

A GUIDE TO FARMING MIRACLES

MAKE YOUR HARD JOB EASY

Presented by Austin Govella at the Big Design conference in Dallas, TX, Friday, July 15th, 2011.

THE BACKCHANNEL

Use the hashtags to follow the conversation:
#miraclefarm or **#miraclefarmer**

Ask questions on the discussion forum:
<https://convore.com/freaks-of-the-industry/>

(Or catch me on Twitter: **@austingovella**)

THE BACKCHANNEL

Use the hashtags to follow the conversation: [#miraclefarm](https://twitter.com/miraclefarm) or [#miraclefarmer](https://twitter.com/miraclefarmer)

Ask questions on the discussion forum:
<https://convore.com/freaks-of-the-industry/>

(Or catch me on Twitter: [@austingovella](https://twitter.com/austingovella))

**Grab the slides from SlideShare
and follow along:**

<http://www.slideshare.net/austingovella/a-guide-to-farming-miracles>

Austin Govella

@austingovella

austin.govella@gmail.com

My blog:

www.thinkingandmaking.com

Blog post about the presentation:

<http://www.thinkingandmaking.com/view/ux-miracle-farming>

A Guide to Farming Miracles for UXers by Austin Govella • Big Design Conference, Dallas, TX, July 15, 2011 • www.thinkingandmaking.com • [@austingovella](https://twitter.com/austingovella) • [#miraclefarm](https://twitter.com/miraclefarm)

4

THE NEXTCHANNEL

If you have any questions, want to share your own tactics, or need help with your own horror story, get in contact.

Austin Govella
[@austingovella](https://twitter.com/austingovella)
austin.govella@gmail.com

My blog:
www.thinkingandmaking.com

Blog post about the presentation:
<http://www.thinkingandmaking.com/view/ux-miracle-farming>

THE MANIFESTO

Designers don't really design anything. Organizations design everything.

What if your organization sucks? Seriously. What do you do then? And then — while you're at it — do it “agile”, do it “lean”.

Organizations face seven barriers when designing better products and services: value, focus, time, memory, quality, understanding, and improvement. We'll look at seven tactics you can use to help overcome these seven barriers.

Instead of changing what you do, change how you do it.

Change the how and enable better design. You can build better, more balanced teams; better interfaces; and better experiences.

THE MANIFESTO

Over the last six years, I've been fascinated with how teams work together to create experiences. Much of this time was spent in organizations joining the agile development movement. Slowly, my personal practice as a user experience designer has evolved from focusing on what I can do to improve the experience to what we all can do to improve our organizations. During these last six years, I've discovered three driving principles:

1. Designers don't really design anything. Organizations design everything.

What if your organization sucks? Seriously. What do you do then? And then — while you're at it — do it “agile”, do it “lean”.

2. Organizations face seven barriers when designing better products and services:

value, focus, time, memory, quality, understanding, and improvement.

3. Instead of changing what you do, change how you do it.

Change the how and enable better design. You can build better, more balanced teams; better interfaces; and better experiences.

In this presentation, we'll look at seven tactics you can use to help overcome these seven barriers, ways to change how you do, and how to help your organizations and teammates create better experiences. Although these tactics were developed to help me work lean in an agile world, you'll find they're useful in almost every organization.

HOMework

due MONDAY

When the conference ends, you will be an **EXPERT MIRACLE FARMER**.

At every conference, designers commiserate. Whenever someone complains about clients, projects, or coworkers, ask yourself:

"WHAT BARRIER ARE THEY FACING?"

HOMework

Before we start, though, you have some homework.

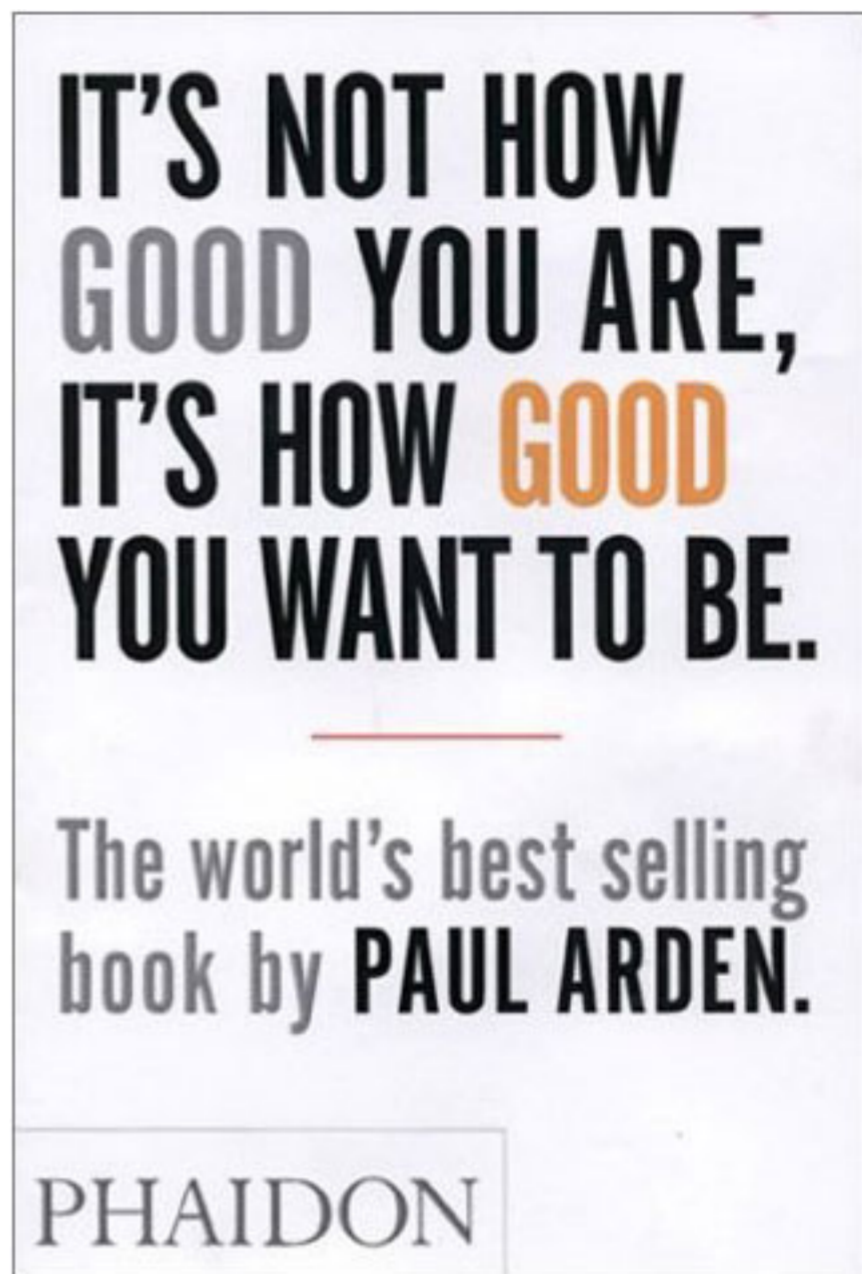
When the conference ends, you will be an **EXPERT MIRACLE FARMER**.

