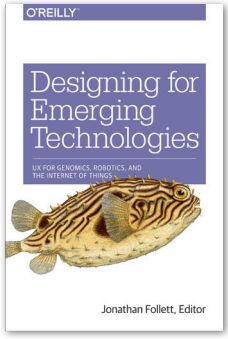
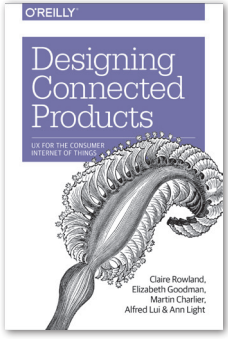
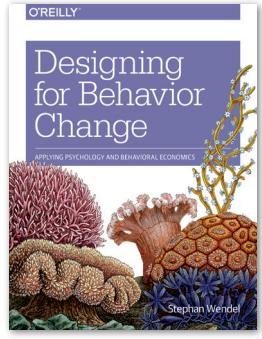
O'REILLY®

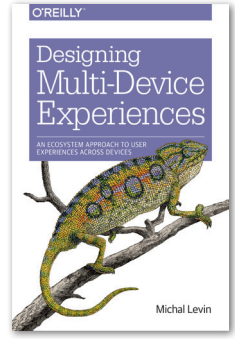
Experience Design

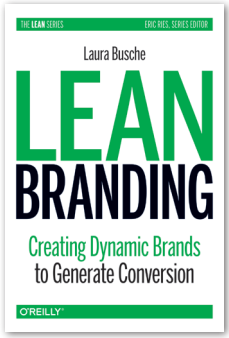
A Curated Collection of Chapters from the O'Reilly Design Library

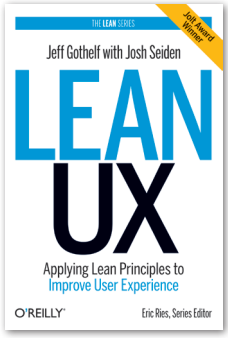


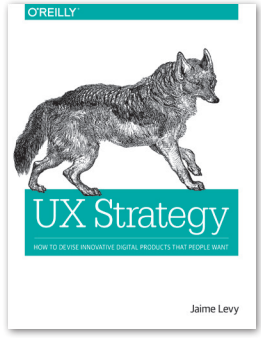


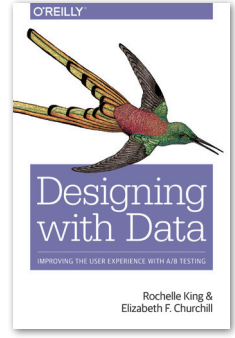












Short. Smart. Seriously useful.

Free ebooks and reports from O'Reilly
at **oreil.ly/fr-design**

Free ebooks, reports and other articles on UX design,
data-informed design, and design for the IoT.
Get insights from industry experts and stay current
with the latest developments from O'Reilly.

O'REILLY®

Experience Design

A Curated Collection of Chapters from the O'Reilly Design Library

As a designer, you know that continuously learning about the latest methodologies, tools, and techniques is critical to your success. O'Reilly Design books provide experienced designers with the knowledge and guidance to continue to build on your skillset. Whether you're interested in designing for Internet of Things (IoT), adopting Lean UX, or understanding the principles for creating user experiences in a multi-device ecosystem, there's something in this sample for you. We've selected titles from published and forthcoming books that span Process (Lean UX), Strategy, Data-Informed Design, Designing for IoT, Branding, Behavioral Economics, Designing Across Devices, and more.

For more information on current and forthcoming Design content, check out www.oreilly.com/design

Mary Treseler
Strategic Content Lead
mary@oreilly.com

Designing Connected Products

Available in Early Release: http://shop.oreilly.com/product/0636920031109.do
Chapter 1. What's Different About UX for the Internet of Things?
Chapter 8. Interface and Interaction Design

UX Strategy

Available in Early Release: http://shop.oreilly.com/product/0636920032090.do
Chapter 2. The Four Tenets of UX Strategy

Designing with Data

Available soon: http://shop.oreilly.com/product/0636920026228.do
Chapter 1. Data-Driven vs Data-Informed Design: Does It Matter?
Chapter 4. The Culture of Data

Lean Branding

Available now: http://shop.oreilly.com/product/0636920032106.do
Chapter 3. Brand Strategy

Lean UX

Available now: http://shop.oreilly.com/product/0636920021827.do
Chapter 5. MVPs and Experiments

Designing for Behavior Change

Available now: http://shop.oreilly.com/product/0636920030201.do
Chapter 3. Strategies for Behavior Change

Designing Multi-Device Experiences

Available now: http://shop.oreilly.com/product/0636920027089.do
Chapter 1. An Ecosystem of Connected Devices
Chapter 3. The Continuous Design Approach

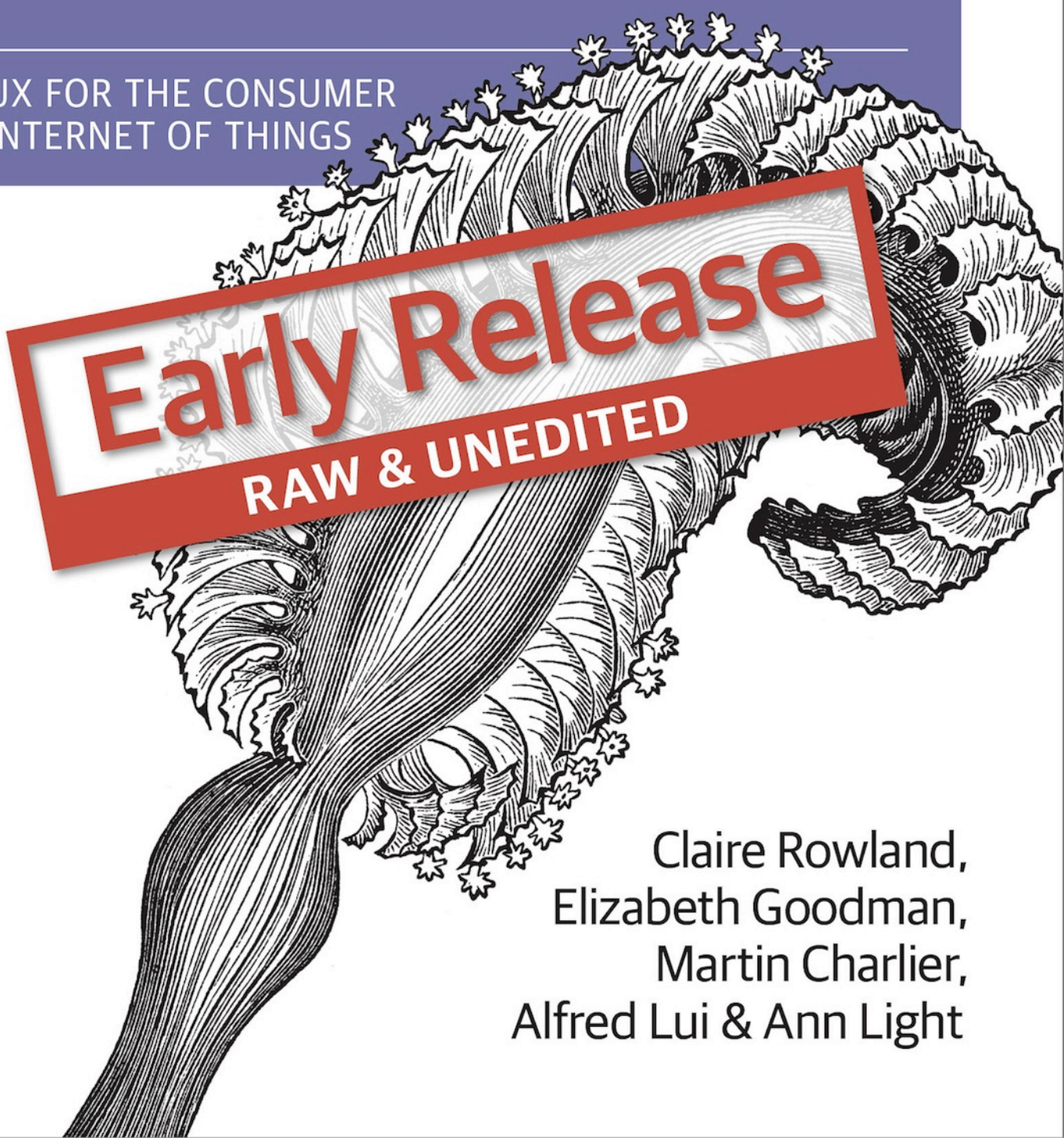
Designing for Emerging Technologies

Available soon: http://shop.oreilly.com/product/0636920030676.do
Chapter 1. Designing for Emerging Technologies
Chapter 2. Intelligent Materials: Designing Material Behavior

O'REILLY®

Designing Connected Products

UX FOR THE CONSUMER INTERNET OF THINGS



Early Release

RAW & UNEDITED

Claire Rowland,
Elizabeth Goodman,
Martin Charlier,
Alfred Lui & Ann Light

1

What's different about UX for the internet of things?

Introduction

UX design and human-computer interaction emerged in a world of desktop computing. But our experience of computing has changed radically in the last 10-15 years. Many of our interactions now take place on mobile phones, tablets, e-readers and smart TVs. And it's common to use one service across multiple devices with different form factors (figure 1-1).

Figure 1-1: BBC iPlayer can be used on connected TVs, games consoles, set top boxes, smartphones, tablets and desktop computers.

We're still figuring out the best ways to design for new devices and experiences. Interactions can happen in a wide variety of contexts, especially for mobile devices. They can happen on a variety of scales, such TV UIs (user interfaces) viewed from 10 feet away. Even academic researchers in HCI (human-computer interaction) have published relatively few papers on cross-platform design.

Designing for IoT raises all the challenges of cross-platform design, and more.

An obvious difference is the much wider variety of device form factors, many without screens (e.g. figure 1-2).

Figure 1-2: the Lockitron connected door lock is one of a huge number of connected devices with no screen.

Less obvious differences include the effects of many IoT devices being only intermittently connected. And even a simple task, like unlocking a door, can quickly become complex when it forms part of a system spanning many interconnected devices, services and users.

IoT is still a technically driven field. At the time of writing, the UX of many IoT products is some way off the level expected of mature consumer products. For

example, the UK government commissioned a study on the usability of connected heating systems in late 2013. They found that none of the 5 major connected heating devices on the market in the UK offered a good UX[1].

In this chapter, we begin by introducing the differentiators that make UX design for IoT a new and challenging domain.

This chapter introduces:

- What's different about UX for IoT
- A design model for IoT

It considers the following issues:

- The challenges of distributing functionality across multiple devices *(see page 3)*
- How the focus of the UX is increasingly in the service *(see page 3)*
- Whether we are ready for the real world to start behaving like the internet *(see page 3)*
- How the ways devices connect to the network affects the UX *(see page 4)*
- How multiple devices create more complexity for the user to understand *(see page 5)*
- How controlling distributed devices is similar to programming *(see page 6)*
- The problems of having many different technical standards *(see page 7)*
- How what seem like simple systems can rapidly become complex *(see page 7)*
- How data is at the core of many IoT services *(see page 8)*
- The layers of UX thinking required to create a successful IoT product: from UI and interaction design all the way down to the platform *(see page 9)*

How is UX for IoT different?

Designing for IoT comes with a bunch of challenges that will be new to designers accustomed to pure digital services. How tricky these challenges prove will depend on:

- The maturity of the technology you're working with
- The context of use or expectations your users have of the system
- The complexity of your service (e.g. how many devices the user has to interact with).

[1] Amberlight Partners for the Department of Energy and Climate Change, 2013: 'Usability testing of smarter heating controls'. https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/266220/usability_testing_smarter_heating_controls.pdf

Below is a summary of the key differences between UX for IoT and UX for digital services. Some of these are a direct result of the technology of embedded devices and networking. We'll explain the technology issues in more detail in chapters 2 and 3. But even if you are already familiar with embedded device and networking technology, you might not have considered the way it shapes the UX.

Functionality can be distributed across multiple devices with different capabilities

IoT devices come in a wide variety of form factors with varying input and output capabilities. Some may have screens, such as heating controllers or washing machines (see figure 1.3). Some may have other ways of communicating with us (such as flashing LEDs or sounds) (see figure 1.4).

Figure 1-3: The Honeywell evohome connected radiator valve has a basic LCD screen

Figure 1-4: The GlowCaps connected pill bottle lid uses light and sound notifications to remind the user to take medication

Some may have no input or output capabilities at all and are unable to tell us directly what they are doing. Interactions may be handled by web or smartphone apps. Despite the differences in form factors, users need to feel as if they are using a coherent service rather than a bunch of disjointed UIs. It's important to consider not just the usability of individual UIs but *interusability*: distributed user experience across multiple devices (e.g. figure 1-5). *This is explained further in chapter 10: Cross-device Interactions and Interusability.*

Figure 1-5: The Nest thermostat can be controlled by the on-device UI, a smartphone app or a web app

The locus of the user experience may be in the service

Although there's a tendency to focus on the novel devices in IoT, much of the information processing or data storage often depends on the internet service. This means that the service around a connected device is often just as critical to the service, if not more so, than the device itself. For example, the London Oyster travelcard is often thought of as the focus of the payment service. But the Oyster service can be used without a card at all via an NFC enabled smartphone or bank card (figure 1-6). The card is just an 'avatar' for the service (to borrow a phrase from the UX expert Mike Kuniavsky)[2]. *For more on service business models see chapter 4: Product/service definition and strategy. Service design is covered in chapter 7.*

[2] Mike Kuniavsky, 'Smart Things: Ubiquitous Computing User Experience Design', Morgan Kaufmann 2010

Figure 1-6: The London Oyster service can be used with an NFC phone or bank card instead of an Oyster card.

We don't expect internet-like failures from the real world

It's frustrating when a web page is slow to download or a Skype call fails. But we accept that these irritations are just part of using the internet. By contrast, real world objects respond to us immediately and reliably.

When we interact with a physical device over the internet, that interaction is subject to the same latency and reliability issues as any other internet communication. So there's the potential for delays in response and for our requests and commands to go missing altogether. This could make the real world start to feel very broken. Imagine if you turned your lights on and they took 2 minutes to respond, or failed to come on at all.

In theory, there could be other unexpected consequences of things adopting internet-like behaviors. In the Warren Ellis story 'The Lich House'[3] a woman is unable to shoot an intruder in her home: her gun cannot contact the internet for the authentication that would allow her to fire it. This might seem far-fetched, but we already have objects that require authentication, such as Zipcars (figure 1-7).

For more information on networking and its impact on UX, see chapter 3 on Networks.

Figure 1-7: When you book a Zipcar online, the service sends details of the reservation to the car. Swiping your card authenticates you as the person who made the booking.

IoT is largely asynchronous

When we design for desktops, mobiles and tablets, we tend to assume that they will have constant connectivity. Well-designed mobile apps handle network outages gracefully, but tend to treat them as exceptions to normal functioning. We assume that the flow of interactions will be reasonably smooth, even across devices. If we make a change on one device (such as deleting an email), it will quickly propagate across any other devices we use with the same service.

Many IoT devices run on batteries, and need to conserve electricity. Maintaining network connections uses a lot of power, so they only connect intermittently. This means that parts of the system can be out of sync with each other, creating discontinuities in the user experience. For example, your heating is set to 19C. You use the heating app on your phone to turn it up to 21C. But it takes a couple of minutes for your battery powered heating controller to go online to check for new instructions. During this time, the phone says 21C, and the controller says 19C

[3] http://www.iftf.org/fanfutures/ellis/

(figure 1-8).

Figure 1-8: schematic of heating system with app and controller giving different status information.

For more on the technical background see chapters 2 and 3 on Devices and Networks. For more on the design impact see chapter 10: Cross device interactions and interusability.

Code can run in many more places

The configuration of devices and code that makes a system work is called the *system model*. In an ideal world, users should not have to care about this. We don't need to understand how a conventional internet services, like Amazon, works in order to use it successfully. But as a consumer of an IoT service right now, you can't always get away from some of this technical detail.

A typical IoT service is composed of:

- one or more embedded devices
- a cloud service
- perhaps a gateway device
- one or more control apps running on a different device, such as a mobile, tablet or computer.

Compared to a conventional web service, there are more places where code can run. There are more parts of the system that can, at any point, be offline. Depending on what code is running on which device, some functionality may at any point be unavailable.

For example, imagine you have a connected lighting system in your home. It has controllable bulbs or fittings, perhaps a gateway that these connect to, an internet service and a smartphone app to control them all (e.g. see figure 1-9). You have an automated rule set up, to turn on some of your lights at dusk if there's no one home. If your home internet connection goes down, does that rule still work? If the rule runs in the internet service or your smartphone, it won't. If it runs in the gateway, it will. As a user, you want to know whether your security lights are running or not. You have to understand a little about the system model to understand which devices are responsible for which functionality, and how the system may fail.

Figure 1-9: The Philips Hue system consists of connected bulbs, a gateway, an internet service and a smartphone app

It would be nice if we could guarantee no devices would ever lose connectivity, but that's not realistic. And IoT is not yet a mature set of technologies in the way that ecommerce is, so failures are likely to be more frequent. System designers have to ensure that important functions (such as home security alarms) continue to work as

well as possible when parts go offline and make these choices explicable to users.

For more on matching system models to users' conceptual models see chapter 6: Translating research into product definitions.

Devices are distributed in the real world

The shift from desktop to mobile computing means that we now use computers in a wide variety of situations. Hence, mobile design requires a far greater emphasis on understanding the user's needs in a particular *context of use*. IoT pushes this even further: computing power and networking is embedded in more and more of the objects and environments around us. For example, a connected security system can track not just whether the home is occupied, but who is in, and potentially video record them. Hence the social and physical contexts in which connected devices and services can be used is even more complex and varied.

Remote control and automation are programming-like activities

In 1982, the HCI researcher Ben Shneiderman defined the concept of *direct manipulation*. User interfaces based on direct manipulation "depend on visual representation of the objects and actions of interest, physical actions or pointing instead of complex syntax, and rapid incremental reversible operations whose effect on the object of interest is immediately visible. This strategy can lead to user interfaces that are comprehensible, predictable and controllable."[4] Ever since, this has been the prevailing trend in consumer UX design (see e.g. figure 1-10). Direct manipulation is successful because interface actions are aligned with the user's understanding of the task. They receive immediate feedback on the consequences of their actions, which can be undone.

Figure 1-10: MacPaint was an early example of a popular direct manipulation interface

IoT creates the potential for interactions that are displaced in time and space: configuring things to happen in the future, or remotely. For example, you might set up a home automation rule to turn on a video camera and raise the alarm when the house is unoccupied and a motion sensor is disturbed. Or you might unlock your porch door from your work computer to allow a courier to drop off a parcel.

