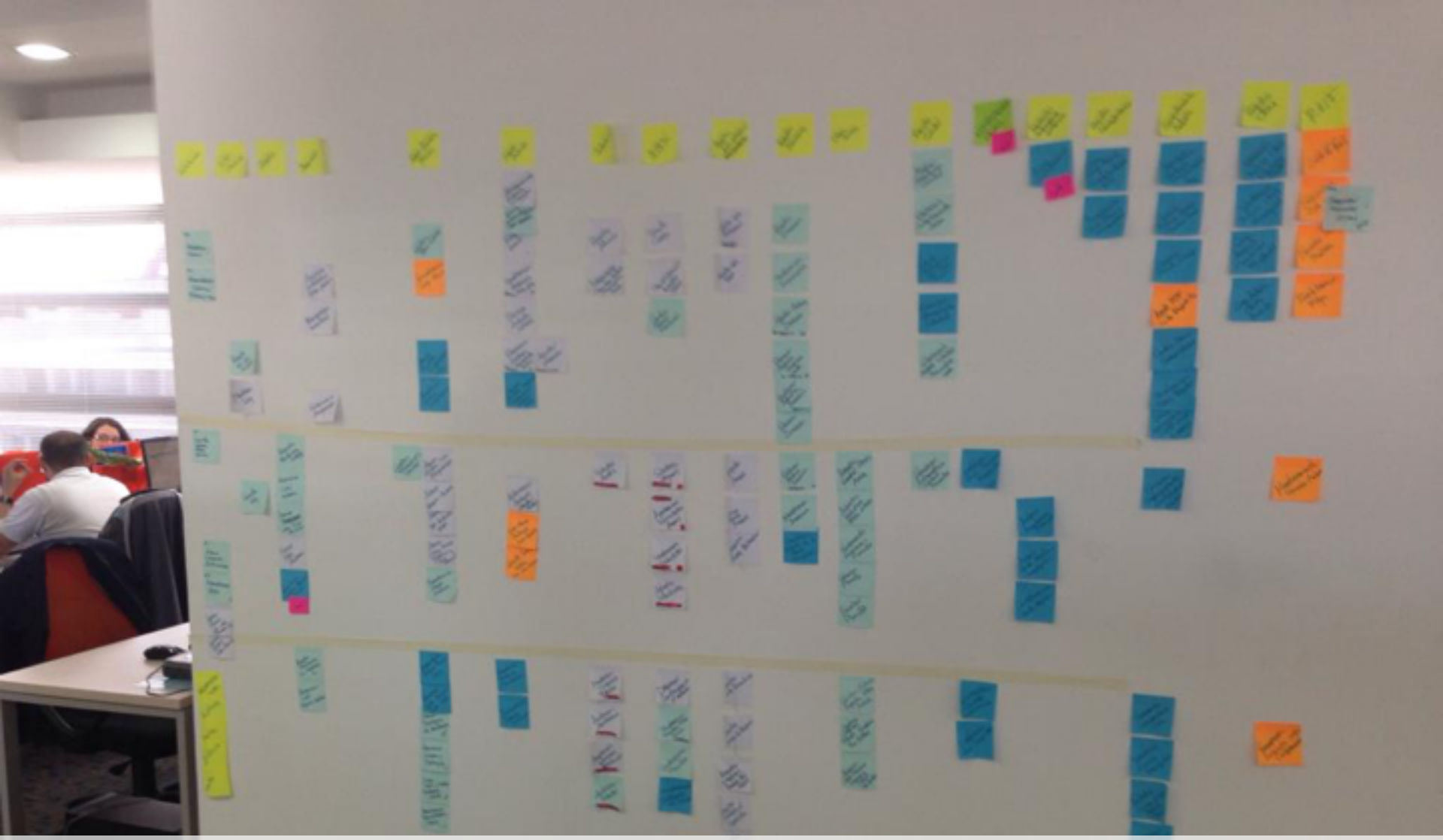
Frank Bieser, Austria, remodeling his bathroom