At every conference, designers commiserate. Whenever someone complains about clients, projects, or coworkers, ask yourself: **"WHAT BARRIER ARE THEY FACING?"**

Don't look for the next opportunity.

The one you have in hand is the opportunity.

— Paul Arden



A Guide to Farming Miracles for UXers by Austin Govella • Big Design Conference, Dallas, TX, July 15, 2011 • www.thinkingandmaking.com • @austingovella • #miraclefarm

7

THE INSPIRATION

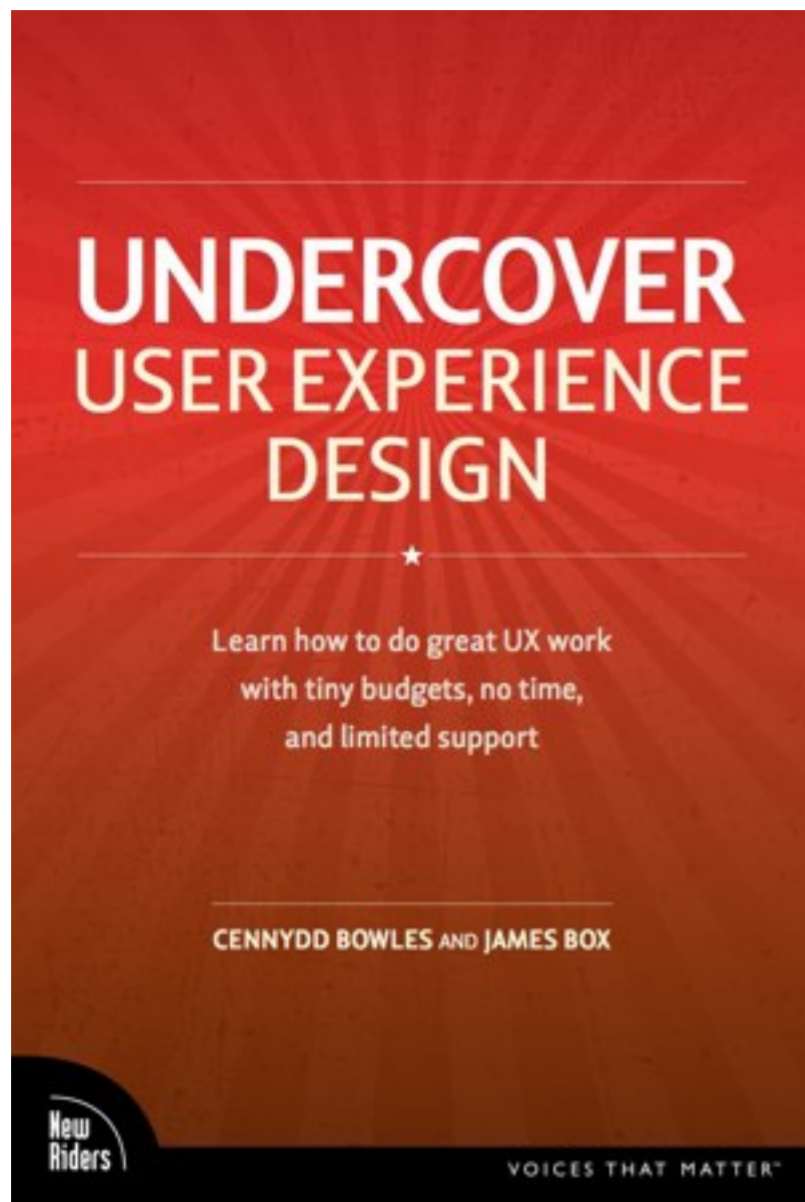
A lot of times, it's easy to get caught up in the day-to-day drudgery and difficulty at your current job. And it's easy to think, "if only I worked at that magical company in the sky that gets it, everything would be so much easier."

But that's really not true. Every organization has its own problems.

That's when I came across this quote from Paul Arden. If you haven't read his book, 'It's Now How Good You Are, It's How Good You Want To Be', you should pick it up right away. It's \$10 on Amazon, and although it's written for the advertising industry, it's packed full of advice every designer needs.