Both of these break the principles of direct manipulation. To control things that happen in future, you must anticipate your future needs and abstract the desired behavior into a set of logical conditions and actions. As the HCI researcher Alan

[4] This text quoted from Ben Shneiderman, 'Direct Manipulation for Comprehensible, Predictable and Controllable User Interfaces, Proc. ACM International Workshop on Intelligent User Interfaces '97, ACM, New York, NY (1997), 33-39

Blackwell points out, this is basically programming[5]. It is a much harder cognitive task than a simple, direct interaction. That's not necessarily a bad thing, but it may not be appropriate for all users or all situations. It impacts usability and accessibility.

Unlocking the door remotely is an easier action to comprehend. But we are distanced from the consequences of our actions, and this poses other challenges. Can we be sure the door was locked again once the parcel had been left? A good system should send a confirmation, but if our smartphone (or the lock) lost connectivity we might not receive this.

For more information, see chapter 13: Designing complex interoperable products and services.

Complex services can have many users, multiple UIs, many devices, many rules and applications

A simple IoT service might serve only one or two devices: e.g. a couple of connected lights. You could control these with a very simple app. But as you add more devices, there are more ways for them coordinate with one another. If you add a security system with motion sensors and a camera, you may wish to turn on one of your lights when the alarm goes off. So the light effectively belongs to two functions or services: security, and lighting. Then add in a connected heating system that uses information from the security system to know when the house is empty. And assume that there are several people in the house with slightly different access privileges to each system. For example, some can change the heating schedule, some can only adjust the current temperature. Some have admin rights to the security system, some can only set and unset the alarm. What started out as a straightforward system has become a complex web of interrelationships.

For a user, understanding how this system works will become more challenging as more devices and services are added. It will also become more time consuming to manage.

For more information, see chapter 13: Designing complex interoperable products and services.

Many differing technical standards make interoperability hard

The internet is an amazing feat of open operating standards, but, before embedded devices were connected, there was no need for appliance manufacturers to share common standards. As we begin to connect these devices together, this lack of common technology standards is causing headaches. Just getting devices talking to one another is a big enough challenge, as there are many different network standards. Being able to get them to co-ordinate in sensible ways is an order of magnitude more

[5] Personal communication. See Blackwell, A.F. (2002). What is programming? In *Proceedings of PPIG 2002*, pp. 204-218.

complicated.

The consumer experience right now is of a selection of mostly closed, manufacturer specific ecosystems. Devices within the same manufacturer's ecosystem, such as Withings, will work together. But this is the only given. In the case of Withings, this means that devices share data with a common internet service, which the user accesses via a smartphone app (see figure 1-11). Apple's Airplay is an example of a proprietary ecosystem in which devices talk directly to each other.

Figure 1-11: The Withings ecosystem of devices.

We're starting to see manufacturers collaborating with other manufacturers too. So your Nest Protect smoke detector can tell your LIFX lightbulbs to flash red when smoke is detected. (This is done by connecting the two manufacturer's internet services rather than connecting the devices).

There are also some emerging platforms which seek to aggregate devices from a number of manufacturers and enable them to interoperate. The connected home platform Smart Things supports a range of network types and devices from manufacturers such as Schlage and Kwikset (door locks), GE and Honeywell (lighting and power sockets), Sonos (home audio), Philips Hue, Belkin and Withings (see figure 1-12). But the platform has been specifically configured to work with each of these. You cannot yet buy any device and expect it to work well with a platform such as Smart Things.

Figure 1-12: The Smart Things gateway and some compatible devices.

For the near future, the onus will be largely on the consumer to research which devices work with their existing devices before purchasing them. Options may be limited. In addition, aggregating different types of device across different types of network tends to result in a lowest common denominator set of basic features. The service that promises to unify all your connected devices may not support some of their more advanced or unique functions: you might be able to turn all the lights on and off but only dim some of them, for example. It will be a while before consumers can trust that things will work together with minimal hassle.

For more information, see chapter 13: Designing complex interoperable products and services.

IoT is all about data

Networked, embedded devices allow us to capture data from the world that we didn't have before, and use it to deliver it better services to users. For example, drivers looking for parking spaces cause an estimated 30% of traffic congestion in US cities. Smart parking applications such as Streetline's Parker use sensors in parking spaces to track where spaces are free, for drivers to find via a mobile app (see figure 1-13). Opower uses data captured from smart energy meters to suggest ways in which customers could save energy and money (see figure 1-14).

Figure 1-13: Streetline's Parker app

Figure 1-14: Sample Opower energy report.

Networked devices with onboard computation are also able to use data, and in some cases act on it autonomously. For example, a smart energy meter can easily detect when electrical activity is being used above baseload. This is a good indicator that someone is in the house and up and about. This data could be used by a heating system to adjust the temperature or schedule timing.

To quote another phrase from Mike Kuniavsky, "information is now a design material."[6]

See chapter 12: designing with data.

A design model for IoT

As shown above, designing for IoT will confront you with some extra challenges and complexity that you wouldn't encounter on a purely digital service. You'll need to think about some different and perhaps new areas of design that all serve to shape the UX.

The two most visible and tangible forms of design for IoT are:

- The UI/visual design: e.g. the screen layout and look and feel of the web or mobile apps, or devices themselves (see figure 1-15). (UIs don't have to be visual, they can use audio, haptics and other channels as discussed in chapter 9. But it's rare for a service to have no screen based UI at all).
- The industrial design of the physical hardware: the form factor, styling and capabilities of the connected devices themselves (see figure 1-16).

Figure 1-15: The Philips Hue UI allows users to change the colour of light emitted by an LED bulb.

Figure 1-16: The Nest thermostat has a striking industrial design

UI and industrial design are important but not the whole picture. The UX is not just shaped by what the user can see or encounter directly. To create a valuable, appealing, usable and coherent IoT service we have to consider design on many different layers.

[6] Mike Kuniavsky, 'Smart Things: Ubiquitous Computing User Experience Design', Morgan Kaufmann 2010

In 2000, Jesse James Garrett produced his 'Elements of User Experience' diagram (and subsequent book) to explain how different design specialties fit together in web UX design[7]. This represented the different types of design required, where uppermost layers (i.e. visual design, information, interface and navigation design) are most visible to the user, but depend on the structure provided by the lower layers (e.g. site objectives, content requirements), which are dealt with earlier in the project. Just as there were dependencies in his model, where work that was not directly apparent to the user determined aspects that they could directly experience, so designing for connected products has a similar flow, with critical decisions that affect UX made in early stages of the design.

If you're more familiar with engineering models, you might think of this as being a little like a technology stack (as in the internet network stack in chapter 3), in which each layer of technology is dependent on the lower levels functioning well. It's not an exact analogy as the dependencies between different layers of design aren't as rigid. But it's a useful comparison to make the point that good design at the higher, more concrete layers requires a clear framework at the lower layers.

IoT encompasses a broad spectrum of design specialties. Figure 1-17 sets out the 'big picture' of different types of design involved in creating an IoT service.

[7] New edition: Jesse James Garrett, 2010: 'The Elements of User Experience: User-Centered Design for the Web and Beyond (2nd Edition)' 2010

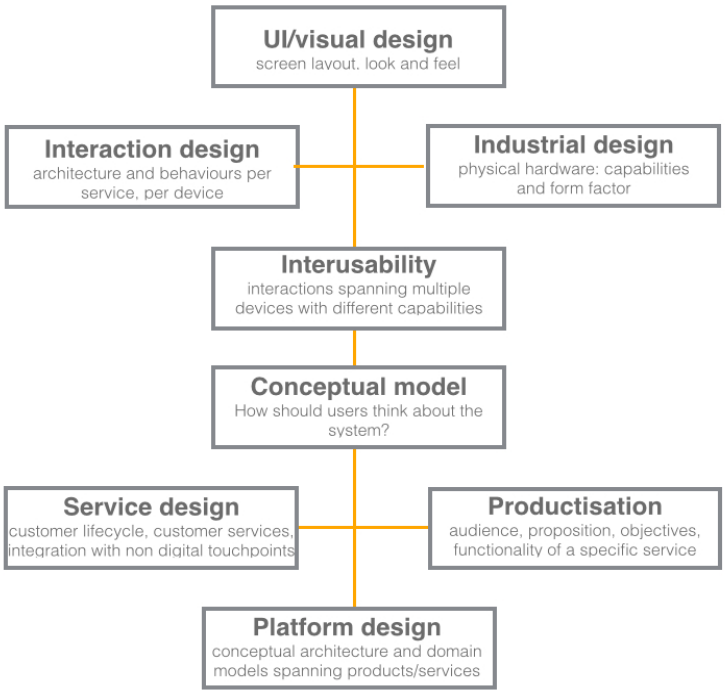


Figure 1-17: The 'design stack' for IoT

This isn't a set of discrete activities required in your project plan. It says nothing of user research, for example. Nor is it a set of job roles you need on your project. You might, for example, need data analytics, and you'll certainly need engineers!

It's aspects of the user experience that need to be considered. Some of these things will evolve in tandem. For example, UI, interaction design and interusability need to be thought about together. UX design at the platform layer will emerge as a need once you start adding multiple devices to a service.

A good overall product requires integrated thinking across all these layers. A stunning UI means nothing if your product concept makes no sense. A beautiful industrial design can't mask terrible service.

Depending on the type and complexity of your service, layers will require more or less of your time. IoT services aspire to extend over more devices and become more complex over time, and the parts you need less now may become more relevant to you in future.

UI/visual design

UI/visual design refers to screen layout, visual styling and look and feel on a device. This is the *form* that a device interface takes. The outputs of UI/visual design are typically high fidelity screen mockups (figure 1-18). Not all UIs are visual, of course: for a gestural or audio interface, the equivalent function might be defining the

aesthetics of the gestures or voice.

Figure 1-18: visual design moodboard and UI scamps.

See chapter 9, Interface and interaction design, for more details on designing UIs for embedded devices.

Interaction design

Interaction design is the design of device behaviors. Interaction designers shape the sequences of actions between the user and the device needed to achieve particular goals or activity. They also determine how to organize the user-facing functions of the device. For example, a heating controller might have several modes (such as schedule on/off or frost protection) and some hierarchical functions, such as schedule setting. The organization of these functions defines how easy or otherwise it may be for users to find their way around them.

Interaction design is closely aligned to UI design in the sense that the two are usually done in tandem and often by the same people. But interaction design is primarily concerned with behaviors and actions, whereas UI/visual design is concerned with layout and aesthetics. (Just to confuse matters, some people use UI design as a shorthand term to include both interaction design and visual design). Typical outputs for interaction design might include user flows, low-medium fidelity interactive prototypes and, for a visual UI, screen wireframes (figure 1-19).

Figure 1-19: interaction design deliverables

You sometimes hear the term ‘information architecture’ used to describe organization schemes for functionality, but technically this refers to the equivalent activity for content-driven systems, such as content-based web sites.

See Chapter 9, Interface and interaction design.

Interusability

Interusability is a relatively new term. It refers to the additional considerations of designing interactions that span multiple devices. The goal is to make the overall experience feel like a coherent service, even when the devices involved may have quite different form factors and input/output capabilities.

Interusability isn’t a separate set of design activities. It’s an extra set of considerations to be addressed in tandem with interaction and UI design. The key differences to a single device UX design process would typically be:

- Specifying which functionality belongs on each device
- Creating design guidelines that span multiple device types
- Designing cross-device user flows for key interactions
- Designing multiple device UIs in parallel

See chapter 10, Cross-device interactions and interusability.

Industrial design

Industrial design refers to the aesthetic and functional design of the physical hardware in the service: the choice of form, materials, and capabilities it may have (see figure 1-20). Connected devices contain electronic circuitry and radio antennae, which impose particular requirements on the industrial design. Devices can also have input and output capabilities, which require collaboration between industrial designers and UI/interaction design/interusability.

See chapter 8, Embedded device design.

Figure 1-20: industrial design process artefacts

Service design

A connected device is rarely a one-off purchase. It comes with the expectation of an ongoing service, at the very least the provision of the internet service that keeps it running, and customer support. All of this forms part of the user's overall experience with the product.

Service design is an emerging discipline which addresses this holistic view of user experience. It looks at the whole lifespan of a user's experience with a service, provides a view of all the components of the user experience, and specifies how these function together as a coherent whole (see figure 1-21).

As well as device interactions it might include:

- Customer support interactions
- Instructional guides
- Marketing or sales materials
- In-store experiences
- Email communications and notifications
- The UX of software updates and rolling out new functionality

See chapter 7, Designing the service.

Figure 1-21: service design process artefacts

Conceptual model

The conceptual model is the understanding and expectations you want the user to have of the system. What components does it have, how does it work, and how can they interact with it? It's the mental scaffolding that enables users to figure out how to interact with your service. Regardless of whether a conceptual model is designed or not, users will form one. If they get it wrong, they'll struggle to use the system. IoT services are often inherently complex systems. You can create a clear conceptual

model through careful system and interaction design and supporting documentation. You want users to feel in control from the start: confident that they will be able to use the system, even if they don't understand all the details yet.

See chapter 6, Translating research into product definitions

Productization

Productization is the activity of defining a compelling product proposition. It addresses the audience, proposition, objectives and overall functionality of service (and often its business model). Does your product solve a real problem for a real audience? Is it presented so that they understand that? Does it appeal to them? This isn't always the domain of the UX designer on a project but it's the underpinnings of good UX. All the front-end design in the world won't make a killer product unless it does something of value for people, in a way that appeals to them and makes sense.

See chapter 4, Product/service definition and strategy

Platform design

A platform is a software framework. It takes care of low-level details to help developers build applications more easily. IoT platforms, such as Xively or Thingworx, make it easy to put data from connected devices onto the internet and enable devices to interoperate with each other.

A software platform will aim to solve many technical issues, many of which may not directly have an impact on UX. For example, they may provide a way of collating device data in one place. But some platform functionality is very much to do with UX.

For example, a platform may provide standard ways to:

- Discover new devices and applications
- Add them onto the system
- Manage devices and users, and
- Manage how devices share data

These are basic building blocks for the UX. If they don't work well for your users, your UI and interaction design will be full of awkward workarounds.

A more complex platform might also provide ways of organizing and coordinating multiple devices. For example, a user adds a light to an existing home system. They might reasonably expect the system to know that it should control it along with their other lights and/or offer it as part of the security system. It's not important to make it talk to the toaster. That may be common sense to a human. But the system won't know that unless this kind of logic is encoded in the platform.

Many IoT systems and services aren't this complex yet, so you might not need to worry about platform design for some time to come. But in any system of

interconnected devices, there will be design challenges that will require platform logic to solve. Designers should get involved in shaping platforms to ensure they support good higher-level UX.

There is no commonly understood set of activities for this yet. Chapter 14, 'Designing complex interconnected products and services', discusses what these challenges may look like and some of the approaches we might take to solving them.

Chapter summary

UX for connected devices is not just about UI and interaction design. It also requires designers to think about: interusability, industrial design, service design, conceptual models, productization and platform design.

In summary, it differs from ‘conventional’ UX in the following ways:

Embedded devices often save power by connecting only intermittently	which means...	parts of the system can be out of sync, creating discontinuities in UX
Latency on the internet is out of your control (and reliability is not 100%)	which means...	although we expect physical things to respond immediately and reliably, this might not happen
Code can run in many more places	which means...	users have to engage with the system model to predict how it will work if parts are offline
Devices are distributed in the real world	which means...	social and physical context of use is complex and varied
Functionality can be distributed across multiple UIs	which means...	designers need to consider not just usability but interusability
Much of the information processing happens in the internet service	which means…	the service experience is often equally or more important than the single device UX
Remote control and automation are programming-like activities	which means...	IoT breaks direct manipulation: the basis of most successful consumer UXes
Many differing technical standards	which means...	getting things to work together is hard
Complex services can have many users, many UIs, many devices, many rules and applications	which means...	understanding and managing how they all interrelate can be extremely difficult. Users will turn off if admin becomes too onerous.

IoT enables us to capture and act on data we didn't have before	which means…	designers need to understand how to use information as a design material

8

Interface and interaction design

In this chapter we will look at interface and interaction design for connected products. Many connected products involve web and smartphone interfaces. But there are many good resources available on these platforms already. This chapter will focus on the possibilities beyond point-and-click or touchscreen interfaces. This has two reasons:

First, point-and-click and touchscreen interfaces dominate the majority of UX design work. But both interface types are less likely to be used on embedded devices like smart thermostats and or smart plugs.

Second, the Internet of Things provides an opportunity to redefine how we interact with digital devices.

This second point is brilliantly argued by designer Bret Victor in his "Brief rant about the future of interaction design". Victor discusses the diverse and intricate ways we use our hands on a daily basis. He critiques the many concept explorations that, although looking into the future, still revolve around touch screens. The aim of this chapter is to inspire designers to experiment with new ways of interacting with products.

This chapter introduces:

- The benefits and drawbacks of different interface types and interaction paradigms.
- Design recommendations and best-practice examples of interface and interaction design
- Specific interface design challenges for embedded devices.

This chapter addresses the following issues:

- Why a screen isn't necessarily better than no screen.
- What strategic decisions designers face about the interactivity of a device.
- How devices can receive complex input through simple interfaces.
- How interfaces can reduce the amount of attention asked from users.

- How interfaces can be kept accessible or universal.

Types of interaction

Our bodies and abilities define how we can 'talk' to devices (input). Our sensory system defines how we can 'listen' to devices (output).

Yet, the way we interact with devices has so far been very limited. Tom Igoe and Dan O'Sullivan illustrated this in their book 'Physical Computing'. Figure 8-1 shows their illustration of 'how the computer sees us'. One eye instead of two because we only look at two-dimensional screens. One finger only, as we input through sequential tapping, which might as well be the same single finger. Two small ears to hear stereo audio output from laptop speakers and the like.

Figure 8-1. 'How the computer sees us' from Physical Computing (Tom Igoe, Dan O'Sullivan)

If the illustration were updated to include new interface technologies as they become available, it would gradually become more human-like. In the same spirit, this chapter aims to inspire you to think about different input and output possibilities. Think about some of the ways in which we could provide input, or receive output and how they might be used:

Input through	*Used in…*
Touch, Press	*Physical controls, Touch screens*
Move & manipulate	*Tangible UIs*
Speech	*Speech recognition*
Whole body	*Gesture recognition, Proximity sensing*
Galvanic skin response	
Thoughts	*Brain-computer interfaces*
Heart rate	*Determine stress, anxiety, sleep,…*

Receive output through	
	LEDs, Screens
	Sound, Voice output
	Vibration, Force feedback, shape
	Scent messaging

	Temperature output

There are more interface types we could list. And not all the types we listed are readily applicable at the moment. Breaking down interaction into these components lets us combine them in different ways: A device might take voice input but output only through a screen. Or it might have physical controls, but only respond using a voice through a speaker.