https://twitter.com/FBieser/status/506074949946470400



Ericson Costa, Brazil

https://twitter.com/eriksencosta/status/609549276755152896

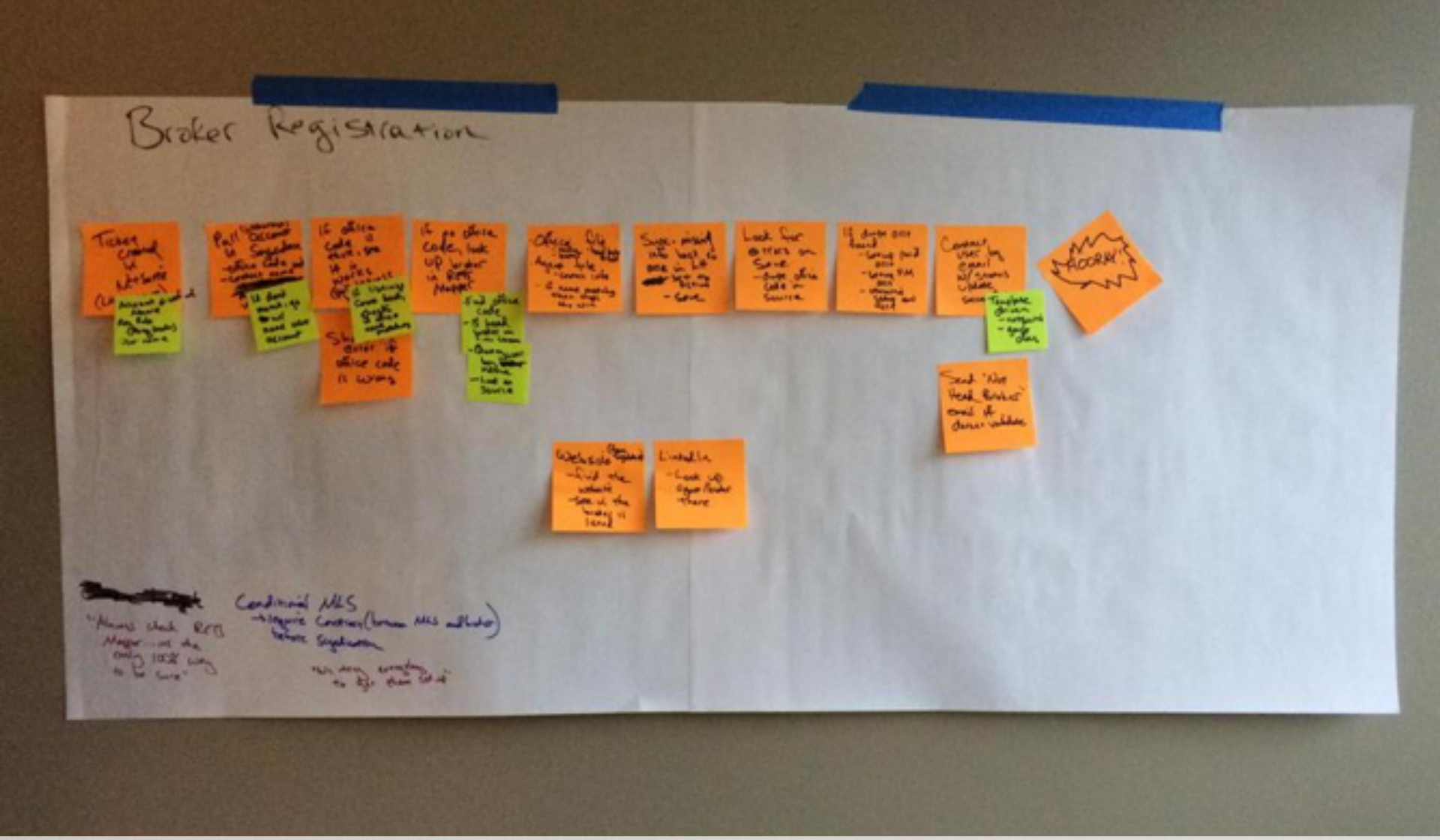


Tiziano Forero, Bogota, Columbia

https://twitter.com/tforero/status/609481839850450944