That's where I found this wonderful aphorism: "Don't look for the next opportunity. The one you have in hand is the opportunity."

And it's the truth. Stop fantasizing about how great UX is somewhere else, because it's not. UX can be great where you work now. These seven barriers and tactics can help you turn your organization into a better designing organization.



Undercover UX Design
Cennydd Bowles & James Box
New Riders, 2010



UX Team of One
Leah Buley
Rosenfeld Media, 2012

A Guide to Farming Miracles for UXers by Austin Govella • Big Design Conference, Dallas, TX, July 15, 2011 • www.thinkingandmaking.com • @austingovella • #miraclefarm

8

MORE THINKING

Before we get started, if you like what you see here, you can find more thinking along these lines.

Bowles and Box published 'Undercover User Experience Design' last year, and it features tips and techniques for sneaking in UX in ways that are hidden, fast, and cheap, so you can get UX in front of skittish teams.

Similarly, Leah Buley has a great presentation on being a "UX Team of One" you can find online or catch at a conference. She has a book on the subject coming out next year from Rosenfeld Media.

SEVEN BARRIERS

Organizations face seven barriers to designing better experiences.

The organization:

- ...doesn't **VALUE** design
- ...can't **FOCUS** on the design activities it needs to focus on
- ...doesn't have **TIME** to design everything it needs designed
- ...has no **MEMORY** about its design decisions
- ...has a low **QUALITY** of design by non-designers
- ...has no **UNDERSTANDING** about what it takes to do UX
- ...can't validate **IMPROVEMENT** in the user experience

SEVEN BARRIERS

Organizations face seven barriers to designing better experiences.

The organization:

- ...doesn't **VALUE** design
- ...can't **FOCUS** on the design activities it needs to focus on
- ...doesn't have **TIME** to design everything it needs designed
- ...has no **MEMORY** about its design decisions
- ...has a low **QUALITY** of design done by non-designers
- ...has no **UNDERSTANDING** about what it takes to do UX
- ...can't validate **IMPROVEMENT** in the user experience

We'll look at each of these barriers, one at a time. Then, for each barrier, we'll look at one tactic that can help you overcome it.

BARRIER 1

The organization
doesn't **VALUE** design.

A story about a brand
new UX department at
Comcast (the devil you
know is always better)

BARRIER 1: The organization doesn't VALUE design.

TRY: TELL A STORY

People understand the world through stories.
Tell a story about design that your coworkers
can use to understand what you do.

Good stories:

...are **TRUE**

...use **MEMORABLE** words and images

...remember something **IMPORTANT**

TRY: TELL A STORY

If your organization doesn't value design, it's not because they don't understand that design is valuable. Everyone knows it's valuable. We're surrounded with information about the value of design.

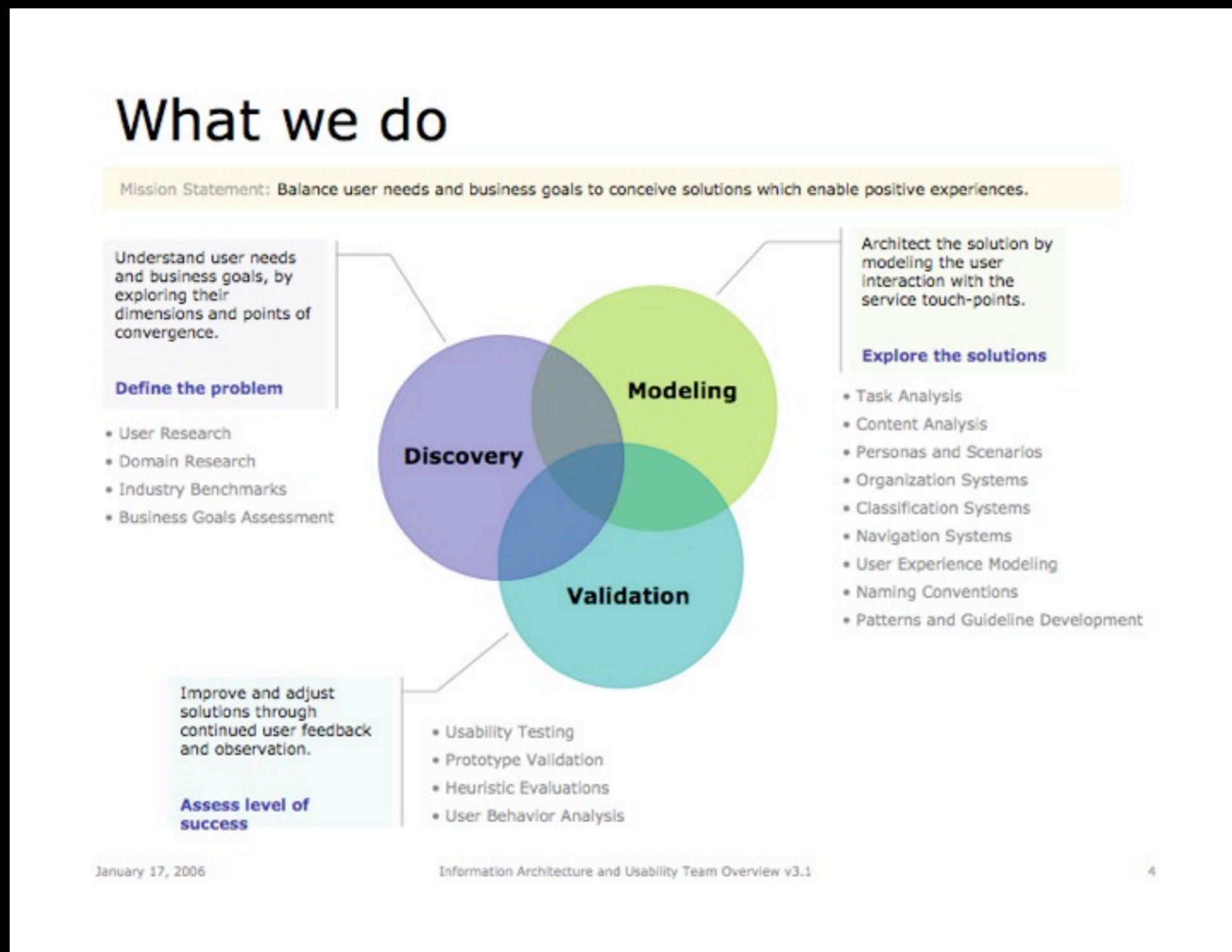
The problem is your organization doesn't easily understand how design can fit into their current process. If they're not clear on how it fits in, then when they start a project, it's easy to skip. The devils they know, like development and project management always get included because it's easy for them to understand how they fit into the project.