In this chapter, we will discuss combinations of inputs and outputs that are particularly relevant for connected products.

Physical controls

Physical controls are everywhere. Push buttons that activate something. Switches that let us chose between two states. Sliders and rotary knobs that let us set a value on a range or select between multiple settings. Figure 8-2 shows a few different kinds.

Figure 8-2. A few physical controls from around a house (Photos from the author)

Physical controls aren't just input, sometimes they show a current setting. An electrical outlet switch controls the outlet, but also tells you if it's on or off. The tactile nature even allows you to feel the switch in the dark and use it without seeing it. In electronic devices, this can make things complicated. When users can change a setting from more than one place, controls might need to be motorized (like HIFI systems where the volume knob rotates even when the remote is used). Or a combination of physical control and display is required.

In addition to the type of control (button, switch, dial,…) designers also need to think about the tactile aspects. The haptic and tactile characteristics of a control influence the experience of a physical device. From the outside, they might be invisible.

To enable designers to work with the nuances that exist, the specifications for controls include operating force graphs. Figure 8-3 shows such graphs for different versions of what (from the outside) looks like the same button. Note the differences in both stroke (the distance the button travels when pressed) and operating force (the force at which the button 'activates').

Figure 8-3. Operating force specifications for different models of an identical looking button component.

From the outside, a user won't be able to spot that difference. But it will make a noticeable difference to the way the interaction feels. This has a practical aspect: making sure a button is harder to press so users don't triggered it by accident. There is also an emotional aspect: making sure pressing a button feels satisfying rather than flimsy.

To get an idea for how nuanced the tactile quality of a button can feel, just read a few reviews on the blog http://knobfeel.tumblr.com. It's not exactly a serious resource, but it gives an idea of how a certain button feels to somebody over another one. For example, one of the reviews reads: “I don’t quite understand how it simultaneously is light to move, but [has] a heavy feel. The rotation can only be described as creamy.”

Physical controls can become an identifying feature that makes a product stand out. Think of the click wheel control of the iPod. Another example is the Aether Cone (see figure 8-4): a connected speaker that uses physical controls as part of its minimalistic interface. The device combines voice input with physical controls. It learns about personal preferences and automatically curates a playlist.

Rotating the front speaker grille slightly skips the current song. Rotating it further changes the entire musical genre. Pressing in the center of the speaker activates the voice input feature that lets users request specific songs. This interface is intriguing and makes the product stand out from its competition.

Figure 8-4. Some of Aether Cone’s less obvious controls: Rotating the speaker grille and pushing in the centre of the speaker.

Using unconventional physical controls that are integrated with the device design can have drawbacks. Discovering the controls can be hard when they are non-obvious. Users need to learn where the controls are and how they work. Good usability can also be a challenge. Controls defined by the form factor, instead of interaction requirements, might be suboptimal for the interactions required. Keeping controls to the bare minimum is often part of what makes a product simple and elegant. But too few controls can negatively influence the usability and value. A single control might need to do different things at different times. This means users have to switch modes or accept less functionality. If you’re using unconventional controls on your device, consider helping users with discovery and learning. To do this, you can utilize marketing and advertising to educate people about how they can use your product. The early TV adverts for Apple’s iPhone took this approach. The short clips demonstrated how features worked by stepping viewers through the interaction. This has no doubt helped early customers know how to use the device before they even bought it. Once a customer has purchased the product, the out-of-box experience (the moment where the product is taken out of its packaging) can help, too. Printed instructions that users reveal as they unbox the device or removable labels affixed to the product can convey essential information about its controls.

Visual & screen interfaces

Light output

Almost every electronic device has at least one LED somewhere showing that it is switched on or indicating something. Connected devices often have more things to communicate. Hubs and bridges often have a number of LEDs to show a variety of connectivity statuses. When devices need to communicate even more, it can get complicated. Color-coding or blink patterns might be required to convey complex information through a simple LED. This can make understanding what is going on tedious. See the list in figure 8-5 that explains different light colors and blink patterns of the Toymail Mailman.

Figure 8-5. When using only one multi-color light and different blink patterns, debugging can become more complicated.

Light can also be the main interface of a device. The Ambient Orb (see figure 8-6) is a ball shaped device that glows in different colors. It can be configured to display data such as current energy prices or stock prices. Creator David Rose describes it as a "single-pixel browser [that] glows a color to reflect online information". Light here isn't a secondary signal, it's key for the primary purpose of the device: Conveying information in a glanceable, ambient way, requiring less attention and providing less distraction.

Figure 8-6. The Ambient Orb uses different colored light to display information, for example current energy prices.

The Ambient Umbrella (see figure 8-7) uses light in a similar way. Its handle glows when it is going to rain to remind the owner to take it out.

Figure 8-7. The Ambient Umbrella has a light integrated in its handle that lights up when rain is forecasted.

Visual input

Users can also give device input through light or other visual means. A device can use a camera to interpret a graphical pattern like a barcode or a QR code. Devices can also use light sensors to react to ambient lighting conditions or receive input from other devices through LED blink patterns.

Because these kinds of interactions and interfaces aren't often referred to as 'visual input', we'll discuss them in a separate section below called "Computer vision, barcodes & devices 'seeing'".

Screens and displays

Sometimes you need more than some lights. When a device needs to display complex information or users need to provide more sophisticated input, screens are more appropriate. As a UX designer you'll be familiar with screen-based user interfaces of smartphones or personal computers. But there is a range of (simpler) display types that require different design approaches UX designers are less likely to be familiar with.

Custom segment displays

Custom segment displays are what many cheap electronic devices use. Like the water filter system shown in figure 8-8. The underlying technology is often LCD (liquid crystal display). This means transparent segments become visible when an electric charge is applied.

Figure 8-8. A custom segment LCD display. The overall display needs to be large enough to include all segments that might be shown.

When creating a product with such a display, every possible state the display needs to be able to show needs to be considered in the design phase. Even segments that might only appear in certain situations need to be given real estate. If a heating controller can show temperature in both Fahrenheit and Celsius, segments for both '°C' and '°F' need to be part of the segmentation layout. They cannot be in the same place.

Segment displays are low cost and use little power. But they are unable to display dynamic alphanumeric information. To do that, you can use a character set display.

Character set displays

Many will have seen a simple character set display in an electronic calculator. Typically a 7-segment display was used (see figure 8-9).

Figure 8-9. A 7-segment display used in a calculator.

Character set displays work through systems of 7, 14 or sometimes 16 segments (see figure 8-10) that are switched on in combination to create numbers. 14 and 16-segment displays can also display basic alphabetic characters.

Figure 8-10. The segment layout of 7, 14 and 16-segment displays. (Illustration from the author)

Displays like this are cheap, but have a limited character set. A 7-segment system is only usable for displaying numbers. Other types such as 14- or 16-segment systems can display letters, but with limitations. See figure 8-11 for an example of the character set of a 14-segment display.

Figure 8-11. The character set of a 14-segment display.

By combining character set and custom segment displays, you can display both dynamic alphanumeric data and illustrations or graphical elements. Figure 8-12 shows such a combination used in a heating controller. A custom segment display that shows boiler and radiator illustrations is combined with a character set display for dynamic messages.

Figure 8-12. Note the 'Message display' area using a 14-segment display.

Segmented character set displays don't have to be ugly. The creators of the CST-1 watch designed a custom segmented font to display numeric characters (see figure 8-13).

Figure 8-13. The segmentation used by the CST-1 watch. Note the first digit misses unnecessary segments as it will only ever need to display "1" or "2" to display time.

Dynamic displays

Displays with the fewest limitations have a matrix of pixels each of which can be turned on or off individually. In principle, displays like this are the kind we're all used to from smartphones or computer monitors. A notable distinction is between monochrome and color displays.

Monochrome displays tend to be cheaper, but (as their name suggests) unable to display colors or shades. The Fitbit activity tracker uses such a monochrome OLED matrix display. It shows both alphanumeric information and simple graphics (see figure 8-14).

Figure 8-14. The Fitbit activity tracker uses a monochrome matrix display.

UX designers will be familiar with color pixel matrix displays from smartphones, laptops and tablet devices. Touch screens are essentially pixel matrix displays combined with a transparent touch sensitive layer.

Electronic ink displays

A more unusual display technology with unique advantages and disadvantages is electronic ink.

Made popular through eBook readers like Amazon Kindle, the technology is used in more and more devices like connected watches or smartphones. Electronic ink displays resemble the appearance of paper and are easy to read in direct sunlight. Unlike the other technologies described above, they only require electricity to change the state, not to hold it. This means the device uses less power.

This opens up new possibilities for designers, too. For example, an electronic ink display can show something when a device is off. To greet and guide users, a display could already show information when the user unpacks the product and hasn't even switched on.

There are also some disadvantages that constrain the design. Electronic ink displays have a very low refresh rate. They also suffer from an effect called 'ghosting', where shadows from a previous display state can still be seen. This rules out interactions like scrolling, the use of mouse pointers or animations in the UI.

Is a screen better than no screen?

You might think it's good to put a screen on a device where possible, but this isn't necessarily the case. Screens and physical controls increase the cost of a device. They mean extra components and add design and development effort.

When designing apps or websites a lot of work has already been done for designers. They can rely on platform features or design frameworks. If you add a screen to a connected device, you essentially need to develop an interaction paradigm from scratch. How do menus work? How do you navigate them? How do users learn the unfamiliar UI?

Moreover, once a screen is available, it becomes harder to say ‘no’ to adding all sorts of information and features since real estate seems endless. So it becomes harder to keep devices simple and focused.

That’s why designers need to take a step back early on in the design process to ask: What if this product didn’t have a screen? Could it still work? You might just discover an opportunity for a beautifully simple and easy-to-use connected product.

Audio & voice interfaces

Audio output

Audio can be a useful output method. Users don’t have to look at the device directly and don’t even have to be in near it. At the same time, anyone who has experienced the persistent beeping of a washing machine that has finished a cycle or a smoke alarm indicating its low battery knows how annoying audio output can be.

Audio is pervasive. Unlike lights, audio is almost never subtle. Sounds immediately trump visual indications. They are hard to ignore. Audio also has a bigger emotional character. This is obvious for speech: Tone and accent can be perceived as sympathetic and friendly, or the opposite. It’s also the case for other sounds: Pitch and quality immediately carry emotions. A sound can convey urgency like an ear-piercing high-pitch beep. Or it can convey a sense of satisfaction like a fulfilling swoosh sound when an email is sent.

Voice interfaces

Voice as output

Instead of sound effects, audio output can also be used for computer-generated speech. This is a powerful method for providing a wide range of information.

For example, the Nest Protect smoke alarm uses its speaker to play warning sounds. But it also gives more information using speech: A voice tells the user in which room smoke was detected.

And we’ve all used car navigation systems that use speech. The advantage is twofold: We can keep our eyes on the road, and something that can be hard to convey visually is much easier explained in speech. An illustration can become confusing when it doesn’t match the real layout of the roundabout ahead, or when the junction involves a complex road layout that is quite straightforward when explained verbally.

Voice as input

Speech recognition has become an ordinary feature of smartphones. When it works it can be a quick way to give complex commands. Setting an alarm using a touch screen, for example, requires a series of steps. A voice command is quicker and often easier.

Voice input can remove the need for many other controls. The Amazon Dash (see figure 8-15), a device that lets users add items to a shopping list, only has two buttons: One to activate voice input, one for a barcode scanner. Voice input replaces an entire keyboard in this case.

Figure 8-15. A person speaking into Amazon Dash

But there are also a few challenges with the technology.

Speech recognition is still unreliable. Anyone that has used it has encountered situations where commands were misinterpreted. Feedback mechanisms are required to confirm correct recognition. This can be done using a screen, but might defeat the purpose of not having to look at something. Confirmation can also be done through speech output. This quickly becomes a frustrating loop going back and forth.

Another challenge is to know what commands are possible in the first place. Users must recall (instead of recognize) what 'functions' are available, or how they need to phrase commands. Devices with screens often display a list of possible commands to help users with this problem.

Devices need to know when to listen for a command. One approach is to press a button, but this means the overall interaction isn't hands free. Another approach is activation phrases like 'Ok, Google' that indicate that a command follows. The downside of this approach is that users agonize over the idea that a device is constantly listening to everything that is being said. It also makes false positive detection a problem - when you say something that sounds similar to 'Ok, Google'.

A prank exploiting this is to play Xbox online multiplayer games using the gamer name "Xbox sign out". Other players will eventually say the name out loud, perhaps to vent their confusion about it in the game's audio channel. If those players have the voice control features enabled, they will soon disappear from the game server.

Tangible & tactile interaction

Tangible user interfaces

Tangible interfaces let users give input to a device or system through moving or placing physical objects or touching objects together.

In 1992 designer Durrell Bishop created one of the earliest explorations of this idea. Bishop envisioned an answering machine that made messages tangible through marbles that represent messages. (See figure 8-16)

Figure 8-16. Still from the concept video: The user places a marble into the message playback position.

Colored marbles represent recorded messages. The machine releases them into a groove for the user to pick up. To play a message, the user places a marble into a depression on the surface of the machine. Playback stops as soon as the user removes the marble from that spot. Users can either keep messages by holding on to the marbles, or delete them by putting the marbles back into the answering machine through a hole.

A more recent exploration of tangible interaction is the 'Spotify Box' designer Jordi Parra created as a student project. (See figure 8-17)

Figure 8-17. Jordi Parra's Spotify Box with a playlist token attached to the 'play' spot.

The box plays music streamed from Spotify. By placing a physical magnetic token on the box users can select a playlist. No menu or button combination is required. This gives an

analogue quality to the interactions with digital music. Users can label tokens or draw on them. They can also exchange them with other owners of the Spotify Box.

The marble answering machine and the Spotify Box are interesting concepts because they explore giving something intangible (voice messages or music) a physical representation. This changes the nature of interaction with a digital system, turning it into something more direct and immediate.

Tangible interfaces offer a way to simplify the control of a device. The direct and physical nature of manipulation can be easier to learn and understand than abstracted menu structures or conceptual mappings of controls.

A great example of this are Sifteo cubes (see figure 8-18). They are matchbox-sized cubes with a screen that sense when they are next to each other. Through tilting, placing or rearranging them, kids can play educational and entertainment games across the cubes.

Figure 8-18. Sifteo Cubes sense when they are next to each other to enable games played by manipulating and placing the cubes.

A simpler example is the Muji 'Multi Clock' alarm clock. It's a cubic device with labels for its four functions placed along each side of its display. Users select a function by orienting the clock with the corresponding label on the top. Figure 8-19 below shows the 'calendar' function activated.

Figure 8-19. The Muji Multi Clock with the calendar function activated.

A promising application of tangible user interfaces is in musical instruments. The Reactable (see figure 8-20) is such a musical instrument with a tangible user interface. Musicians manipulate and create sounds using physical objects placed and moved on the table's surface. The tangible nature means the musician can create and control many different sounds and keep the overview during a live performance.

Figure 8-20. A Reactable in use making music.

Experimenting with tangible user interfaces is thought provoking and inspiring. But interfaces involving many small physical tokens to manipulate or represent data have an obvious impracticality: If you lose parts the interface breaks.

Imagine not being able to listen to messages because your cat or a child has made off with the marbles. So you need to think carefully about the application of tangible interfaces.

Tactile output: Vibration, force feedback and shape shifting

We've all experienced vibration alerts of our mobile phones. It's a useful feature when it's noisy or when sounds would cause disturbance. But it requires body contact, which few devices have reliably.

The potential of tactile output is intriguing: Researchers at the University of Amsterdam have explored the use of vibration for cyclists. Their "vibrobelt", a GPS based navigation using vibration, is worn around the waist and gives the cyclist tactile feedback when they

need to take a turn. (http://www.newscientist.com/blogs/onepercent/2013/01/vibrating-navigator-cyclists.html) Apples watch has a similar feature. It can 'tap' users on the wrist during navigation, using different taps for 'go left' or 'go right'.

A much more sophisticated form of tactile output are force feedback mechanisms. These use motors to push back on the user in specific directions. Gaming joysticks use this to great effect to let players feel the g-forces involved in flying a plane for example.

A more serious device using force feedback is the Geomagic Touch (see figure 8-21). This is a device that makes virtual objects tangible.

Figure 8-21. The Geomagic Touch uses force feedback to make virtual objects tangible.

The central feature is a pen-like device mounted on multiple motorized axes. Users can give input in three dimensions using the pen, but the device can also 'push back' in all three dimensions, too. The applications are specific, such as 3D CAD modeling or surgery robots. An advantage is that users can physically feel virtual boundaries.

As with tangible interfaces, it's intriguing and inspiring to explore how tactile interfaces can change everyday interactions. The 'Proverbial Wallets' project from MIT explores this with financial data. The researchers created three prototype wallets, each using a different kind of tactile output for different kinds of data.

The first wallet is 'Bumblebee'. It uses vibration to notify the owner of every transaction that is happening with their bank account. The second is 'Mother Bear' (see figure 8-22). It uses its force feedback hinge mechanism to make it harder (or easier) to open the wallet depending on the current budget. The last is 'Peacock'. It's a wallet that swells and shrinks, reflecting the balance in the owner's account.

Figure 8-22. One of the three prototype wallets from the 'Proverbial Wallets' project.

Researcher Fabian Hemmert has explored the possibilities of tactile output in mobile phones. His prototypes include devices that shift their center of gravity and devices that shift their shape (see figure 8-23).

Figure 8-23. Fabian Hemmert's prototypes for shape- and centre of gravity shifting devices.

As with the Proverbial Wallets, it is intriguing to think about the ambient representation of data. Instead of displaying it in concrete terms, users feel it while interacting with the device. Hemmert gives the example of representing progress through a playlist and the current song at the same time using the center of gravity. Or giving non-visual directions in navigation applications.

Gestural interaction

Gestural interactions exist on touch screens (like 'swiping' or 'pinching'). Using computer vision, devices can also recognize mid-air gestures as user input.

Perhaps one of the most famous examples is from the film Minority Report. Chief John Anderton (played by Tom Cruise) controls a computer system using both his hands in mid-air, almost like conducting an orchestra. (See figure 8-24)

Figure 8-24. The film Minority Report made gestural control popular although it is an interface method with many flaws.

The film has had major influence on the development of gestural interfaces. But research has since shown that gestural interfaces might not be appropriate for real world applications. Holding out your arms in front of your body, gesturing or doing precise motions for extended periods of time causes aching muscles and stiffness. This condition has been dubbed 'Gorilla Arm'. Rumor has it that Tom Cruise had to take breaks from even just acting the scenes for the movie.