Tiziano Forero, Bogota, Columbia

https://twitter.com/tforero/status/609481839850450944



Jeremy Jarrell

https://twitter.com/jeremyjarrell/status/609473883457105920

Jose Manuel Beas, Madrid, Spain

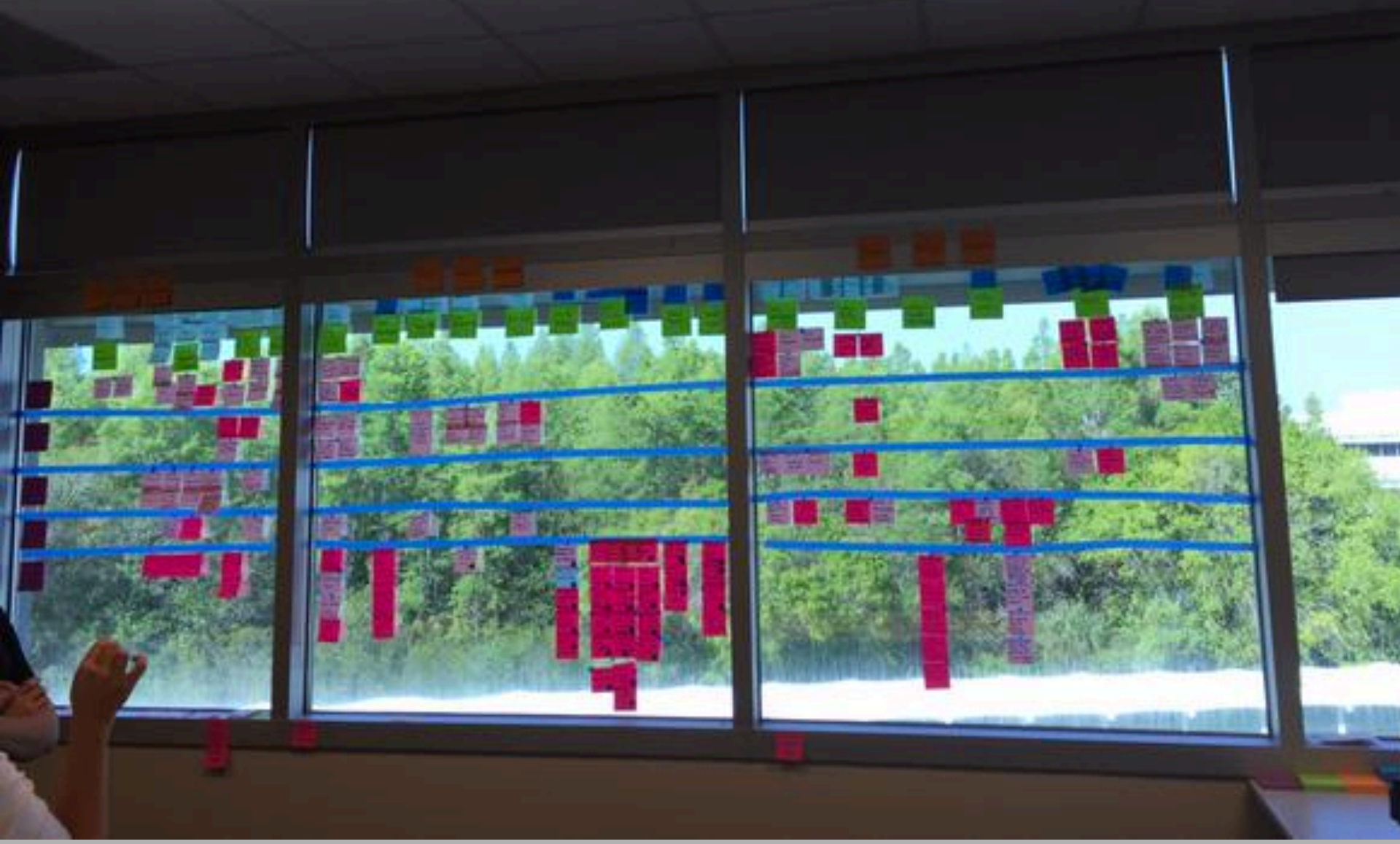
https://twitter.com/jmbeas/status/609469856430354435