Instead of telling your coworkers about how valuable UX is, tell a story about how UX fits into their work flow. People understand the world through stories. Tell a story about design that your coworkers can use to understand what you do.

Good stories have three properties:

1. They're TRUE.
2. They use MEMORABLE words and images.
3. They remember something IMPORTANT.

EX: DISCOVER MODEL VALIDATE



By **Livia Labate** for Comcast Interactive Media, January 2006

A Guide to Farming Miracles for UXers by Austin Govella • Big Design Conference, Dallas, TX, July 15, 2011 • www.thinkingandmaking.com • @austingovella • #miraclefarm

12

EXAMPLE STORY: Livia Labate's "Discover, Model, Validate"

Livia Labate, the Director of IA at Comcast in 2006, created this story to communicate to other departments how her troupe of IAs could add value to their projects.

This is a great story, it's a true representation her department's capabilities. You can see three lists of specific activities people could choose from like a menu. Second, it's memorable. "Discover, model, validate" flows. It has a nice mnemonic ring, so effective that five years later, I still think about projects this way. Lastly, it's easily shared. She created this for an internal presentation she gave to other departments, but it works just as well as one sheet.

EX: UX HONEYCOMB



“The UX Honeycomb” by Peter Morville for Semantic Studios, June 2004


A Guide to Farming Miracles for UXers by Austin Govella • Big Design Conference, Dallas, TX, July 15, 2011 • www.thinkingandmaking.com • @austingovella • #miraclefarm

13

EXAMPLE STORY: Peter Morville’s “UX Honeycomb”

Morville’s UX honeycomb is a great depiction of the types of value a UX practitioner can provide. Morville has this great story about how he would actually use this as a menu to help his clients communicate what they needed help with.

EX: STUPID USERS



The only way to help **STUPID** users is **SMART** design.

It is an honest question: how smart are your users? The answer may surprise you: it doesn't matter. They can be geniuses or morons, but if you don't engage their intelligence, you can't depend on their brain power.

Far more important than their IQ (which is a questionable measure in any case) is their Effective Intelligence: the fraction of their intelligence they can (or are motivated to) apply to a task.

<i>If your user is</i>	<i>Then design for</i>
S tressed	S implicity
T ired	M emory
U ntrained	A utopilot
P assive	R ecovery
I ndependent	T esting
D istracted	

“Are your users S.T.U.P.I.D.” by Stephen Turbek, **Boxes and Arrows**, March 2011

A Guide to Farming Miracles for UXers by Austin Govella • Big Design Conference, Dallas, TX, July 15, 2011 • www.thinkingandmaking.com • @austingovella • #miraclefarm

14

EXAMPLE STORY: Stephen Turbek's “S.T.U.P.I.D. Users”

You don't have to create a story about the value of user experience. You can also create stories that share best practices or communicate product mental or concept models.

This recent example from Boxes and Arrows article condenses the entire article into one sheet you can hang on your cube to remind yourself about the issues that make your users “stupid” and ways you can help them stay “smart”.

END: TELL A STORY

Good stories are **TRUE**.

Good stories use **MEMORABLE** words and images.

Good stories remember something **IMPORTANT**.

Another tactic to explore:

Design vision (what one thought would you change in the minds of your coworkers)

END: TELL A STORY

Remember, stories have three important components:

Good stories are TRUE.

Good stories use MEMORABLE words and images.

Good stories remember something IMPORTANT.

(It's a waste of time crafting a story about something unimportant.)

BARRIER 2

The organization can't **FOCUS** on important design activities.

A story about the unsexy UX at Comcast.

TRY: UX BUCKET

Day-to-day user experience maintenance and improvements are never as exciting or important as this week's newest priority.

So, protect your tasks with a UX bucket.

INVIOABLE: only UX can schedule the bucket

RESPONSIVE: schedule what you want when you want

IRRESPONSIBLE: no one is responsible; unless you want them to be

Another tactic to explore:

Strict roles (split team between strategic and tactical design)

TRY: UX BUCKET

Day-to-day user experience maintenance and improvements are never as exciting or important as this week's newest priority. It's not uncommon for the UX team to have a laundry list of small fixes, tweaks, and glaring UX errors they want to see fixed, only to have the development team be all booked up rolling out whatever feature has the product manager all hot and bothered this week.

So, protect your tasks with a UX bucket. A UX Bucket is a reserved block of developer time during a sprint/release/whatever. For example, if your sprint is 100 story points or 100 hours, a UX bucket would be 20 hours reserved for UX work. It has three properties:

INVIOABLE: only UX can schedule the bucket

RESPONSIVE: schedule what you want when you want

IRRESPONSIBLE: no one is responsible; unless you want them to be

With the UX bucket, regardless of the latest priority or sexy new feature, you can always make sure to have time to fix and optimize the experience, or even experiment and research.