Yet, industries like gaming have adopted gestural interfaces with great success. The Microsoft Kinect lets gamers control Xbox games or navigate menus without any further device or controller. Users wave hands or move their body to control what they want to do.

Simpler forms of gesture interaction have also been integrated into more mundane products. The original Nest Protect smoke alarm had a featured called 'Wave to hush'. The feature was meant to solve the problem of alarms going off because of burnt toast or smoky frying. Users could stand underneath the device and wave their hands at it to silence it for a short period of time. (See Nest's illustration in figure 8-25)

Figure 8-25. The now disabled 'wave to hush' feature let users silence the smoke alarm by waving their hand at it.

But, the possibility of users unintentionally activating it (perhaps waving their hands during a real fire) could delay an alarm. In April 2014 Nest had halted sales and disabled the feature on already installed units to investigate potential problems with the feature. So, gesture interaction can be an elegant solution to interacting with a device that is hard to reach (like a smoke alarm). Yet, designers need to think carefully about the context and risk of false positives.

Context-sensitive interaction

Context-sensitive interaction is similar to gesture interaction, in that it revolves around the body. The difference is that context-sensitive interaction is user input in a less attentive manner. For example, instead of gesturing to make a command, simply being (or not being) in a certain room can provide input to a system, too. Context can also mean whether you're at home or standing at a bus stop, this can inform and tailor the user experience. Simpler still, being close by or further away from a product can be useful information to tailor an interface.

David Rose, founder of Ambient Devices, demonstrated a product that works in such a way at the LIFT'09 conference. Rose showed a connected weather forecasting device with a screen and a proximity sensor.

Using the proximity sensor, the device changes the amount of information it displays (see figure 8-26).

Figure 8-26. Example illustration of a weather forecasting device that displays more or less information depending on the proximity of a person in front of it. (Illustration from Alfred Lui)

When a user is standing across the room, the screen displays only essential information. The current outside temperature fills up the screen in large digits and the background gives clues to the weather (e.g. rain drops around the temperature).

When the user is closer to the device, the screen shows more information using smaller fonts. It now includes a forecast for the next 24 hours. Finally, when users tap the device, it shows even more information. The interface is sensitive to proximity, which acts as a proxy to the user context. The device uses this context in two ways. It enlarges text to make it easier to read from across the room. It also reduces the amount of information given to reduce the cognitive load on the user. Instead of having to 'study' the weather forecast, users see the most essential information at a glance.

The Aether Cone (figure 8-27) connected speaker is also context-sensitive. It's a mobile device that learns about the user's musical preference in different rooms of the house. It uses this to automatically adjust what it plays based on what room the user places it in.

Figure 8-27. The Aether Cone connected speaker remembers what music the owner prefers to listen to in different rooms.

Like tactile interfaces, context-sensitive interfaces demand less of the user's attention. This approach promises to make interacting with connected products less overbearing. In a way, less like operating a computer.

Designing interfaces like this requires a good understanding of human behavior. Designers can only ever assume the user's context through a proxy that a device can sense, like location or proximity. It's the designer that needs to do the interpretation. He or she needs the human insight that people listen to different kinds of music in different rooms of the house. Careful user experience design then needs to ensure that the device works well, whether this is true for its particular user or not.

Computer vision, barcodes & devices 'seeing'

In this section, we'll talk about interfaces that involve devices seeing or responding to physical artifacts around them. By this we mean interactions such as face, object or pattern recognition, as well as simpler ways a device can receive input from a physical artifact.

To clarify, computer vision isn't an interface type, it's an enabler. The gestural interactions we discussed above rely on computer vision. Simpler forms such as face recognition can enable useful features such as smartphones that can unlock using the owner's face. Face detection (instead of recognition) can enable smart cameras that automatically follow a person's face.

The power of these kinds of interactions is to simplify or remove user input. Compared to inputting a PIN code, it's faster and easier to just look at a device to unlock it. It may be less secure, but that is a different story.

This power of simplification is also handy when users need enter information from the physical world (like a book title) to start a digital interaction. The Amazon Fire Phone introduced a feature that does just that. In figure 8-28 you can see device recognizing a physical book in front of it. It then displays relevant information and actions.

Figure 8-28. Amazon's Fire Phone can recognise objects and present relevant shortcut functions.

In connected products, designers can use these kinds of interfaces to tailor the experience. For example by recognizing who is in front of a device. A challenge here is not to create a patronizing experience.

The smart drinks vending machines installed in some train stations in Tokyo are a real-world example that caused some debate due to their Orwellian characteristics. Instead of a glass through which available drinks can be seen, this machine displays its selection using a large screen on its front. The machine also uses a camera to determine the age and gender of whoever is standing in front of it. Based on this information the machine displays a targeted selection of drinks. For example, it is said to suggest a slightly sweeter beverage to a woman in her 20s.[1] The device also captures the data and the actual choice the customer made for marketing analytics. The sensitivity to features like this is different in different cultures. Allegedly the Japanese customers don't share the views and concerns some bloggers and commentators expressed.[2]

Marketing applications aside, there are also practical uses of such interfaces. A well-known example is QR codes. It's a useful way for users to input complex information with a minimal interface. Instead of manually entering Wifi credentials, for example, users could present them using a QR code for a device to decode.

Scanning a QR code requires a camera and the ability to interpret the image presented. This means the device to scan needs to be relatively complex. But the principle of encoding and transmitting information optically can be utilized simpler. The Electric Imp, a platform to connected device to Wifi, uses a light sensor (instead of a camera) to receive input. Users enter the Wifi credentials on a smartphone that then turns them into a flashing sequence performed by the screen.

Electric Imp calls this process 'BlinkUp'. The light sensor recognizes the sequence and the device decodes the information contained.

The 'Toymail' connected kids toys use this platform to connect. See figure 8-29 for the key steps in the process.

Figure 8-29. Wifi credentials are entered on a smartphone, the device is placed on the smartphone screen with its feet, the screen flashes on and off for a short period, an LED in one of the feet lights up green when the process succeeded.

[1] http://www.telegraph.co.uk/news/worldnews/asia/japan/8136743/Japanese-vending-machine-tells-you-what-you-should-drink.html

[2] http://globalitemagazine.com/2010/10/10/economy-japanese-embrace-human-feeling-machine/

Are QR codes good or bad?

QR codes have a bad reputation amongst designers. The apps required to scan them haven't reached widespread adoption. The outcome of scanning one often doesn't match the effort required to scan a code in the first place. An Internet joke mocking this reality is a Tumblr blog that has been setup to collect pictures of people scanning QR codes. The irony is, since its creation in early 2012 no pictures have been posted to it. (See figure 8-30).

Figure 8-30. http://picturesofpeoplescanningqrcodes.tumblr.com/

But, when the scanning logic isn't for the user to worry about, QR codes are a perfect means to input complex data quickly. Great examples are airplane boarding passes or event tickets stored on mobile phones. For connected products they can be a great method, too. A connected lock might use its door camera to validate a temporary key presented using a QR code.

Multi-modal interaction & interface combinations

When we talk to another person, we communicate using multiple channels at the same time. Speech, gestures, facial expression and body language work together. In a similar way, interaction with a connected device can combine different types of interfaces into one. This is called multi-modal interaction.

A great example of this is the 'Room-E' project Jared Ficklin did at frog design (see figure 8-31). By combining gesture and voice interaction, Ficklin demonstrates how users can provide input similar to the way they would talk to another person. In his demo, Ficklin points at a specific light in the room (gesture input) and says "Computer, turn off this light" (voice input). The computer combines both inputs into a single command. It is able to understand that Ficklin specifies which light he means to switch off using the gesture.

Figure 8-31. Jared Ficklin is demonstrating Room-E. A gesture (pointing at a lamp) is combined with voice input ("Computer, turn off this light") to control the room.

Although 'Room-E' is a prototype, it shows the potential of multi-modal interaction. By combining interface types, designers can create interactions that feel more like human-to-human interaction.

IoT specific challenges and opportunities

Deciding on the level of interactivity of a connected device.

Adding interactivity to a device is a cost question. Buttons, switches and the components required add to the bill of materials and increase development costs. It's also a usability question. Designers need to strike a balance between easy-to-use devices and ones with many features and functions. This is sometimes referred to as the "flexibility usability tradeoff". (See figure 8-32)

Figure 8-32. The flexibility usability tradeoff as illustrated in the book "Universal Principles of Design"

However, simple devices with little features have other disadvantages. They often rely on other devices for full-feature interaction. This makes them dependent on things outside of the designer's control, such as a working Internet connection or a compatible browser. Imagine you couldn't change the temperature in your house because of problems with your Internet service provider.

An example of this approach is the Tado thermostat (figure 8-33). Users control it using a website and smartphone app. But the product offers a minimal degree of control on the device, too. A single physical button lets users switch the heating on or off as a fallback option.

Figure 8-33. The Tado thermostat relies on smartphone apps and a website for control - but the box offers a minimal degree of control: A button on the side lets users switch the heating on.

Only having few controls on a device is especially advantageous if you know your product will change and develop. In this case a supporting a physical interface quickly becomes a burden.

There are ways to balance the benefits and drawbacks of physical controls. An example is BERG's connected washing machine prototype 'Cloudwash' (see figure 8-34). The physical device has some controls that only cover selected functions. Full control is delivered through a smartphone app. But, users can perform the everyday tasks without needing the smartphone app. Also, the device uses a combination of physical controls with displays. This keeps the physical interface flexible within reason.

Figure 8-34. BERG's Cloudwash prototype uses displays to dynamically label physical controls.

From BERG's website:

> *"The legends for these washes are in e-ink so they can be updated but still feel like part of the legends one would expect to find on a washing machine. They can be far more descriptive than the descriptions on a traditional dial, since there's no need for generalisations like 'cottons.' We like human-readable descriptions like 'Joe's sports kit' or 'really really quick' or 'use this for baby stuff'."*

Glanceable and ambient interfaces

We've already touched on how designers can create interactions that require less attention. Considering the ever-increasing amount of information around us, this will become ever more important. We'll soon all crave a less distracting environment.

This kind of interaction is already around us. Consider how a wall clock 'disappears' into our surrounding. We completely ignore it for the most part. The moment we're interested in the time, we can get it at a glance. Even from across the room. We might only be able

to see roughly the angle of the two main hands - but it's enough to give us an idea of what time it is. If we want more detail, we can walk closer to see exactly what time it is.

The ambient devices mentioned earlier want to achieve just that. These kinds of devices share certain characteristics. David Rose, founder of Ambient Devices, has created a list of these characteristics that designers can use.

Pre-attentive	*Glance-able, no cognitive load required.*
Calm	*Non-intrusive, seamless with environment. Evergreen. Friendly.*
Universal	*No language, characters or numbers.*
Open	*Able to represent multiple types of data, coded, private.*

We highly recommend exploring how you could bring some of these characteristics into the products you're working on. Even if it's just a thought experiment.

Working with limited input and output capabilities

A particular challenge in connected devices is to work around limited input and output capabilities. Many connected devices only require complex input rarely and a simple interface is sufficient most of the time. So how can designers handle these rare, but complex interactions without having to bloat the interface with capabilities that aren't needed most of the time?

The Withings Smart Body Analyser provides a great example. The smart body scale supports multiple users each with their own accounts. It automatically identifies which user is standing on it by comparing the reading with its historic data. If the new reading is within a realistic extrapolation of only one account, it can only be that person.

This doesn't always work. When a reading is within range of more than one account, the device cannot reliably decide which user it is. In this case, the device needs user input. But instead of having specific controls for this situation, the designers found a solution that keeps the interface simple. They work with the capabilities available. In this case, the display shows a selection between the likely accounts. Users then lift either their left or the right foot to make a selection. (See figure 8-38)

Figure 8-35. The Smart Body Analyser displays a choice between two users when it is unsure who the current reading belongs to. By standing only on the right or left side of the scale, the user can make a selection.

Another strategy is to support devices with more capabilities. Navigating and controlling Apple TV with the minimalistic remote is sufficient for most of the time. When it comes to entering search terms or login credentials, using the remote becomes tedious. By supporting an iPhone as an alternative input device, users can switch to the more capable device when required. (See figure 8-39)

Figure 8-36. Apple TV Remote, Remote App and TV interface.

The Amazon Fire TV remote (see figure 8-40) uses speech recognition to overcome this same hurdle. Its remote has a dedicated button for voice search and users can speak a

search term into the remote rather than having to type. That way, the interface stays simple even though complex input is possible.

Figure 8-37. Amazon Fire TV remote

Mobile and web UIs

In addition to on-device interfaces, many connected products involve web interfaces or smartphone apps. General advice on how to design for these touch points is outside of the scope of this book. But in chapter 10: Key Interactions, we discuss how smartphone and web interfaces can help with specific challenges like setup or device management in the interaction with connected products.

Universal Design & Accessibility

It is easy to forget that many products we work on as designers are inaccessible or much harder to use for people with disabilities like visual impairment. Sometimes, products that used to be accessible even become less accessible with the integration of new technologies. Like washing machines that move from physical controls to touch screens. Connected products offer an opportunity to create more accessible and universal products.

In this section we'll take a brief look at accessible and universal design in relation to connected products.

Accessibility

There are many aspects of a physical device that designers need to consider to make it appropriate for users with a certain disability. Checklists published by organizations such as the Royal National Institute of Blind People (RNIB) help designers with this task.

Below is an excerpt from the 'handling' section of RNIB's checklist "35 Questions to ask yourself when designing products for people with sight loss, disabilities and older people".

Checklist excerpt: Handling		
6. Is the product easy to orientate? Could a visually impaired and/or older person easily locate the front, back, top, and bottom of the product?	*Yes*	*No*
7. Does the product require little physical strength to use? Consider for example, an older person with arthritis – would they be able to lift, open, turn, grip or rotate the product to use it effectively?	*Yes*	*No*
8. Does the product have smooth edges and surfaces (not sharp or rough) and is without finger traps? Consider that an older and/or visually impaired person may have limited vision, reduced reaction time, motor control and dexterity.	*Yes*	*No*

The full checklist is a great tool if you want to make your product more accessible. It is available here: *NEED LINK*

Another simple but powerful method that helps designers make products accessible is to simulate disabilities. You can close your eyes and see if you can still use your prototype.

You can even use props to help you. Wear earplugs and earmuffs to simulate deafness. Or wear gardening gloves to simulate how somebody with arthritis might experience your product. (See figure 8-41)

Figure 8-38. At a Hackathon organised by the non-profit 'Enabled by design', participants used gloves and modified glasses to empathise with low dexterity or vision conditions.

A third strategy to create accessible products is to utilize existing accessible platforms. Android, iOS and the web all have great accessibility features built-in. If you don't have the resources to make your device accessible, a less resource intensive approach is to ensure you support these built-in features. You could make sure an app or website is available that lets users control every aspect of the product. Users can then use braille displays or built-in accessibility features like Apple's VoiceOver to control your product.

Universal design - we're all disabled sometimes

Universal design is a term coined by architect Ronald L. Mace. It refers to a design approach that aims to make products and environments usable for as many people as possible, regardless of their age or ability. Rather than designing for certain disabilities, the approach aims to create a design solution that works for everybody, whether disabled or not. You could say the approach assumes that everybody is disabled sometimes. Listening to music with headphones means you're temporarily 'deaf' to other audio alerts. Carrying your grocery shopping in one hand means you temporarily only have one hand available for other interactions. Universal design aims to make products usable in the broadest range of situations, whether the user is permanently disabled or not.

When it comes to technology, this can mean including multiple modes of output and input. For example, including light output in a smoke alarm in addition to audio makes the product accessible to deaf users without alienating non-deaf users. Including audio cues and tactile signposting in a device like a washing machine make it accessible to blind users without alienating the sighted.

Openness and compatibility with other systems also enable users create such systems themselves. For example, the Nest smoke alarm and the Lifx connected light bulbs are compatible. This means users can setup light bulbs to flash red in case of an alert from the smoke alarm.

As part of your design process, you should spend some time examining your concept through the lens of universal design. The 7 principles of universal design can help with that.

Principle 1: Equitable Use
The design is useful and marketable to people with diverse abilities.

Principle 2: Flexibility in Use
The design accommodates a wide range of individual preferences and abilities.

Principle 3: Simple and Intuitive Use
Use of the design is easy to understand, regardless of the user's experience, knowledge, language skills, or current concentration level.

Principle 4: Perceptible Information
The design communicates necessary information effectively to the user, regardless of ambient conditions or the user's sensory abilities.

Principle 5: Tolerance for Error
The design minimizes hazards and the adverse consequences of accidental or unintended actions.

Principle 6: Low Physical Effort
The design can be used efficiently and comfortably and with a minimum of fatigue.

Principle 7: Size and Space for Approach and Use
Appropriate size and space is provided for approach, reach, manipulation, and use regardless of user's body size, posture, or mobility.

The 7 principles of universal design were developed in 1997 by a working group of architects, product designers, engineers and environmental design researchers.

Chapter summary

There are many different interaction channels through which users can control a connected product. They all have different benefits and drawbacks that designers need to be aware of to create the most appropriate interface.

New kinds of connected products might require hitherto unfamiliar combinations of inputs and outputs to create intuitive and user-friendly experiences. To identify these, designers shouldn't settle for familiar or established interfaces too quickly. Instead, they should experiment with and explore new possibilities of interacting with a device or system.

Deciding on the input and output capabilities of an embedded device almost always involves compromising on either the device's flexibility or its usability.

As the number of devices people own increases, creating interfaces that require less attention, are less intrusive and require less cognitive load will become more important. But only some interface types are suitable to create such glanceable or ambient interfaces.

When input and output capabilities are limited, designers need to develop strategies to enable edge-case interactions that require complex interaction. This can be about finding creative ways of using these limited capabilities, or about supporting other devices.

Designers need to make sure devices are accessible or universal. Supporting multiple ways to input and output, relying on accessible software platforms and creating open systems that allow users to connect different devices together all help with this.

In summary, the benefits and drawbacks of the interface types discussed in this chapter are:

Physical controls

are great…

… for direct and fast control. Giving users access to something without having to find an app or navigate a menu. Think volume, mute or camera buttons on smartphones to provide quick access anytime.

… when fine adjustments are required. Tactile qualities like friction and momentum can be leveraged. Controlling volume on a heavy, slow-to-rotating knob allows for finer adjustment than a slider on a touch screen.