80+ Story Mappers at SxSW

https://twitter.com/jeffpatton/status/577945438415798272

Story Mapping at Agile Richmond Via @archiemiller

https://twitter.com/archiemiller/status/608795873422790657

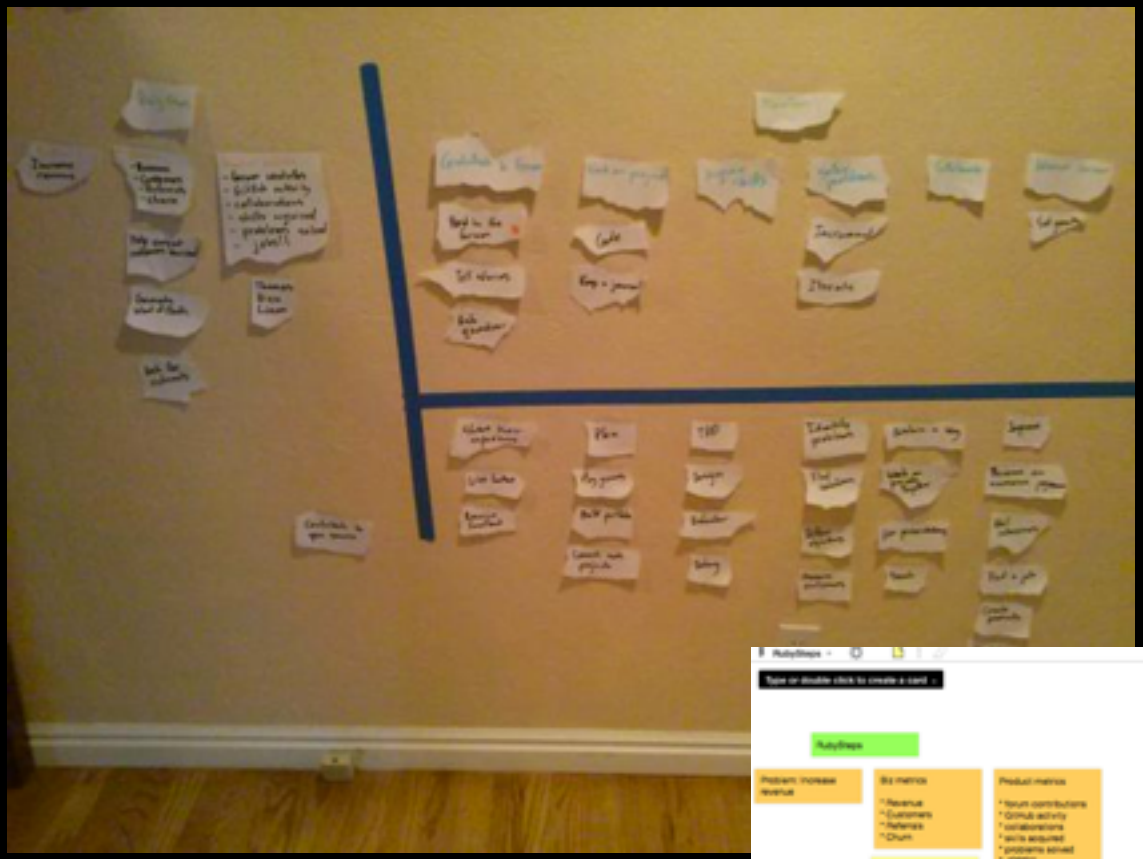


Story Map Plan via @ingeniousagile

https://twitter.com/ingeniousagile/status/606169536962920450

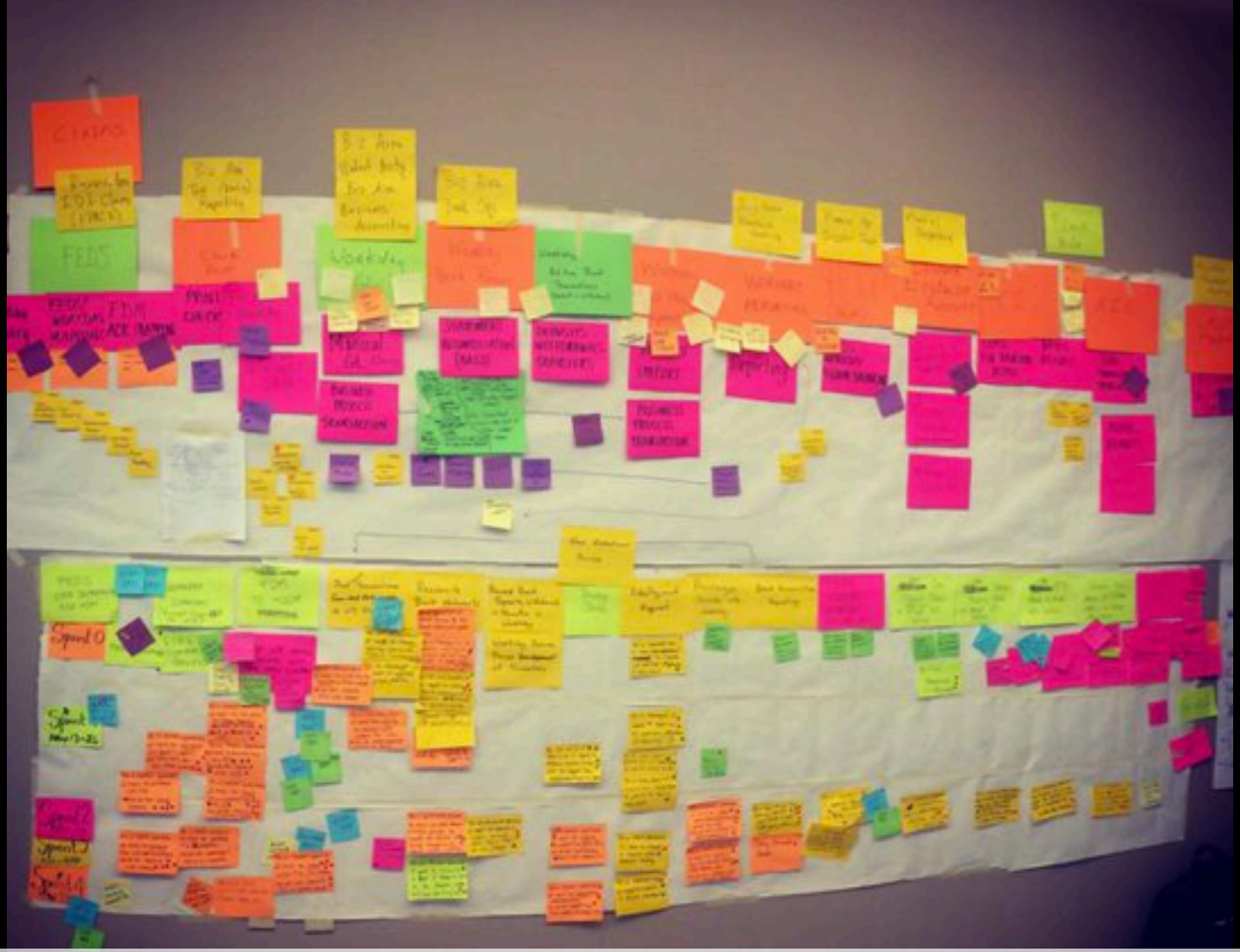
Story Mapping conversations wherever you're sitting via Christian Heldstab, @derdavoser

https://twitter.com/derdavoser/status/603921980103008256



I have no idea what Pat Maddox used to Map this, but he quickly moved it to a tool

https://twitter.com/patmaddox/status/603211750440243201



Miljan Bajic, @miljanbajic, Portland Maine

https://twitter.com/miljanbajic/status/599936891232919552

Team mapping via Adam Taylor @adamtaylo, London, UK

https://twitter.com/adamtaylo/status/583210182613864448

Paul Bellows, Edmonton, Alberta

https://twitter.com/paulbellows/status/606135867925618689

Photo courtesy of Kim Lees, Ecolab, St. Paul Minnesota

And, I forgot who sent me this one... but wow!

Photo courtesy of Boltmade, Waterloo, Ontario, Canada