Here's how it works. When the sprint starts, a developer is assigned to the UX bucket and the block of time associated with the bucket is deducted from that developer's overall availability.

Now, you have a choice of what to do next. You can give that developer a sprint's worth of tasks, or — and this is what I recommend — give them a couple day's worth. That gives you a couple of days to decide what's next. Maybe you just go down your laundry list of tasks, or maybe after two days something else has come up. Maybe UX on another planned feature is lacking due to time, so you want the UX bucket developer to help out. Or maybe you want to iterate on the work done in the first two days.

Whatever. The UX bucket is your ultra-agile, super pivot-ey opportunity to work in code.

Agile purists will note this essentially violates the concept of an agile backlog, and they're right. The UX Bucket is a tactic that's necessary only on teams that do no value user experience as much as other development tasks. In organizations that can't focus on design activities because they're too focused on engineering activities, the UX Bucket provides an escape valve to make sure important UX activities don't fall by the wayside.

BARRIER 3

The organization doesn't have **TIME** to complete all designs.

A story about letting dead design lie at Convio.

TRY: SLOUGH WORK

If you don't have time to do it all, then don't.

Decide what UX you need to do. Leave the rest to everyone else.

WHAT'S CRITICAL? Important designs want your love.

WHAT'S AWESOME? If it can be awesome, make it so.

WHAT'S HARD? Complex designs need your love.

WHAT FITS? If it fits, do it. If it doesn't, don't.

Advantages:

Level-up your team: more practice, more review, better designers.

Demonstrate value: your UX will be distinctly better

Illustrate impact: the interface reveals time trade-offs

More tactics to explore:

Overbook yourself; More, faster; Doc systems

A Guide to Farming Miracles for UXers by Austin Govella • Big Design Conference, Dallas, TX, July 15, 2011 • www.thinkingandmaking.com • @austingovella • #miraclefarm

19

TRY: SLOUGH WORK

If you don't have time to do it all, then don't. Decide what UX you need to do. Leave the rest to everyone else.

This can seem like a scary position to take. This can feel like you're letting the team down. But you're not. When you have more work than you can do you have a couple of choices:

1. Do as much as you can, as well as you can.
2. Do everything asked, and do none of it well.
3. Do only what you can do well.

Which would you choose? Do what you can do well, or do everything, and try not to screw anything up? The professional answer is to only commit to what you can do well. The professional never overcommits and regardless of your UX hero delusions, you don't have super powers. If you over commit, you will under deliver. That is the definition of over committing. You feel like you need to over commit, so how can you respond in a productive fashion?

Try sloughing work. Only take on the projects that you can commit to executing well.

How do you choose what projects to take?

WHAT'S CRITICAL? Important designs want your love.
WHAT'S AWESOME? If it can be awesome, make it so.
WHAT'S HARD? Complex designs need your love.
WHAT FITS? If it fits, do it. If it doesn't, don't.

Sloughing work can generate the following advantages:

1. Level-up your team: the more practice they have, the more reviews, and the better designers they become.
2. Demonstrate value: your UX will be distinctly better than the rest, and this demonstrates the value of good UX.
3. Illustrate impact: the performance of the various interfaces will reveal the impact to spending time on UX versus other activities.

We're taught to say yes. Telling your manager or your team members or your manager's manager "no" can feel like you're letting the team down, like you're a failure, like you're not trying hard enough. Get over it. You're not. You're being a professional and professionally committing to what you can professionally deliver in a professional fashion.

In organizations that have more design to do than time and people to do it, the only thing you can do is slough work. So get lazy.

ACTIVITY

Stop. Close your eyes and think for a moment. Remember the worst, most frustrating project you've ever worked on.

As we continue, think about how you might have applied these tactics.

ACTIVITY

Stop. Think for a moment. Remember the worst, most frustrating project you've ever worked on.

As we continue, think about how you might have applied these tactics to to improve how that project evolved.

BARRIER 4

The organization has no **MEMORY** about its design decisions.

A story about too many cooks in the preferences kitchen.

TRY: DOCUMENT RATIONALE

If you can't remember, then forget it.

Record your rationale alongside your design so you don't need to remember.

Rationale includes:

WHY?

WHAT'S THE GOAL?

WHO'S IT FOR?

WHAT'S THE CONTEXT?

WHAT ARE THE CONSTRAINTS?

OPEN QUESTIONS?

TRY: DOCUMENT RATIONALE

If you can't remember why you made a design decision, then don't.

Record your rationale alongside your design so you don't need to remember.

Your design rationale includes:

WHY?

WHAT'S THE GOAL?


WHO'S IT FOR?

WHAT'S THE CONTEXT?

WHAT ARE THE CONSTRAINTS?

OPEN QUESTIONS?

EX: DOCUMENT CONTEXT




Application	Public website
Section	User Profile
View Title	Email Preferences
View Type	Content
Process Name	n/a
Process Step	n/a

The email preferences page displays the users current email preferences and allows them access to edit the preferences.

Persona

Connie the constituent
an active constituent



Goal

Connie wants to manage her email relationship with the organization.

Paths

Previous view

Page name	Call-to-action
User profile	Email preferenecs (link)

Next view

Page name	Call-to-action
Edit email preferences	Edit email preferences
Edit Center and/or interests preferences	[Center Name] email preferences
Edit site-wide interests	Email subscriptions

Capabilities

Constituent understands the email they receive (or have opted-out of receiving) from the client
This includes site-wide opt-in, site-wide interests, Centers, Center interests, and email format.

Constituent can edit site-wide email preferences
Includes site-wide opt-out, email format, and site-wide interests.