… **to make a product more accessible for users with impaired vision.** Blind users memorize button positions or affix tactile labels to guide them. Replacing all controls with a touch screen makes the product far less accessible to this group. (More on accessibility at the end of this chapter)

are less appropriate…

… on products that change and develop through software updates. For obvious reasons, it is hard to change controls around after the product is manufactured. Abstract labels (such as colors or shapes) or combinations with screens can keep physical controls dynamic.

… **when functions or settings can be controlled from multiple different places**. Controls might need separate displays to show state. They might even have to be motorized to move autonomously.

Lights

are great…

… for glance-able and non-intrusive information output. Whether it is about seeing something across a room, or integrating lights in a way where they don't distract - lights are great to give simple feedback without asking a lot of attention.

are less appropriate…

… for conveying complex information. System status or error messaging can be complex with connected devices. With only few lights available, UX quickly becomes painful and complex (like interpreting blink patterns with the help of a manual).

Screens

are great…

… for making physical objects dynamic. For example by combining screens with physical controls to get dynamic labels.

… for keeping products flexible. Because the entire UI can be updated and changed through software, screens more easily enable updates and iterations through software. This depends on the screen technology used, though, as discussed above under 'pre defined displays'.

are less appropriate…

… for keeping the user experience simple. Once a product has a screen, the ability to display more information at no extra cost can lure design teams away from simplicity.

Audio output

is great…

> **… for urgent and time-critical alerts.** Because audio is pervasive, it is great to quickly grab a user's attention.
>
> **… to give a product emotional qualities.** The quality of sounds can be a great way to convey emotions. An audible alert can be designed to sound 'efficient' or 'playful', depending on what the designer wants to achieve.

is less appropriate…

> **… for environments where it becomes annoying.** The biggest challenge with audio is to not make it annoying. This requires a good understanding of the environment the product will be in and careful application of audio.

Voice interfaces

are great…

> **… when other channels are occupied.** Think car navigation. Drivers can keep eyes on the road and still get information.
>
> **… to give complex information on an otherwise simple product.** Think the Nest smoke alarm that uses speech to say where in the house smoke was detected.
>
> … **for products with strong connectivity that are used in low noise environments**. Speech recognition is often done server-side. A fluid experience requires good connectivity. Noise or other people talking make recognition less reliable.
>
> **… for products where complex data input is required**. Tasks that take a number of steps on a touch screen UI, like setting an alarm clock, are much simpler and faster using speech recognition.
>
> **… when a minimal physical interface is desired.** Although a device might be capable of complex input, the impact on industrial design and the physical appearance can be minimal.
>
> **… for contexts where hands free interaction is desired.** For example in the kitchen, to keep working while interacting with a device. Also to keep the device sterile because it isn't touched.

are less appropriate…

> **… when the nature of commands and terms used is hard to pronounce or recognize.** A speech controlled music player is likely to struggle with unusual artist and song names, not necessarily due to recognition, even just due to the fact that correct pronunciation might not be clear to users.
>
> **… for products aimed at many different global markets.** Not all languages might be reliably supported. A fast global rollout might be impossible. At the time of writing, even Apple's Siri is not globally supported.

Tangible user interfaces

are great…

… **for digitally enabled experiences that don't feel like interacting with a computer.** Like museum exhibits that let visitors explore and experiment, or musical instruments like the Reactable that lets users interact with sounds and music.

… for educational products. Learning through manipulating objects is widely accepted as an important aspect of education. Tangible user interfaces can keep learning experiences tactile while enhancing them with dynamic digital content.

are less appropriate…

… when keeping all parts together is critical. If lost parts mean lost data or lost access, the drawbacks of tangible user interfaces might outweigh their benefits.

… when there is no time to learn. If the application doesn't allow for a learning phase to let users understand how to use the interface, other interface types might be more appropriate.

Tactile output

is great…

… for creating interfaces that demand less attention. Tactile output can reduce cognitive load. Instead of interpreting visual output, it can be faster (but less precise) to *feel* the information.

is less appropriate…

… (currently) for affordable, reliable systems. Mechanical systems involve friction, wear and tear, and they have a limited lifespan.

Gestural input

is great…

… **for video games.** Gestural control makes games more immersive and physically challenging.

… for short interactions. The 'gorilla arm' becomes a problem with lengthy use. If gestural interaction is kept short this is less of an issue.

… when it is obvious what gestures are possible. Users need to know or recall what gestures are possible, or have a guide.

is less appropriate…

… when precision and lengthy interaction is required. Giving precise input is hard with gestural interfaces. Lengthy interaction leads to fatigue and muscle pains.

… when false positive recognition can have serious consequences. Recognition is still unreliable and false positives can occur, gestural input is not appropriate if this can have serious consequences.

… when there is no time to learn. If the application doesn't allow for a learning phase to let users understand how to use the interface, other interface types might be more appropriate.

Contextual interaction

is great…

> **… to manage levels of complexity with little interaction.** Contextual interaction can be a great to move between levels or detail without requiring methodical user input.
>
> **… when there is an intuitive or obvious relationship between contexts and device features.** For example: Approaching a device and revealing information are easy to connect conceptually.

is less appropriate…

> **… when it might be perceived as patronizing.** Limiting options or taking action on behalf of the user might be undesired and perceived as negative and overbearing.

Computer vision & barcodes

are great…

> **… for replacing cumbersome input.** Computer vision & barcodes can replace manual input like typing in WIFI credentials, URLs or book titles.

are less appropriate…

> **… when the interaction is more complex than its alternatives.** If additional software (like a QR code reader) is required, the alternative interaction may be seen as simpler by users.

O'REILLY®

UX Strategy

HOW TO DEVISE INNOVATIVE DIGITAL PRODUCTS THAT PEOPLE WANT

Jaime Levy

2

The Four Tenets of UX Strategy

After this very dense chapter, I'm going to teach you some cool lightweight tools and techniques for devising digital products. But first I need you to understand the theory behind my UX Strategy framework so that you will have context to understand how it relates to those tools and techniques. Don't worry if some of the business terminology sounds bewildering. Just understand that a framework is simply a structure that keeps things contained, like a fence around a dog run. And a tenet is simply a principle, or a dogma (pun intended)!

How I Discovered the UX Discovery Phase AKA UX Strategy

When attorneys in the US are preparing for trial, each side is allowed to ask the other side for their evidence. That process is called "Discovery". It's designed to avoid "trial by ambush," where one side doesn't learn of the other side's evidence or witnesses until the trial, when there's no time to obtain answering evidence. Obviously, when trying to win a court case or releasing a digital product, you want to do everything you can to avoid surprises. It is why one *must* be strategic.

In 2007, I was exposed to my first incarnation of UX strategy. At the time, I was the UX lead at Schematic (now Possible) for the website redesign of Oprah.com. I flew to Chicago with the other team leads to meet with Oprah's stakeholders to kick-off our Discovery Phase. No, Oprah wasn't there. But the experience really opened my eyes to the world around me. I was intrigued about how our UX director, Mark Sloan, got everyone in the room on the same page by using consensus-building techniques such as content maps, vision statements, and feature prioritization exercises. Although I had already been designing for interactive for 15 years, it was my first time focusing my energy purely on strategy.

Within weeks, our team submitted a discovery brief of the future site to the stakeholders. It contained personas, user scenarios, concept map analyses, and a recommended feature list. Upon approval, we were off and running on the implementation phase. The UX phase of the Oprah project took over six months of emotionally fueled hand-offs - there were hundreds of pages of wireframes and functional specifications traded between stakeholders, designers and developers. A full year later, the redesigned site actually launched! But I never looked at it. By then, I had moved on to

another large agency with big projects and I could focus most of my energy on the discovery phase - the UX Strategy phase - of projects.

Today, I run my own practice that specializes in user experience strategy. But I want to make it clear that my methodology is my version of user experience strategy and may be different from other UX strategists. Why is that? Well, the truth is that UX strategists in general are rare birds. There are currently only a few dozen of us in the wild, or at least that I've met in person. This might be because it's inefficient for a client to hire two of us simultaneously on a project. Or maybe it's because the discipline is just so new that we've yet to find each other. I actually went to the first UX Strategy conference in Atlanta last year. There were experts with job titles other than UX Strategist who talked about what they thought UX strategy was. It reminded me of the early days of the web or when UX design first hit the scene and how no one could agree on what the right definition was. And that's OK. When a new discipline arises, there are always discrepancies in the frameworks and methodologies. It's up to you to discern the value behind each one.

Because this is my book, I'm going to teach you what I've learned, think, teach, and sell to clients. To me: UX strategy is an iterative process and a cross-functional initiative in which stakeholders, designers, product developers, and the marketing department work together to connect the dots between the customer's experience and the business' goals. It requires intense collaboration between team members no matter the work environment—startup, agency or enterprise—in order to prevent team members from working myopically. Because when everyone shares a product vision, things get done more cost-effectively and efficiently.

More importantly, this process should be used to build a disruptive product. I don't want to build another version of an average product that already exists. I want a UX strategy that will help build an innovative product with the potential to be sustained by an innovative business model. What's the point in spending time and energy crafting a product strategy that isn't unique? Or isn't at least a better alternative to current solutions in the online marketplace? A stellar UX strategy is a means to achieve disruption in the marketplace. And to keep me from forgetting this, I have the sticker below on my laptop lid.

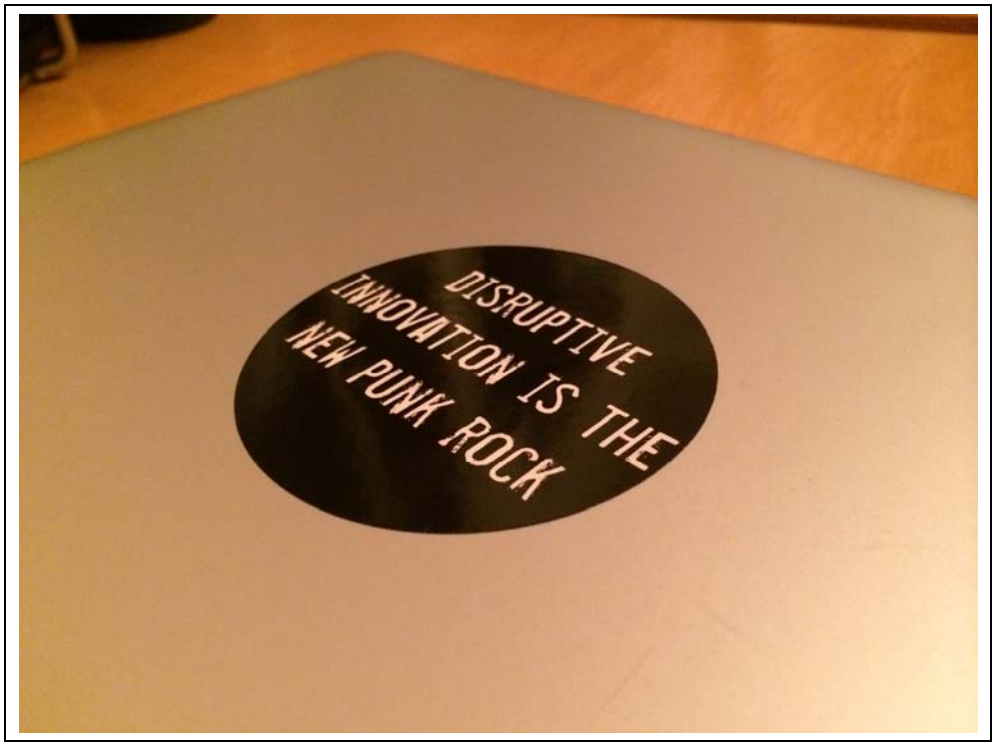


Figure 2-1. Photo of a sticker on my laptop that says, "DISRUPTIVE INNOVATION IS THE NEW PUNK ROCK

The Four Tenets of User Experience Strategy

Here's my UX strategy framework and equation:

Figure 2-2. Diagram of a square that says "UX Strategy" with four circles in it that say "Business Strategy", "Value Innovation", "Validated User Research" and "Killer UX Design"

UX Strategy = Business Strategy + Value Innovation + Validated User Research + Killer UX Design

All of these tenets are too complex to understand in their entirety, so I'm going to break down the most important components that I need you to understand for this book. If you already have an MBA or have read a stack of business books, then you should be familiar with the basics. And if you haven't, this is going to be a quick primer to get you up to speed on both traditional and contemporary business principles.

TENET 1 - BUSINESS STRATEGY

Business strategy is the top-line vision of the company. It is why the company exists. It ensures the long-term growth and sustainability of the organization. It is the basis for the core competencies and offerings, which are the products. In this book I will use the term "products" to describe goods and services.

The business strategy is what gives the product makers their wherewithal to grow in the marketplace while beating the competition. The strategy identifies the company's guiding principles of how it will position itself and still achieve its objectives. For this to happen, the business strategy MUST continually identify and leverage a competitive advantage. A competitive advantage is essential to the company's long-term existence.

In our rehab example, the stakeholder's guiding principle was to help people get better care. Their original business strategy was to create a better system that matched empty beds in legitimate treatment centers directly with consumers that needed them. When their website failed to make this happen, they needed to step back and look at the entire industry. Going head-to-head in online advertising campaigns against unscrupulous direct competitors with deep pockets was a no-win situation.

"In war, let your great object be victory, not lengthy campaigns."

—Sun Tzu, Art of War[1]

In his classic book *Competitive Advantage*[2], Michael E. Porter lays out the two most common ways to achieve a competitive advantage: cost leadership and differentiation.

The advantage behind cost leadership comes from offering the lowest price for products in a particular industry. Whether it is the cheapest car, television or hamburger, this was the traditional way that companies achieved dominance in the marketplace. After all, allowing the private sector to compete without government regulation is what free market economy is all about! I mean, look at the rampant success of stores like Wal-Mart and Target. They can offer consumers the best prices and widest selection of merchandise. But what happens when prices hit rock bottom? Then the battle needs to be about what makes the product better.

This brings us to Porter's second type of competitive advantage: differentiation. Since we are product inventors planning to build disruptive technologies, this is where our actual power lies. With differentiation, the advantage is based on a new or unique product or a unique aspect of the product that customers will pay a premium for because of its perceived value. That perceived value is what transforms a simple little coffee shop and cup of joe into the crazy success story of Seattle-based Starbucks. There's a reason why people pay $5 for a cafe latte—it's the experience that's also wrapped into the product. It starts the moment a customer steps into the store and ends when that person tosses their cup and sleeve into the trash. As consumers, we choose one product over another based on the things **we** personally value, ranging from the product's usefulness to how much pleasure we derive from it.

Today, differentiation is the product game changer. It's completely revolutionized the way we communicate with the world. Consider when micro-blogging and Twitter hit the scene. It was derided for the 140-character limit, which required users to choose their words carefully. But the limit has made it a valuable perk, especially with respect to instant updates. The royal British wedding announcement between Will and Kate and even news of Osama bin Laden's death broke on Twitter first. Even now, protesters from the Arab Spring, to Syria, to Venezuela continue to use the platform to

---

[1] Sun Tzu, *Art of War* first published by Lionel Giles in 1910

[2] Michael Porter, *Competitive Advantage*, New York: The Free Press, 1985

rally supporters and keep the world informed with up-to-the-second Tweets. Last spring (March 2014), Prime Minister Recep Tayyip Erdo'an tried to blackout the platform in Turkey. Within seconds of the shutdown, people began tweeting about it[3].

Another tool that uses differentiation to its advantage is the map app Waze. It combines social traffic with GPS navigation. By merely driving around with Waze open, users passively contribute traffic and other road data to the network. Users can also take a more active role by sharing road reports on accidents, police traps, or any other hazards along the way, helping to give other users in the area a heads-up about what's on the road. In June of 2013, Waze (an Israeli startup) was acquired by Google for $1.1B. Now, Waze still offers its user experience to its users, but its data is also channeled into Google Maps.[4] Clearly, Google recognized the competitive advantage of mass collaboration and chose to adopt Waze rather than compete against it.

While these amazing products kick ass with their competitive advantages, they also achieve success through a sustainable business model. Traditionally, the purpose of a competitive advantage was to make a product that was self-sufficient through a revenue stream. A revenue stream is how the company gets paid. And when a customer pays more for the product than what it costs to make, value is created for the stakeholders.

The process of business model construction is foundational to the business strategy, and there is not one way to do it. If you thumb through the seminal book *Business Model Generation*[5] by Alexander Oswerwalder and Yves Pigneur, you'll see there are hundreds of ways to skin a business model.

Figure 2-3. Business model canvas goes here showing the nine essential building blocks

But the importance of business models is often lost on young tech entrepreneurs. They grew up in a world where products like Facebook and Twitter started without obvious revenue streams and still managed to conquer the world without becoming solvent. However, if you were working on the web when the Dotcom bubble burst in the 1990s, you have first hand experience of all of the risks involved in creating products without proven revenue streams. When the investment money runs out, and there isn't any more coming in, it *is* bleak.

The mega-successful digital products that define our everyday lives didn't just stumble onto business models, they innovated new ones. Facebook didn't kick MySpace or Friendster's ass because it was cheaper. It did it because it offered a differentiated user experience that was more valuable because of its utility. It became a ubiquitous part of world culture because it was the de-facto communication tool that appealed to many types of people. It's not that a few people adopted Facebook. Everyone adopted it, and that's why it remains the undisputed social network of our time. That's how Facebook built its business model. It's able to use its mass adoption to prove its competitive advantage to other companies and sell that advantage through advertising and data mining.

---

[3] "Twitter Blocked in Turkey as Prime Minister Pledges to 'Eradicate' It" (http://tinyurl.com/pfpy6g4) Mashable. March 20, 2014

[4] "New features ahead: Google Maps and Waze apps better than ever" (http://tinyurl.com/lx9sq8c) Google Maps's Blog. August 20, 2013

[5] Alexander Oswerwalder and Yves Pigneur, *Business Model Generation*, Wiley 2010

This is why it's so important to understand how mass adoption of new technologies and business models interact. Also, as the product scales and the market evolves, the business strategy must be nimble. What may have been the business model or competitive advantage in the early life cycle of the product may not be the same in later phases. In the startup world, the business strategy generally revolves around getting just enough product/market fit to raise financing. For a mature company, the business strategy is more focused on the company's core value proposition while keeping the company's infrastructure and processes in place. That way it can continue to scale and remain competitive in an ever-changing marketplace.

So why is understanding business strategy so important to the UX Strategy framework? First off, strategy is a way of thinking. It's not formulating and executing a perfect plan, but rather being able to research what's out there, analyze the opportunities, run structured experiments, fail, learn, and iterate until you devise something of value that people truly want. While devising a solid strategy, we need to take risks and accept failure. Later, we'll talk more about the difference between "smart failure" and "dumb failure". We'll learn how to fail smartly by doing small-structured experiments to validate that our strategy is moving us in the right direction.