Constituent can edit Center-level email preferences
Includes Center opt-out and Center interests.

Problems we need to solve

Current page makes site-wide opt-out too prominent
We'd rather constituents opt-in to a sub-set of interests instead of opting out of everything.

Current page doesn't accurately communicate the constituent's opt-in and -out status
The database records three states: opt-in, opt-out, and null, but the interface uses two states (a checkbox) to show

Current page urges an interaction (save) when no interaction may be required
We conflate the act of viewing your preferences with editing them and have optimized the interface for editing. We use a form when we should be using text.

Current page does not accurately record user preferences
Because we can only show two of three possible states, we make assumptions about a users preferences when we save instead of saving exactly what they want.

Ex: DOCUMENT CONTEXT

In this example, I recorded the persona we were building for, the persona's goal, and previous and next pages (context).

In addition, on the right I listed the capabilities we needed to enable for the persona, as well as a list of problems we needed to solve.

Any time the design is discussed, it's easy to refer to this information to make sure new decisions stayed on target.

EX: DOCUMENT WHY

2.1 Video/Screencast On landing page. Promotes and explains how the DBC program works. Why? Videos increase engagement and conversion. A video will increase the number of who leverage the DBC promotion.	4.1 Add/Manage Students Form for adding/editing and managing students. 4.2 Send Referrals Form for selecting students and emailing them DBC referrals.	4.4 DBC checklist/getting started info Includes directions to check with the gym as well as recommendations on selecting students, managing free classes, and converting non-students to students. Why? Explicit guidance and checklists will increase the number of who participate. It will also make their participation more successful.	4.5 DBC tools Downloadable 11x17, 8.5x11, and 4-up posters and flyers can download to promote their classes or the DBC promotion. Why? These tools help gyms and better promote the program and will boost the number of ticket redemptions and conversions to students.
3. Admin Page Admin home screen where access Admin functionality. (Already exists?)	4.3 Add/Manage Classes Form for adding/editing classes. Facility adding/management accomplished on same screen.		

Ex: DOCUMENT WHY

The previous example required lots of typing. And I'm lazy. So, now I just add the word "why" and follow it with a statement of why I made this decision or recommendation.

In this example, I recommend we include a checklist and getting started info. Such a small thing, it seems easy to remove from the project when it comes time to start cutting features. Why include a checklist and getting started info? Because it increases participation, our main goal.

Sure, it's possible, the feature may still be cut, but now, at least, the team has an understanding of why the feature exists and they can extrapolate the impact to the product.

END: DOCUMENT RATIONALE

Record your rationale so everyone remembers.

Rationale includes (in order of importance:

WHY did you make this statement?

WHAT is the project's or the user's goal?

WHO did you design this for?

WHEN+WHERE is the interaction's context ?

WHAT are the constraints?

Do you have any **OPEN QUESTIONS**?

More tactics to explore:

Personas and sample users; Design goals

END: DOCUMENT RATIONALE

Record your rationale so everyone remembers. Rationale includes (in order of importance):

WHY did you make this statement or recommendation?

WHAT is the project's or the user's goal?

WHO did you design this for?

WHEN+WHERE is the interaction's context ?

WHAT are the constraints?

Do you have any OPEN QUESTIONS?

The rationale is more important than annotations about what it does or sort order or the transition you'd like to see. All of that information can be figured out by developers with little to no input. The rationale behind your design is easier to forget. It's also the most important part.

BARRIER 5

The organization's design **QUALITY** by others is too low.

A story about good engineers designing bad things.

TRY: BETTER STORIES

Developers deliver the user story they're given.
If it's not the story you expected, then you're reading different stories.

Stories have:

A MAIN CHARACTER The session attendee
EVENTS tweets a point from the presentation
OTHER ACTORS to their stream
MOTIVATION so they can share the point with their friends.

Shared stories have:

COMMON LANGUAGE The interaction should be like sharing something on Google+.

More tactics to try:

Sketching, Critiques, Guerilla testing, RITE testing, Patterns, Design seeds

TRY: BETTER STORIES

Developer aren't idiots. If they come back to you with something that doesn't match your expectations, it's not because they're lazy or don't care. It's because you and the developer have different expectations.

Developers deliver the user story they're given. If it's not the story you expected, then you're reading different stories.

Stories are important. They're how all humans have shared everything ever since we started speaking to each other.

Stories have:

A MAIN CHARACTER

EVENTS

OTHER ACTORS

MOTIVATION

A story might go like this:

A session attendee (the main character)

tweets a point from the presentation (the event)

to their Twitter stream (the other actors)

so they can share the point with their friends (their motivation)

That's the kind of story you can tell anyone, but odds are everyone will see something slightly different. And if you give that story to ten developers, you can bet you'll get back ten different interfaces.

Shared stories, the kind that everyone understand the same way, also share a common language. How do you improve the shared language in your development stories? Use a shared experience as a comparison.

We might adjust our story with one small line: "The interaction should be like sharing something on Google+."

Now, if you give ten developers that same story, you'll still have ten different interfaces, but they will all be really, really like sharing something on Google+. Adding comparisons to your development stories is an easy way to ensure the entire team is sharing the same design vision.

If you have developers designing things, and you don't like what they come back with, start adding comparisons so both of you have a similar idea. So the two of you have the same expectations.

BARRIER 6

The organization
doesn't **UNDERSTAND**
what it takes to do UX.

A story about bouncing
off the walls at Convio.

BARRIER 6: The organization's doesn't UNDERSTAND what it takes to do UX.

TRY: USE THE WALLS

Make the invisible, visible. Photocopy your next sketch. Print your next wireframe. Create your next sitemap out of Post-Its on a public wall.