TENET 2 - VALUE INNOVATION

Value! Value! Value!

The word seems to be used everywhere. It's found in almost all traditional and contemporary business books since the 1970s. In *Management: Tasks, Responsibilities, Practices*[6] (1973), Peter Drucker talks about how customer values shift over time. He gives an example of how a teenage girl will buy a shoe for its fashion, but when she becomes a working mother, she will probably buy a shoe for its comfort and price. In 1984, Michael Lanning first coined the term "value proposition" to explain how a firm proposes to deliver a valuable customer experience. That same year, Michael Porter defined the term "value chain" as the chain of activities that a firm in a specific industry performs in order to deliver a valuable product.

All these perspectives on value are important, but let's fast-forward to 2004 when R.S. Kaplan[7] discussed how intangible assets like computer software were the ultimate source of "value creation." He says, "Strategy is based on a differentiated customer value proposition. Satisfying customers is the source of sustainable value creation."

Woah! There are a lot of things in that quote that align with what we just learned about business strategy—differentiation and satisfied customers. But there's one thing that we didn't discuss—the fact that we are designing digital products: software, apps and other things that users find on the Internet and use everyday. Often the users of these digital products don't have to pay for the privilege of using them. If a business model is supposed to help a company achieve sustainability, how can you do that when the online marketplace is overrun with free products? Well, we learned how many companies like Waze found a sustainable business model—sharing their crowd-sourced data made them lucrative to other companies like Google. But in order to get the data, they had to provide

[6] Peter Drucker, *Management: Tasks, Responsibilities, Practices*. HarperBusiness, 1973

[7] Robert Kaplan, *Strategy Maps*. Harvard Business School Press, 2004

value to their customer base for mass adoption, and that value was based entirely on innovation.

"Innovative" means doing something that is new, original, and important enough to shake up a market. As W. Chan Kim and Renée Mauborgne describe in *Blue Ocean Strategy*[8], value innovation is "the simultaneous pursuit of differentiation and low cost, creating a leap in value for both buyers and the company." This is accomplished by looking for ways that a company can reduce, raise, lower and eliminate factors that determine the cost and quality of a product.

When we transpose this theory to the world of digital products, the value proposition manifests itself as unique feature set. Features are product characteristics that deliver benefits to the user. In most cases, less features equals more value. Value can be created by consolidating features from relevant existing solutions (i.e. Meetup and Evite) and solving a problem for users in a more intuitive way (i.e. EventBrite). Value can be created by transcending the value propositions from existing platforms (i.e. Google Maps + Crowdsourcing = Waze). Sometimes it's consolidated from formerly disparate user experiences into one single solution (1-stop shop for a user task), i.e. sharing a video you made with your phone on YouTube, into one elegant simple solution (i.e. Vine and Instagram). We will deconstruct these complex techniques in Chapter 7 Storyboarding Value Innovation for Digital Products.

But for now, let's discuss the most important reason that we want to be unique and disruptive with both our products and our business models. There are bigger opportunities in unknown market spaces. We like to call these unknown market spaces "blue oceans." This term comes from the book *Blue Ocean Strategy* that I mentioned two paragraphs ago. The authors discuss their studies of 150 strategic moves spanning more than a hundred years and thirty industries. They explain how the companies behind the Ford Model T, Cirque du Soleil and the iPod chose unconventional strategies rather than fighting head-to-head with direct competitors in an existing industry. The sea of other competitors with similar products is known as a "red ocean." Red oceans are full of sharks that compete for the same customer by offering lower prices and eventually turning a product into a commodity.

Figure 2-4. Photograph of an unappetizing hamburger

In the corporate world, the impulse to compete by destroying your rivals is rooted in military strategy. In war, the fight typically plays out over a specific terrain. The battle gets bloody when one side wants what the other side has - whether it be oil, land, shelf space or eyeballs. In a blue ocean, the opportunity is not constrained by traditional boundaries. It's about breaking a few rules that aren't quite rules *yet* or even inventing your own game that creates an uncontested new marketplace and space for users to roam.

A perfect example of a company with a digital product that did this is Airbnb. Airbnb is a "community marketplace" for people to list, discover and book sublets of practically anything from a tree house in Los Angeles to a castle in France. What's amazing about this is that their value proposition has completely disrupted the travel industry. It's affecting the profit margins of hotels so much that Airbnb was banned in

[8] W. Chan Kim and Renée Mauborgne, *Blue Ocean Strategy*. Harvard Business School Press, 2005

New York City[9]. Its value proposition is so addictive that once customers try it, it's hard to go back to the old way of booking a place to stay or subletting a property.

Figure 2-5. Public domain photo of a legal doc or image from the AP showing Airbnb's legal issues in NYC

For instance, I just came back from a weekend in San Francisco with my family. Instead of booking a hotel that would have cost us upwards of $1200 total (two rooms for two nights at a 3.5 star hotel), we used Airbnb and spent half of that. But for us, it wasn't just about saving money. It was about being in a gorgeous and spacious 2-bedroom home closer to the locals and their foodie restaurants. The 3% commission fee we paid to Airbnb was negligible. Interestingly, the corporate lawyer who owned this SF home was off in Paris with her family. She was also staying at an Airbnb, which could have been paid for using some of the revenue ($550+) from her transaction with us. Everybody won! Except, of course, the hotels that lost our business.

Airbnb achieves this value innovation by coupling a killer user experience design with a tantalizing value proposition. A value proposition is the reason why customers accept one solution over another. Sometimes the solution solves a problem we didn't even know we had. Sometimes it creates an undeniable desire. A value proposition consists of a bundle of products/and or services ("features") that cater to the requirements of a specific customer segment. Airbnb offers an unbeatable value proposition to both sides of its two-sided market—the people who list their homes and those who book places to stay.

Airbnb chose not to focus on beating the existing competition (other subletting sites and hotels) at their game. Instead they made the competition irrelevant by creating a leap in value for all of their potential users. They did this by creating a marketplace that improves upon the weaknesses of all of their competition. Airbnb is more trustworthy than Craigslist. It has much more inventory than HomeAway and VRBO because listings are free. They provide an incredible amount of value along the way - from the online experience (booking/subletting) to the real-world experience (showing up on vacation/getting paid for your sublet).

If you are looking to create such a value proposition, then read on. To create a Blue Ocean product, you need to change the way that people think about doing something. Value innovation is about changing the rules of the game.

Airbnb did this by enabling a free-market sub-economy in which quality and trust were given a high value that spanned the entire customer journey from the online experience to the real world experience. And they catered to both of their customers (sub letters and renters) with distinct feature sets that turned what was once a creepy endeavor (short-term subletting) into something with incredible potential for everybody involved.

Figure 2-6. Visual illustration showing chart of value props for both hosts and guests corresponding to their distinct feature sets

- Gives people an easy way for people to monetize their extra space and showcase it to an audience of millions.

[9] "NY official: Airbnb stay illegal; host fined $2,400" (http://tinyurl.com/k7oyx3j) C|Net. May 20, 2013

- Gives people an easy way to discover amazing and affordable places to stay all over the world.
- A seamless user experience for both sides where the process of subletting and booking leaves both sides feeling pretty great about themselves.
- A pricing strategy where Airbnb takes a nominal cut (3%) of both sides of the transaction. Using Airbnb from the customer side now makes it less expensive and better than traditional methods. (Blue ocean)
- Risk is low similar to Ebay because of user reviews of both sides. (Trust is value innovation.)

There are many other products causing widespread disruption to the status quo. Uber, which matches drivers with people who need rides, is threatening the taxi and limousine industry. Kickstarter is changing the way businesses are financed. Twitter is disrupting how we get news. And we can never forget how Craig's List broke the business models of local newspapers by providing us a far superior system for personal listings.

TENET 3 - VALIDATED USER RESEARCH

Not realizing a product's value is one of the primary reasons that the product will fail. Stakeholders are dreamers in that they assume - instead of verify - what is valuable to their customers. Much like Kevin Costner in the movie *Field of Dreams,* these entrepreneurs believe that if they build it, they [the users] will come. But the truth is that any product is a risk. Remember our software engineer at the beginning of this book? His assumptions about what his customers wanted turned out to be wrong. His heart was in the right place. His idea was timely, different, very innovative and even had a unique and sustainable business model. But the users didn't come. And when my team eventually went out and asked his target users, we found out they wouldn't have paid for the product as it was being positioned.

User research is how you verify that you're on the right track with your value proposition. There are lots of ways to do it—ethnographic field studies, contextual inquiries, focus groups, diaries and journals, card sorting, eye-tracking, personas, and more. I don't want to talk about any of these traditional methods. Instead, I want to talk about Lean Startup.

It's weird to admit, but before 2011 when Eric Ries' *Lean Startup*[10] (which you must read) went critical mass, founders didn't make it their mission to confront customers "early and often." The empirical, fast-moving, and transparent nature of Lean Startup riffed on ideas from Steve Blank's customer development methodology[11] and the highly theoretical Design Thinking approach. Sure, organizations had UX designers around to do "user-centric" design (as opposed to engineer-centric), but Lean Startup made conducting validated user research a make-or-break aspect of moving forward on a product. Lean Startup forced user research to become measurable.

This leads us to our third tenet—validated user research. "Validation" is the secret sauce of the Lean Startup business approach. Validation is the process of *confirming* that a specific customer segment finds value in your product. Without validation, you are simply *assuming* customers will find use for your product. Validated

[10] Eric Ries, *Lean Startup*. HarperBusiness, 2011

[11] Steve Blank, *The Four Steps to the Epiphany*. K&S Ranch Press, 2005

user research goes beyond just observing and establishing empathy for potential users. It is a process based on a reality-check that focuses on direct feedback from interaction with users. It helps your team determine if the vision of your product is a dream or a potential nightmare.

Eric Ries popularized the term “Minimum Viable Product.” It simply means learning if potential customers want your product by building just the core features of your value prop. This is far different from traditional product development in which building a prototype was often a simulation to show potential investors the future product. Instead, we build the MVP to test if customers want the product. By getting customer buy-in on our value proposition we are de-risking our product. And if users don’t like what they see, then we need to either “pivot” to a different customer segment or pivot to a different problem that our value prop can address.

Iterations like the MVP require your team to immediately conduct research and gain validation before developing a solution. It helps verify that your team is targeting the RIGHT customer (something our rehab startup in Chapter 1 failed to do) and not just a general persona. Once you’ve validated that a specific pain-point addresses your end user’s needs and wants you will continue to add features, then test those features using the same research methods. This is known as the Lean Startup feedback loop of build-measure-learn. Use your research to validate your decisions and ensure the product vision is aligned with the end user’s needs. Keep doing this and eventually you’ll have a killer user experience design.

Validated user research is a collaborative process that should involve as mean members of the product team as possible. Collaboration will actually help organically build consensus on the value proposition. Now this may sound naive since we are all working in different environments with a range of folks with dynamic personalities who are in various positions of power. In an enterprise environment, there are typically many stakeholders who each have a say on the product requirements based on their personal agenda or preference. When I work for agencies, the product requirements are typically locked in stone during a requirements gathering phase that I’m not involved in. For me to suggest doing validated user research or creating an MVP to test during the design phase is blasphemy because it’s counterintuitive to the agency model. The last thing an account executive wants to hear from their UX resources are ways to cut the project fee down for their client.

But if you find yourself in this familiar position, that’s the exact moment that you need to become *intrapreneurial*. Intrapreneurship is the act of behaving like an entrepreneur while working within a large organization. You need to decide to take the fate of the product into your own hands through assertive risk-taking and innovation. Stand up and ask for the extra week or two to conduct validated user research. If you get a “no” or are too afraid to ask, then it’s time to start working off-hours. The worst thing that can happen is that you will discover something about yourself and/or start looking for ways to improve your own work process.

The bottom line is that confronting your users is non-negotiable. We must learn as quickly as possible if the idea we are working on is stupid and worthless. We need to have an open mind to experiment and to fail. That’s right, we are betting. And the odds are against us. But in the end, this approach is more cost-effective and efficient.

TENET 4 - KILLER USER EXPERIENCE DESIGN

Concision is the art and practice of using no more words than necessary to convey an idea. Airbnb facilitates a positive and trustworthy customer experience that starts online with a transaction and then moves offline in real-time. It culminates with both participants in the transaction hoping for positive reviews from each other. The entire user experience is built around the idea of making sure each guest and each host is a good customer. Airbnb not only provides a good user experience, it asks its users to help build it as well.

For those of you who don't know, the "user experience" (UX) is how a human feels when using the interface of a digital product while attempting to accomplish a task or goal. Yes, we can say a door handle is an interface and go off the non-digital highway into the world of 100% physical products. But in practice, the term user experience refers to whether a person had a good or bad time trying to utilize a digital product. The user experience of Vine starts when a mobile phone user opens the app and pushes down anywhere on the screen to capture a video. With simply one touch they're instantly recording video of a special moment – maybe their child's first lick of an ice-cream cone. Then painlessly with just a couple more taps, that video automatically lands on the Interwebs where grandma can immediately watch right from her Facebook feed.

Killer UX Design is accomplished when a product allows its users to easily accomplish a goal with little resistance. Simply put, killer UX design enables an experience that is FRICTIONLESS! A killer user experience is also when a product team INVENTS something that people can't live without, or at least don't want to give up because it's so much better than the way they used to do things. I know I would be hugely bummed if you took away the experiences afforded to me by my iPhone, LinkedIn, Airbnb, and Google docs.

But killer UX design doesn't happen by magic. It must be informed by business strategy, value innovation, and validated user research. And these are things that user experience designers don't typically do.

Traditionally (if I dare use that word for a discipline barely two decades old), user experience design is associated with deliverables for development and design execution—site maps, wireframes, process/task flows and functional specifications. Recruiters for enterprises and agencies identify user experience design with the job titles that create these deliverables, including interaction designer, information architect and UX designers. These definitions are used by large enterprises and agencies and are pretty much how user experience design is currently practiced.

What ultimately happens in this "traditional" system is that the UX designer is often excluded. They miss the boat when it comes to making the important product decisions. Sometimes this is the fault of the UX designer because she is still learning the craft, or worse, perfectly content to hide behind her monitor knocking out wireframes. But if those wireframes are not informed by the business strategy and the user research, then stakeholders can expect mediocrity and copycat design.

Killer UX design is accomplished when the UX designer works collaboratively with stakeholders and teammates at the idea's inception. Then, the UX designer can be involved in designing structured experiments. These experiments need to be focused on how successful the value proposition can be communicated to the customer from the moment the customer hits the landing page. Using measurable results, design decisions can be made based on real evidence rather than hunches.

Killer UX design is also about determining which key moments and features are absolutely critical to your product. We will explore techniques such as storyboarding and journey mapping that will weave key experiences together in simple and elegant ways. We will dissect successful products such as Vine and Snapchat that offer a "less is more" design approach.

Figure 2-7. Screen shots of Vine, Snapchat, Waze interfaces

Traditionally, UX designers tend to be more concerned with user engagement and design than with customer development and revenue generation. But it's the latter that really concern business strategy. Decisions about user experience design should not be subjective because its goals are objective. You want to convert new users into satisfied customers who tell their friends to try your product. The common problem that UX designers really need to be aware of is just how much their decisions are tied to customer acquisition. Just think about any transactional website or even a simple sign-up process. The UX design should be very concerned with barriers to entry, which may prevent customers—even validated customers who have previously engaged with the product – from converting to customers (We'll talk more about this in Chapter 10 - Designing for Conversion.) Wireframe layouts should be geared towards the desired response of the user. This is where UX design can really positively affect the visual design and content strategy because the look, feel, messaging, and calls-to-action really matter!

Once a UX designer becomes proficient using these tools and techniques they begin to morph into a UX strategist. But as I said in the beginning of this Chapter, UX strategists are currently unicorns. You really need to pay your dues working as a UX designer, front-end developer and/or project manager because there's a lot to understand about how software should work. And it takes time to understand the big picture.

With practice and mindfulness, the UX designer will come to understand the product as a sum of its parts. This means that she must take the lead and define the product with features that balance user value and business value. She also must balance the level of effort needed to implement those features. To get a product out the door, the UX designer must be able to work collaboratively with stakeholders and teammates. It takes a lot of emotional intelligence to sell your UX concepts to both clients and developers. I've learned from experience that people don't like working with "genius designers" or "design heroes." So don't expect to get a "seat at the table" by acting like an asshole.

Sidebar - Top 10 Not-Strategies!

1. A killer idea for a new product!

2. A laundry list of features!

3. A thoroughly researched game plan where all possible scenarios have been considered and is ready for implementation. No need for customer feedback because you are 100% certain you have nailed it!

4. A creative permutation of trending buzzwords that were just used by another startup that raised financing. (i.e. Peer-to-peer Sharing Economies)

5. A generic set of motivational statements (i.e. Go Team Challenge Conquer)

6. An arrogant statement from some expert—"Our product sprung from the genius of Professor Iam Awesome, the visionary of Social Lean Disruption."

7. A hypothesis that has unvalidated risky assumptions—"Well, all women DO like pink."

8. A grandiose vision that doesn't align with their core values that your company has no capability of delivering (i.e. A patent-pending, new-method-of-discovery dream).

9. A vague affirmation that sounds like a good Hallmark card—"You, too, can achieve Social Lean Disruption."

10. The North Star

RECAP

I certainly don't miss the heyday of Flash when designers went off the deep end creating game-like web interfaces for new digital products. But if we are inventing a product that does something new, that's our chance to make a new convention. In *Lean Entrepreneur*[12] Patrick Vlaskovits and Brant Cooper advocate "If you are doing best practices, you are not innovating." This is a provocative statement because established interaction design patterns help make consistent user experiences. But there is no harm in breaking a rule or two through experimentation. That's what Amazon did to become the superpower of successful transactions. By optimizing our product's entire user experience, we help Alice trip down the rabbit hole.

That is what UX Strategy is.

So the real trick is to keep all four of these tenet "plates" spinning in the air.

Figure 2-8. Illustration of a hipster-looking clown juggling plates

In short, we need to think clearly to identify the problem we are trying to solve, and who our customer is. We need to learn everything about the existing market space. We need to be analytical so that we can identify opportunities that can be exploited. We need to pack a one-two punch of incredible user experience coupled with an innovative business model.

There's an existential reason that we want to get good at this. Life is short! Why waste our finite time on earth making something that nobody wants or uses? Who knows… the whole planet could blow up tomorrow and we all might come back as slimy snakes with no hands and maybe even no Internet! So instead, let's do things to safeguard our stakeholders and ourselves from the high probability of failure by "de-risking" our future products.

[12] Patrick Vlaskovits and Brant Cooper, *Lean Entrepreneur*. Wiley, 2013

O'REILLY®

Designing with Data

IMPROVING THE USER EXPERIENCE WITH A/B TESTING

Rochelle King &
Elizabeth F. Churchill

Data Driven vs Data Informed Design: Does It Matter?

On March 14th, 2008, a product manager at Google began her keynote at SIGCSE 2008 in Portland, Oregon. Entitled "Innovation, Design and Simplicity at Google", Marissa Mayer outlined the vision that she had spearheaded at Google for almost a decade.

In one section of her talk, she off-handedly explained the importance of testing different designs at Google, specifically for Google News.

"There was a huge amount of iteration. I think that our UI designer had had it with us by the end of that process, 64 iterations later. It was very much worth it – Edward Tufte complimented [Google News] as the best designed page on the web. I think that the reason that design has come to be is because of that iteration, and the acceptance that we won't get it right the very first time we do something. But we will try very hard to get data and evolve the design in a very scientific way. **I like to think of design as much more of a science than an art**. Let's not say "top navigation is better than left", let's not say "this screen is better in blue", let's just put them both on the site, in what we call a split AB test where we take 1% of our user population and give them different designs and then see which one actually performs better."[1]

She repeated this mantra in several keynotes over the subsequent year. At her AIGA keynote in October of 2009, she went into even more depth using examples to show how Google makes design decisions.

1

The fourth slide of Marissa Mayer's AIGA October 2009 Keynote

"One of the techniques that we use is now known as split AB testing. Amazon developed it around the same time Google did. We did our first split AB test in 2000. And what it basically means is that you take your user population and you split them into groups A and group B and you give them different experiences. The idea being that then you can see, in a natural situation, what would happen if you made that change on the site. And, for all of our changes – you know, all design starts with an idea – but then we take that idea, we'll do a usability test where we look at qualitatively, "are there any large problems?", "is it generally understandable?". And once you get through those two hurdles, then it's really about putting it on the site and letting the data – letting the metrics decide. SO the best design along various metrics is what wins."

She goes on to explain, with data, why she had designers test 41 shades of blue within gmail. At the time of writing, the full video can be found on the AIGA website, free of charge[2].

As many remember, this was extremely contentious. Is testing 41 shades of blue still design? When algorithms and humans collide, at what point does the human end and skynet begin? This was on Doug Bowman's mind in March of 2009. In a very public departure, he derided Google's practice of data driven design.

[2] Marissa Mayer's October 2009 AIGA keynote http://www.aiga.org/video-makethink-2009-mayer/

"Yes, it's true that a team at Google couldn't decide between two blues, so they're testing 41 shades between each blue to see which one performs better. I had a recent debate over whether a border should be 3, 4 or 5 pixels wide, and was asked to prove my case. I can't operate in an environment like that. I've grown tired of debating such minuscule design decisions. There are more exciting design problems in this world to tackle.

I can't fault Google for this reliance on data. And I can't exactly point to financial failure or a shrinking number of users to prove it has done anything wrong. Billions of shareholder dollars are at stake."[3]

It took five years for Google to respond in very tangible terms to Doug's complaint. Dan Cobley, Google's UK managing director, in speaking at a DLA Piper event, revealed the following insight.

"And we saw which shades of blue people liked the most, demonstrated by how much they clicked on them. As a result we learned that a slightly purpler shade of blue was more conducive to clicking than a slightly greener shade of blue, and gee whizz, we made a decision. But the implications of that for us, given the scale of our business, was that we made an extra $200m a year in ad revenue."[4]

Two hundred million dollars a year. Think about that. A single six digit hex color code made Google and extra two hundred million dollars a year.

But what about the rest of us mere mortal working in smaller companies? For those of that don't have the words "VP at" Google, Facebook, Baidu, Twitter LinkedIn on our resumes, do these same rules apply? And before we even begin to answer that, do the words "data-driven" or "data-informed" even make sense?

Data Informed vs Data Driven

The word "data driven" has, by now, long been part of buzzword-bingo card sets. It's been heard in the halls of the web analytics conference "eMetrics" for over a decade with countless sessions aimed at teaching audience members how to turn their organizations into a data-driven one.

When spoken of in a positive light, the word "data driven" conjures visions of organizations with endless streams of silver-bullet reports like the one shown in figure 2.

[3] http://stopdesign.com/archive/2009/03/20/goodbye-google.html

[4] http://www.theguardian.com/technology/2014/feb/05/why-google-engineers-designers

A Data-Driven Organization with Data-Driven Reports

You know the ones – they're entitled "This Chart Will Help Us Fix Everything" that show how a surprise change can lead to a quadrillion increase in revenue along with world peace. One shade of blue – a 200 million dollar yearly increase in revenues.

When spoken of in a negative light, the term is thrown around as a descriptor of Orwellian organizations with panopticon-level data collection methods; with management imprisoned by relentless reporting, leaving no room for real innovation.

On Being Data Driven

I think it's kind of problematic how data-driven so many companies are today, as crazy as that sounds. I see this mentality that I think is common, especially in Silicon Valley with engineer-driven start-ups who think they can test their way to success. They don't acknowledge the dip. And with really hard problems, you don't see market success right away. You have to be willing to go through the dark forest and believe that there's something down there worth fighting the dragons for, because if you don't, you'll never do anything good."

Evan Williams, Co-Founder, Blogger & Twitter

One thing is certain. Businesses can't do LinkedIn-level data driven decision making with the yearly budget of a half of one of LinkedIn employees' salary. As we'll see later on in this chapter, real data-driven decision making isn't *even possible* at certain stages in the development of a company. Early-stage companies have, in most cases, far more pressing issues than split testing the color of a button.

One of the best descriptions that we've ever seen on the difference between data-driven and data-informed comes by way of Andrew Chen. In a well referenced post entitled "Know the difference between data-informed and data-driven" [5], he explains that "[…]the difference […] in my mind, is that you weigh the data as one piece of a messy problem you're solving with thousands of constantly changing variables. While data is concrete, it is often systematically biased. It's also not the right tool, because not everything is an optimization problem. And delegating your decision-making to only what you can measure right now often de-prioritizes more important macro aspects of the problem."

The words "not everything is an optimization problem" sum up the philosophy behind this book very well. We also want to broaden the definition of "data" from merely being AB test results to representing a wide breadth of information that you can gather from many different sources. We want to acknowledge that the intersection of data and design can be so much more than AB testing and to recognize that it can be equal parts art and science.

To be honest, the great debate between "data-driven" vs. "data-informed" can sometimes feel like semantics. It is never as black and white as the blogs might have us believe. There are people that call themselves "data-driven" but in practice exercise judgment and gut to inform their decisions as much as the data and even if you place yourself in the "data-informed" camp, if you aren't using the data correctly then you may run as much risk at doing as much or even more damage than someone who religiously uses data to drive every single decision. We generally felt though, that data-informed better represented the balance we are trying to strike and this is why data-driven design wasn't the title we chose for this book

What Is Data Informed Design?

Data-informed design is approach to design where decisions are informed by specific, objective evidence; data. And though data comes in many forms, and it is collected in a number of different ways, it should be used to give teams a better understanding of what customers are doing with a product and how they are reacting to the changes made to it.

Being smart about data-informed decision making has considerable advantages. It helps managers hire smarter, and helps designers

Having common success metrics within your company can also help designers and the broader product team to align around common goals and to understand what kind of data is the most important to track and follow.

---

[5] http://andrewchen.co/2012/05/29/know-the-difference-between-data-informed-and-versus-data-driven/

So more specifically, for our purposes, data-informed design is a framework that you can use as a designer to help you hone your understanding of customer behavior, align you and your team to the larger company objectives and business goals.

Using a data-informed approach can be a core foundational practice and primary methodology for how you and your teams can think about design - but as with most things in life it's not "all or nothing". There are other things to take into consideration when designing as well.

There is a general perception that taking a data-informed approach will limit your ability to take large leaps in innovation or that it stifles creativity. Many raise the concern that using a data driven approach limits designers by taking the "art" out of design, reducing it to just the "science". We believe that it is most important to recognize that solely employing a data-driven approach will not be enough. For us, being data-informed is less about being a "yes" or "no" question and it's much more about being aware that there is a spectrum on which you can operate. Depending on the nature of the project, where you are in the development timeline and the product itself - it may make more or less sense to rely on data. Generally speaking, when people talk about the opposite of "data-informed", they mean relying on experience, gut or instinct or making decisions in absence of having them be vetted by data.

We believe that relying on your instinct and experience is essential during the early stages of a product. When you find that you are trying to pivot or make a dramatic change in your product direction - those initial ideas and product solutions are born from your instinct and experience as a designer. When you are defining principles for your brand or product and thinking about how you want to express these beliefs to your customers - you again rely more on your core beliefs, instincts and experience as a designer.

A significant part of a product designer's job is still the "art and craft" of design. That is leveraging training, experience, instinct and gut based on years of perfecting the craft of creating great experiences in terms of both visual treatment and user experience (flows, information architecture). If you take two designers, one that is mediocre and one that is great, and they both employ a data-driven framework. You'll find that the better designer will still be able to propose stronger and more impactful solutions compared to the mediocre designer. However, in a data-informed framework you might actually be able to measure how significant that difference is.

Now that we've outlined our basic philosophy on data-informed design. It's important to recognize that applying this framework is not "one size fits all". A key fundamental consideration is finding a baseline before you even start to optimize and tune your approach. The type of business you're in and the stage that your business is in will matter tremendously. The following section will help you identify what you should be watching (before optimizing) based on a few simple rules of thumb. We'll end with a cherry on top. Rochelle will close the chapter with a personal, hilarious and face-palm-worthy anecdote of the process of data-informed design, in practice, at Neftlix.

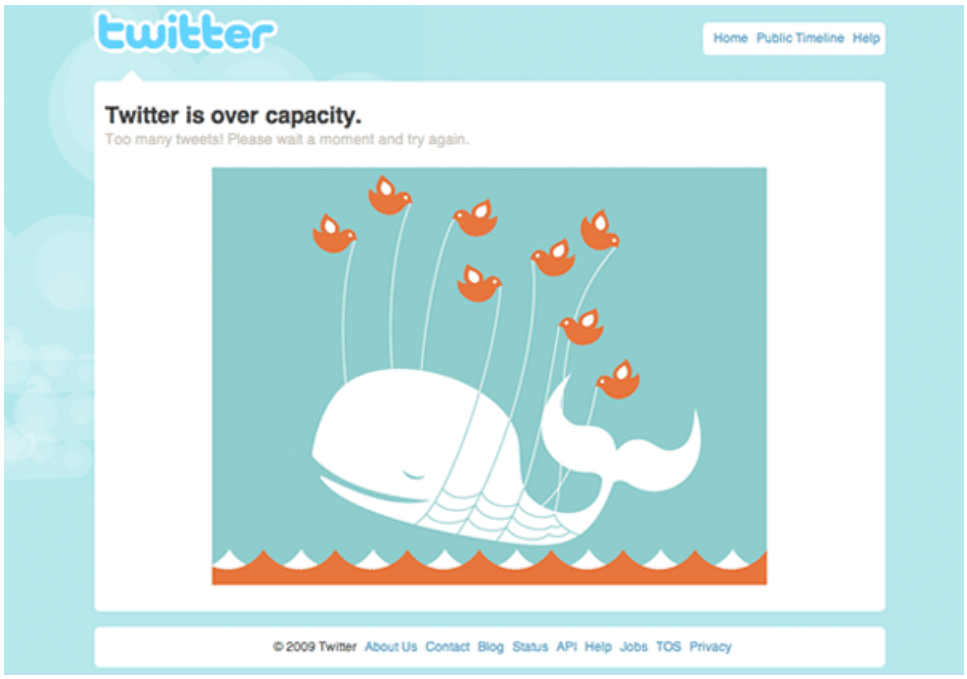
What kind of business are you in?

The kind of business you're in has a huge impact on what you'll care about on a day to day basis. Said another way, working in any position at a social network startup won't necessarily give you the necessary skills to work in a SaaS company, because the considerations of the businesses are very different.

One of the examples I find that illustrates this well is a short tale of two websites going down. Consider gmail and Twitter.

In the early days of Twitter, the platform went down very, very often. The root cause came down to scale – too many people trying to use the product at once. The underlying technology couldn't support it, which lead to many users getting error messages. And that error message was a picture created by Yiying Lu, later dubbed the "Fail Whale" by twitter user @qrush (Nick Quaranto)[6].

"[Yiying Lu] eventually put the image on iStockPhoto.com, where Stone [co-founder of Twitter] spotted it and decided it could serve as an error message that was both cute and symbolic. Better than the LOL-cats they were using at the time, anyway (remember, this was 2008). "I wanted to find something that was still fun but that wasn't so jokey," he told NPR. "This idea [is] that it's a big job but we're all working together to do it."



As a result, when Twitter used to go down, many people seemed to be delighted. Shouts of glee would hit walls on Facebook when Twitter went down.

[6] https://twitter.com/qrush/status/822613478

Business Insider, The Next Web, BBC News, Times Magazine and Read Write Web are all examples of publications that have called the Fail Whale "Iconic" at least once.[7] In short, people love the fail whale.

Compare this to what happens when gmail goes down.

[7] https://www.google.ca/search?q=fail+whale+iconic&oq=fail+whale+iconic

The Culture of Data

Over the course of this book, we'll be covering the fundamentals of setting up a strong framework for incorporating data in your day to day decision making process. We're also going to talk about how the data that you're interested in may differ depending on what kind of business you're in or what stage of growth your company is in, how to get a baseline and then use that data to help inform you as you develop and optimize your product.

While having all the mechanics in place to embrace data as part of the decision making process is important, it's equally important that you have a culture that supports it as well. Creating a culture where all of this comes together well really depends on two key things: the process and the people. It might seem a bit odd to start this book with the chapter on culture, and indeed it was originally the last chapter in the book. But as we wrote more and more of the book, we realized that this first section had to be about putting all the pieces in place for creating a good foundation that you can build on as you learn more about the actual techniques.

By making you aware now about the cultural implications of being in an environment that embraces data as part of the design and product development process, you'll have this in the back of your head as you read through the more practical chapters on how to gather, analyze and interpret the data. This way you can consider what you'll need to do to set yourself up for success when you actually go to implement some of these methods yourself. In the same way that we wanted this book to represent a balanced approach to data and design, we also wanted this book to provide you with a 360 degree approach to implementing data as a framework - which means thinking about the culture as well as the methods.

In this chapter, we'll cover two things:

1) What a data informed culture looks like
2) Some tactics that you can use to get there

I hope we can show you how everything from the vocabulary that you use to the people you hire can help to foster a culture where data informed design can thrive and give you some concrete tools that you might consider using as you go about investigating what a data informed environment looks like at your company. We'll talk to a few companies where the culture of data and design has flourished, but we'll also talk to some places where it didn't work out so well and see what we can learn from those places. In the end, company culture is such a unique and precious thing that we would never want to dictate what culture you should adopt. We only hope to provide you with as much information as possible, so that you can pick and choose which things make the most sense to incorporate into your own situation.

Why does the culture matter to data?

You might be reading this book because your company currently takes a data informed approach to product development and you want to better equip yourself with the tools you need to engage in those conversations. You may also be reading this book because your company currently doesn't leverage data in it's product cycle and it's something that you're thinking of introducing. Either way, whether something is new for you or new for the company it's really important to recognize that the success or failure of an approach or framework is going to be dependent on the environment that you are working in.

With respect to data, there are a handful of things that we've seen work really well at different companies and that are really important have as part of the company DNA. Some of these things apply more at the individual level, and some are ore dependent on the group. If you really want to be effective at integrating any kind of framework into your company, you'll want it in your DNA/Culture. It's a hundred times easier to adopt something when the culture supports it … and that's why the success of a data informed framework is really going to depend on the culture at your company. Some key components that we've seen work in various environments:

- The value and role of data is **universally embraced and understood**
- A **common vocabulary/language** when talking about data
- A **disciplined and consistent** approach to leveraging data
- The organization is encouraged to **continually learn** and grow their knowledge base (and data can play a part in that)
- The **hiring process and the people** who are hired reflect and supports the above

We'll walk through each of these points to help you think about which things you want to adopt for your organization and which may not make as much sense for you or your team. At the end of the chapter, we'll suggest some tactics that you might use as you build up your own culture of data informed design.

Universally embraced and understood

For a data informed approach to be successful in any organization, it must be both universally embraced and understood. By "universal" we mean both in terms of depth (that is accepted at all levels, from senior executives to individual contributors) and breadth (that it is accepted throughout the company, beyond the product development organization as well). Specifically, this means that people at all levels in the company and across a broad set of functions within the company recognize that data is used to inform product decisions. It also means that they have at least a basic understanding of what this process looks like and what the pros and cons are of using data are. This means that there is a good "support net" for data informed design to happen – having this level of transparency in the process and common understanding can help to keep the design and product teams accountable to good practices.

Depth: Top to bottom

Because data driven design is by definition about breaking the hierarchical decision making structure it must be accepted by all ranks in an organization. In fact, it may even be most important for it to be driven by the top levels of management since their voices carry less weight in the day to day decisions when you employ a data informed framework. This is because fundamentally, what you are doing is looking to the "data" that you are getting directly from your customers when making your product decisions. Just to reiterate something that we talked about in Chapter 1, this data may be collected using any number of techniques (survey, usability research, AB testing, etc.) but it's all coming from your customers and much less about having individual executives drive those decisions (and recognizing the possibility that those opinions can be subjective). An excellent early paper about overall AB testing published in 2007 by Ron Kohavi, Randal Henne and Dan Sommerfield titled "Practical Guide to Controlled Expriements on the Web: Listen to Your Customers not to the HIPPO"[1] actually calls this phenomenon out in the title of the paper. This paper points out a fairly familiar situation:

"Many organizations have strong managers who have strong opinions, but lack data, so we started to use the term HiPPO, which stands for Highest Paid Person's Opinion, as a way to remind everyone that success really depends on the users' perceptions."

If you are trying to shift into a data informed framework, it will be critical to look at the way product and design decisions were made previously made. If these decisions were largely in the hands of one or two individuals then it's worth taking the time to make sure that they can understand and recognize the value of bringing in customer data into the decision making process because in the end, they may find that their weight is diminished (as is the "weight" of any individual opinion). This is a good point to check with yourself as well – are you comfortable and willing to see how depending more on data will change the weight of your own opinion within the design and product development process? Is this something that you are comfortable with personally?

Breadth: beyond design and product

In addition to getting all levels of an organization on board with a data informed approach, it can be just as important for teams outside of product and design to have a basic understanding of a data informed framework and how it's being employed. Personally, I've always appreciated company cultures that embrace transparency.