Choose a wall where key influencers will pass by frequently.
Scribble edits on the pages on your wall.
Post iterations on top of the previous version.

Another tactic to try:
Case studies

TRY: USE THE WALLS

When you ask a developer to give you a form where users can edit their profiles, they go off and come back with a form that lets user edit their profiles. But you don't really have any idea what it took for them to do that work.

Developers see you the same way. You hide in your cube and emerge days later with a wireframe of a super simple, easy-to-use interaction. It's one wireframe. It doesn't seem like it should take as long as it did. They don't really have any idea about what it takes for you to do your work.

So how do you explain to your team members all of the effort and analysis it takes to do your job?

You're going to make the invisible, visible.

Choose a wall where key influencers will pass by frequently. You don't want it to be too high traffic because your work will fall off the walls. But you also don't want to choose a wall way back in the corner where no one will see anything.

Next, photocopy your next sketch and put it on the wall. Scribble edits on it, right there on the wall. When you do an iteration, stick the new version on the wall on top of the old version. Offset the two a little so anyone walking by can see there are two versions.

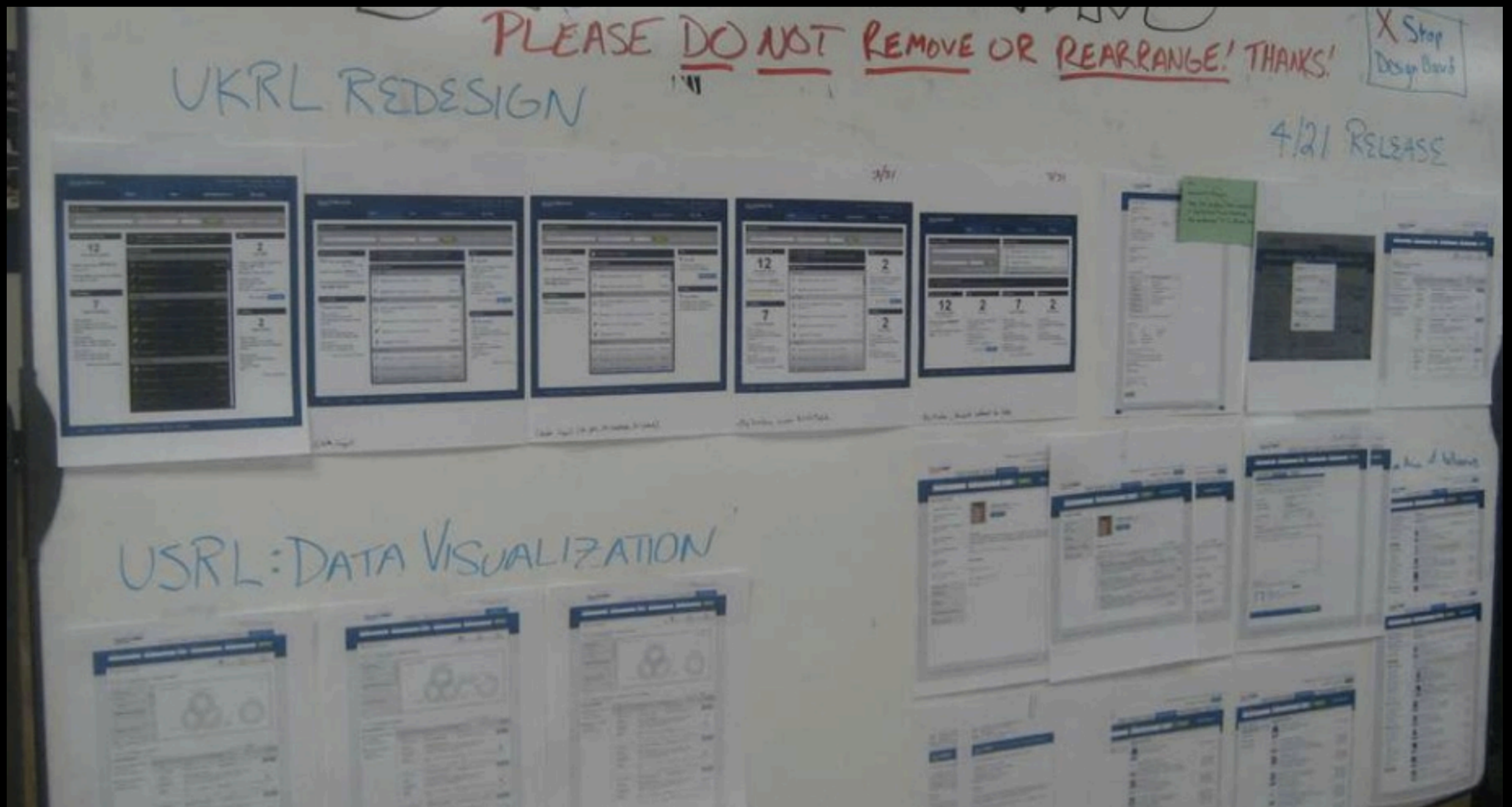
Print your next wireframe. When you review it with the developer, review it at the wall and have the developer scribble edits on it, right there on the wall.

Create your next sitemap out of Post-Its on a public wall. Create flows on the wall. They have these cool arrow-shaped Post-Its you can use to connect your sketches or wireframes into flows.

Do all of your work on the wall where everyone can see.

Demonstrate to your team members how many iterations you go through. Let them see how affinitization works. Invite them to participate by scribbling on your work, moving things around, posting alternate versions. Help them understand what it takes to do good UX work, and they'll help you later on by giving you the time, resources, and understanding you need to do the work you need to do.

EX: THE LADDERS



From **“Lean UX: getting out of the deliverables business”** by Jeff Gothelf, 2011

BARRIER 7

The organization has no way to **VALIDATE** UX improvement.

A story about product physicals at Comcast.

TRY: UX HEALTH CHECK

Evaluating the user experience is as easy said as done. The UX Health Check lets you evaluate any user experience.

Choose what you want to measure. (your features)

Choose your scale. (your comparison)

Set your target. (how good do you want to be?)

Measure yourself. (how good are you now?)

A Guide to Farming Miracles for UXers by Austin Govella • Big Design Conference, Dallas, TX, July 15, 2011 • www.thinkingandmaking.com • [@austingovella](https://twitter.com/austingovella) • [#miraclefarm](https://twitter.com/miraclefarm)

32

TRY: UX HEALTH CHECK

There are a million reasons why you can't validate your user experience. No analytics, not enough volume for analytics to be worthwhile, not enough time, or just no way to measure. That's all bullshit.

We'll try a mental exercise. The chair your sitting on right now: is it comfortable? Yes or no?

There. You just evaluated the user experience of your chair.

Evaluating the user experience is as easy said as done. The UX Health Check lets you evaluate any user experience.

Choose what you want to measure. (your features)

Choose your scale. (your comparison)

Set your target. (how good do you want to be?)

Measure yourself. (how good are you now?)

This is how it works. We're going to evaluate the user experience of your chair (our feature). We're going to compare it to the seats in movie theaters (our scale). We want our chair to be as comfortable as chairs in movie theaters (our target). Now measure.

It's as easy as asking the question: is my current seat as comfortable as the seat in a movie theater? Yes or no?

And you can set your target to whatever works best for your feature. You can want your registration process to be better than everyone else's. Bagcheck has a pretty sweet registration process. So ask yourself: is our registration process better than Bagcheck's? Or maybe you just want to be as good as a decent process. Set your target to be Google or Yahoo. Are you as good as Yahoo?

You always have a way to evaluate your user experience. Just do it.

EX: UX HEALTH CHECK

Capability	Description	Competitors	Target	Mar-09	Gap in Mar-09
Email/Web discussions	Members can discuss via email, or the web, or both.	IxDA List, Yahoo Groups, Google Groups	75	0	75
Body of knowledge	Compiled information, tutorials, and best practices	BA BOK, Usability BOK, About Face, Polar Bear book	75	0	75
Local directory	Members can easily find other nearby members	Highrise, Address Book (OSX)	90	28	62
Multi-language content	Important member content in more than one language	???	90	31	59
Interest Groups	Members can form groups around any interest	Yahoo Groups, Meetup, Barcamps	75	18	57
Local Groups	Members can form groups around any locale	Yahoo Groups, Yelp!, Churches	75	19	56
Non-English groups	Members can form groups that discuss in different languages	Mailman	90	39	51
Members only pages	Easy to restrict portions of the website to members only	LiveJournal, Chi.mp, LinkedIn	50	0	50

END: UX HEALTH CHECK

Evaluate every user experience.

Choose what you want to measure. (your features)

Choose your scale. (your comparison)

Set your target. (how good do you want to be?)

Measure yourself. (how good are you now?)

More tactics you can try:

System Usability Score, Experience value index

WRAPUP

SEVEN BARRIERS

Although organizations face
seven barriers to designing better experiences.

The organization:

...can **VALUE** design

...can **FOCUS** on the important design activities

...can find enough **TIME** to accomplish all design

...can have a **MEMORY** about its design decisions

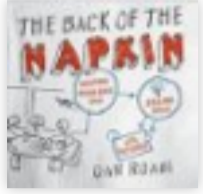
...can improve the **QUALITY** of design by others

...can have an **UNDERSTANDING** about what it takes to do UX

...can validate **IMPROVEMENT**



Made to Stick
Chip & Dan Heath, 2007



Back of the Napkin
Dan Roam, 2009



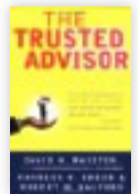
The Effective Executive
Peter Drucker, rev. 2006



Zag
Marty Neumeier, 2006



Rocket Surgery Made Easy
Steve Krug, 2009



The Trusted Advisor
David Maister, Charles Green, & Robert Galford, 2001



Designing With Agile
Anders Ramsey, 2011



Getting Real
Jason Fried, 2009



Storytelling for User Experience
Whitney Quesenbery & Kevin Brooks, 2010



Information Architecture: Blueprints for the Web
Christina Wodtke & Austin Govella, 2009

SHARE!

**Do you remember that
worst project ever?**

**Let's talk about your
horror stories.**

A NOTE ABOUT AGILE & LEAN

One of the hottest memes in user experience land continues to be agile+ux and its emerging, younger sibling, lean+ux.

As overworked, under-appreciated designers, it's your job to **make sure agile and lean don't become synonyms for creating skimpy, malnourished experiences.**

Instead of thinking agile and lean, think about how you can **create organizations with healthy user experience cultures.**

A NOTE ABOUT AGILE & LEAN

One of the hottest memes in user experience land continues to be agile+ux and its emerging, younger sibling, lean+ux.

As overworked, under-appreciated designers, it's your job to make sure agile and lean don't become synonyms for creating skimpy, malnourished experiences.

Instead of thinking agile and lean, think about how you can create organizations with healthy user experience cultures.

BALANCED TEAMS

The Balanced Team blog:

<http://www.balancedteam.org/>

Lean UX: getting out of the deliverables business

<http://uxdesign.smashingmagazine.com/2011/03/07/lean-ux-getting-out-of-the-deliverables-business/>

What is Lean UX?

<http://luxr.posterous.com/what-is-lean-ux>