I've found that data informed decision making can be a really great way to bring in organizations that are not normally part of the design & product development process and make them feel like they have a closer affinity with how the product is shaped. A data informed process is a bit more democratic since it relies mostly on the "voice of the user" to vote with their behavior. If people in organizations beyond product and design feel like they can empathize with your users, then they can sometimes feel empowered as part of this process. In addition, by clearly NOT making decisions only relegated to the arena of the HiPPO, it can also remove some of the abstract, black box, mojo that can intimidate or make other parts of your company skeptical about the decisions that are being made. In other words, a data informed approach is something that many people beyond product and design can understand. By being more transparent with the rest of the company about how product and design decisions are being made (and especially in product and design driven companies), I have seen it build broader support within the company for the decisions that are being made.

There is another positive side effect of having your data informed approach being broadly understood. It may initially seem like it's superfluous to have teams like HR, sales, or finance fundamentally understand a data driven approach. However, it is always helpful to have any team that you work with in any capacity understand how you do your work. In my experience, having HR understand how much we rely on data has helped them recruit better candidates for my teams. They have helped me articulate and recognize the profile of candidate that would work best in this environment (more on recruiting later in this chapter).

Whenever I've done product development related to features like payments or ads, having a finance and sales teams that understand the way that you look to data to support or disprove various ideas is also incredibly helpful. They can become even more engaged in the idea generation stage and we become better aligned about what factors we are looking to when determining what is considered successful or not. As much as HiPPOs can existing within Product and Design organizations, they can also exist in other organizations as well. Imagine the sales executive who feels that they know exactly what you need to build into your product but may be overly swayed by a key vocal client. If that sales executive understands that you build product by looking at the data, then they realize that they need to start to articulate their arguments into the same framework - and it helps to get you and them aligned with what you are building, why you are building it and what impact you think it will have on your overall user base.

There is a huge difference between feeling like using data to make and inform decisions is something that just one part of the company uses as a tool and having it be universally embraced by the entire organization. To get this level of understanding throughout the

company, using data has to be something that you actively share within all parts of the company and has to be something that is shared broadly (e.g. in company all hands meetings and other public forums).

<Sidebar: culture & data/>

Patty McCord – former Chief Talent Officer, Netflix

One of the things that made being data driven so successful at Netflix was that it was a well established practice that was universally accepted throughout the company. It may have started at the top with Reed and Neil (Netflix CEO and CPO) driving and believing in the data driven approach, but it was also embraced by everyone through management to the individual contributors on the team.

For example, when Reed would engage in product testing and had a strong idea for what he wanted to see in the product, he would submit his ideas to the same rigor of AB testing as everyone else in the product team. On rare occasions, Reed might dictate the exact way that a product idea would work, but he would do this on "his cell". This cell would be tested against other cells that were proposed by the product team. If it didn't win, then it didn't get launched. Reed even went so far as to have a philosophy that he had a certain number of "bullets" that he could use. If he continued to propose failing test cells, then at some point the product team wouldn't have to indulge him in creating these cells any more. Having the upper level management hold themselves accountable to the same process as the individual teams definitely reinforced that it was the way we wanted to work.

Something here about how Reed used to always say "we don't have a Steve Jobs, so we use data"

In fact, data driven design was so much a part of how we worked in the Netflix product organization that the philosophy was actually understood outside of the product and technology group. It wouldn't be surprising at all to find a finance or HR person asking us at an all hands meeting when we launched new product what the original hypothesis was and what test cells went into the test.

Common vocabulary/language

A key part of establishing culture is the language that you use. Think about the words that you most often hear at your company. What are the words and phrases that are unique to your company? Which acronyms have you created and embraced? When you first joined what were the things that stood out to you as unique? Now think about how those words and phrases help to set a tone for how business is conducted within the company and how usage of certain phrases make you feel like you're part of an "in-group" or "in the know". To make a data informed decision making framework is universally embraced, it's then important to actually have a common vocabulary and way of speaking about it.

You'll find that a common vocabulary will help in a number of ways, from making brainstorms more effective to making debate over product and design decisions even more rich because you'll have a shorthand for the things that are most important to the discussion. Seemingly simple questions that can get asked over and over again, "How are we going to measure X?", "How do you expect the majority of users will react?" or "What do you expect to learn about user behavior from this design?" can quickly help to train people and set expectations about the role that data can play in decision making. Words and phrase can articulate both positive and negative things – remember the "HiPPO"? What do you think it would feel like if you were called out as succumbing to the "HiPPO" on a project?

Remember though, this should be as much about having common definitions as it is about having a common vocabulary. Since a data informed environment requires a certain amount of discipline and rigor to the practice, you can't ensure consistent application of this framework if you don't also have alignment on your definitions as well as your vocabulary. Another pitfall to look out for is that it can be easy to get caught up in the words and acronyms (especially around metrics) and that you forget that ultimately this is all about your users. Find ways to actively represent the user in your definitions so that everyone recognizes that the data doesn't become dehumanizing and is instead always a reminder of the people that you are building your product for.

Key phrases and vocabulary will be defined throughout this book, however this is a short list of some of the key concepts and phrases to look out for.:

- Hypothesis - by stating "my hypothesis is..." it helps to place your idea within the context of everyone else's ideas. Calling it a "hypothesis" reminds you that it hasn't been proven yet and that it needs to be tested. Hypothesis also has the added benefit of sounding more scientific, in some ways I believe that this makes people less attached to their ideas and therefore more open to testing them and being proven wrong.

- Metrics - We'll spend a lot of time in this book defining how to find the right metrics for your company, and identifying the difference between core, proxy and secondary metrics. For the purpose of this section, "metrics" is a key part of the data informed vocabulary because the metrics that you use and measure represents how you measure success. Clear understanding and agreement on how you are defining metrics (at the company level and at the project level) will ultimately be an indicator of how aligned you are on any given project or discussion and can therefore play a significant role in how effective your discussions are. You might find that you start to use questions like "is this change really going to *move the metrics*?" to ask if you're really investing your time into a project that will have a measurable impact on your users. Questions like "which metrics do you think this will affect?" can help you to make sure you're looking to impact the right level of user behavior.

- Statistically significant - a great shorthand for knowing whether or not you should act on the results from your test. You can always refer back to whether or not the results were "statistically significant".

In reality, the vocabulary itself may not be as important.it's really more about consistent usage of this vocabulary that helps to reinforce a specific mindset and a way of thinking. Like working in any company, the vocabulary can help to shape the culture. As you read

the book, it will be a good exercise for you to think about how you might turn some of the above into phrases that will work out for your unique organization. In the sidebar from Leisa Reichelt you'll see how she instituted a program for everyone to sit in on at least 4 hours of user research every 6 weeks and how it became common practice to ask people "Have you done your 4 hours?" What are the words that are already part of your company culture that you can adopt to support a data informed approach?

Be disciplined & consistent

Later in this chapter, we'll talk more about how to build up the culture of data informed design. As you begin applying a data informed framework, there's no way to just turn it "on" so you'll start with selectively. However, your goal should be to make data informed design a consistent part of product development. To do that, it will require discipline and vigilance. This is probably true of any decision making platform, but if you're going to take a data informed approach to product and design it's important to not apply this framework selectively.

Fundamentally, taking a data informed approach to design is about trying to make decisions about your product and for your customers using as much information as possible along the way. It's about trying to mitigate the subjective aspects of product development by incorporating data from your customers into the process. It's therefore almost inherent to the nature of a data informed design process that you need to apply it consistently across all projects for it to work. I've seen that using data inconsistently within an organization can often cause confusion or skepticism about the larger process.

One of the critiques that I've heard about using data inconsistently is that the data is only used when it supports the opinion of the person who's making the decision. This can definitely happen in organizations and as many of us might know from lab courses in college or high school, there are often ways to "fudge" the data or read it in a way that makes it support the outcome you're looking for. As you're trying to establish the value of data informed design, it's even more critical that they way you leverage and interpret data is such that it reinforces the objective value that it brings.

As you'll see in later chapters of this book, data informed design really does require a balance between subjectivity and objectivity. You will subjectively decide what to test and what data to gather when crafting your solutions or making your decisions. As a designer or Product manager working in this environment, you might find that there can be a dose of skepticism around which cells are being tested and why. Therefore, it becomes important to treat all projects with the same amount of rigor across the board. Crafting solutions and making decisions in absence of data and user feedback should become the exception and when you do make those decisions without testing or user feedback you'll want to be able to articulate and defend why that was the case.

Especially when you are trying to establish the methodology of data informed design, it can be important to showing how you consistently hold yourself accountable to the methodology of data informed design, but ask others to hold you accountable as well. This goes back to the first two points about having data informed design be universally embraced and having a common vocabulary. If you have those two things in place, then your whole company will be helping to keep you and your team accountable as well.

Encourage Learning

I've noticed that data informed design environments work best when they are seen as part of an overall company culture that encourages learning. Many companies are focusing on creating "learning environments" where there is a focus on building individual skills as well as encouraging development. Data informed design can really lend itself to this environment because one of the core attitudes that you need to have about using a data informed framework is to recognize that you are constantly trying to build up and add to your knowledge about your customer's behavior. As we used to say at Netflix, using a data informed approach was fundamentally about "honing your consumer instinct". This is an ongoing activity and every project is an opportunity to get better at developing that instinct through learning, information and data.

There are some very practical habits and best practices that can really help to support this culture. Some things that I've seen be especially effective are:

- Sharing results and information broadly
- Keeping sharp on both theory and practice
- Self awareness/evaluation

We'll outline here a little about what each of those things look like, but give more detail on how to practically implement ongoing processes for each of these items in the second part of this chapter.

Share broadly

If you are actively using a data informed design framework, one of the best things that you can do for your company is to broadly share the results and to give everyone access to the data. In a data informed environment it becomes even more important that designers understand how their designs are being judged and performing. Designers are often very engaged with user research (which we also classify or include as part of the "data informed" framework because the design is still in progress and user research is often more integrated into the design process. However, there are many places where once the design is done, the designer disengages from the product development process and doesn't stay as engaged with the results post launch.

When I joined Netflix, one of the first things I did was to begin to ensure that the designers got the results from any AB tests that were done on their designs. Many of the designers mentioned that getting access to this data was a key shift in their ability to become more strategic. It was also important that they understood (at a high level) how

to analyze this data and how to then use that data to make informed decisions about what to design (and test) next. Of course, none of this could have happened were it not for the fundamental sharing of results in the first place. There are many reasons why it's really important to share results.

Habitual sharing/context setting/broad communication, company-wide, of test results - reinforces that most decisions are influenced by data. It also helps folks to build up shared knowledge about what works and what doesn't work. All this testing doesn't really get you much further if you're only using it to make decisions on a micro level. What should be happening is really taking away learning from these smaller instances and then using them to hone your instincts about your users over time. The best way to do this is to vet it with your peers and to subject your testing to debate and discussion.

Theory and practice

I've found that it's useful not only to share the results of ongoing work, but to also recognize that operating in a data informed environment can be new for a number of people. Sharing results from ongoing work means that you're sharing knowledge and insights which helps people develop a better understanding when it comes to your particular product or project, but it can also be helpful to make sure that people who have less experience with data informed design have an opportunity to build a good foundation about the practice itself.

Discussions and debate around more general topics and "theory" of data informed design can be very helpful to make sure that everyone has a common foundation. Topics might include everything from how to pick the right technique for different kinds of projects, the pros and cons of different methodologies, or even details on how to pick a statistically significant cell size or what a p-value is. The number and level of discussions that you have on these best practices will be determined by how mature the organization is with respect to data informed design. If everyone can have a common baseline of knowledge on the theory behind the practice of data informed design you'll find that your discussions on actual projects will be much richer and much more efficient. You'll also hopefully avoid some of the pitfalls of not using the data properly (and therefore being more likely to subjective or mis-interpretation). You'll find plenty of potential topics for these "theory" discussions as you read this book because we'll be covering some of those basics in the upcoming chapters.

Self awareness/evaluation

As we've mentioned before, one of the biggest benefits of a data informed approach to design is that it gives you a great tool to use to hone your consumer instinct over time. If part of the culture of data informed design is about transparency and a more objective evaluation of your work via the data then as we've hinted before accountability to holding everyone to a similar standard is important. It's really helpful to create an environment where you can encourage self evaluation and to not be afraid of making mistakes and building out ideas that don't actually "move the needle"… so long as you are keeping track of them and then learning from those experiences.

We talk in the next section about "keeping track of track records" as a good way to see how you're actually doing. Being self aware of when your instinct has differed from what has been reflected in the data is one of the fastest ways of improving and if you're in a culture where sharing data and results is very common, then you should have a lot of tools available for you to evaluate yourself. It's important to note that some of the pitfalls of being in a data informed environment (as outlined in Chapter 15) are bad habits that you can develop if you DON'T hold yourself accountable to your decisions and if you don't do this constant self evaluation. For example, "over-testing" can actually be a side effect of not having a strong enough instinct for what are the right things to test. Developing a good instinct for what will give you the best ROI will actually come from constant viligence and self evaluation as to how well YOU are operating and how well your instincts are validated by data. Not only should this be applied to the individual, but to the organization overall and it's best if you can explicitly think about fine tuning and improving your collective understanding about your customers.

People matter

As anyone who has managed a team knows, the people you hire play a huge role in the success of your team. Building a team to work effectively with data informed design requires hiring a certain kind of profile. There are three qualities that I look for when building out the design team in a data informed organization: 1) fundamental design skills, 2) business acumen, 3) affinity for data informed design. You don't always need to hire people who have actually had experience in doing data informed design, but you at least need to find people that seem to have an affinity for working within that framework. If the folks that you hire aren't fundamentally open to the concept of data driven design then it can make it very difficult for them to be successful in the team. This of course doesn't mean that they are "bad" designers, it just means that they would probably be more effective (and more appreciated by their peers) in a different environment. As we mention throughout the book, there are many ways to do design and to build product. What fundamentally determines the success of design or product is finding the right fit between the individuals, organization and the processes that you use to get there.

I've found that the following are a handful of qualities that also seem to correlate to the kind of profile that works well in a data informed organization:

- Curiosity about the business - Ideally the people you hire have a passion and curiosity about the business you're in, but I've found that within a data-informed environment, it becomes even more important for people to have a certain level of business acumen and ability to think strategically about the business. This is because it all ties back to being able to create hypotheses that have the intent of moving the metrics you use to measure your business by. If the designers and product managers who are tasked with building the business solutions don't have a baseline level of business capability, then they will be handicapped in terms of coming up with the right solutions.

- Ego - Let's be honest, designers are often well known for having egos. There are many kinds of designers that exist in companies and all types of designers can be successful in their careers. For some designers, a healthy ego can actually be a necessary and positive characteristic as it can sometimes give designers the confidence and ability to convince others of their vision and to sell them on that. One of my art history professors once pointed out that without ego, no artists

would survive because they have to undergo so much rejection in their career. Ego is a necessary component for creativity to survive. In a data informed design environment, ego can get in the way of success. This is because as part of a data informed organization you often have to be willing to let the data make the decision instead of the individual. I have seen many designers struggle with this and especially when they are coming straight out of design school can be a hard thing to accept.

- Humble - Because your ideas and designs are always being held up to examination and "proof" in the real world, then it's important to be comfortable with being humbled. There will be many times when you find that the design or thing that you believe in doesn't actually prove successful and you need to be able to accept that and then move on.
- Scientific mind/math skills- it is true that having some basic understanding of statistics and analysis can make a big difference in whether or not an individual can succeed in a data-driven environment. You don't need to be great at it…the lowest grades I ever got in college and graduate school were in statistics related classes (statistical thermodynamics - C-) . However, it's important to have some basic level of math is just so that you can have those conversations with the folks that are doing the analysis of your work. You need to be able to question how the success of your work is being measured and you need to have an appreciation for it. Without some fundamental understanding of the theory behind many of the techniques in a data informed environment, you won't be able to engage in the more strategic conversations.
- Focus – Especially in the AB testing part of data informed design, it can be easy to get distracted with different options and test cells and variations. By having a certain amount of focus and discipline, you will be better equipped to resist the urge to fall into the various pitfalls of AB testing around making a decision too early or building out too many test cells.

Practicalities: Best practices in setting up a culture for data informed design

Now that we've talked a little about what a data informed culture might look like, we wanted to make sure that you also had a few tactical best practices that you might consider putting into place and adopting for yourself. The key areas we'll cover in this section are:

- Getting everyone on the same page
- Internal and external examples to help establish the practice
- Finding a pilot group or allies within your company
- Ongoing education – what works and doesn't

This section is really written for people who don't currently have a well established process of data informed design and are looking to set it up for the first time. You'll find a lot of tips for "the first time you do X" in here.

Getting everyone on the same page

It's probably obvious, but worth mentioning that before you embark on setting up a data informed design process at your company, you'll actually want to make sure that the people who are going to be affected by it really do understand what's involved AND that they want it. In my experience, if a process isn't already established in a company there will always be some level of skepticism around something new or something that wasn't developed organically within the company. You might find that there are people who will say "Yes! We should definitely start to take a data informed approach to design, but we shouldn't do X, Y and Z" or that what they think it means to be data informed, might be something very specific (like only AB testing) and that they aren't considering the bigger picture.

A good practice is to get a lay of the land. Find out definitively what people really do know and don't know about taking a data informed approach to design. You might find that people really understand user research, but have much less experience with AB testing (or vice versa – this was the case at Spotify). Find out what the biases and assumptions are about data informed design and do this at all levels of the organization (from CEO to individual designer or product manager). Having all of this understanding up front about what people think and know can really save you a lot of time in the long run. You'll be able to assess where you might want to focus initially in terms of establishing best practices and you'll also become aware of any challenges you might face as you try to get a data informed design process established. I've definitely been in a situation where I've jumped to the conclusion that someone had the same level of knowledge or understanding of data informed design as I did, only to find that we later had to back track and redefine some of our fundamental assumptions together. If we had just taken the time up front to establish what our relative approach and philosophy was, we could have saved hours of mis-interpretation and backtracking.

Once you've established what your baseline is, you might then want to explicitly talk about what practices you want to establish and why. Getting your team and others that you work with to agree to try out some of these things that might be new to them will be important if they're not familiar with some of these tactics. It's always good to set expectations for what you will do and what you expect to see as a result, so saying "I'd like to take an extra few days on this to make sure we have time for user feedback on how usable this flow is." Makes it clear that the cost will be a little more time to introduce something new and what level of results you hope to get from it. I've definitely seen situations where teams were hoping to learn something from AB testing that actually would have been much better suited for usability research or vice versa. They then leave that situation feeling like they didn't get what they were hoping for and are much more reticent to try out that technique again in the future.

Internal and External Examples

One of the main reasons people consider adopting a data informed design approach is that it can save you a lot of time in learning something about your product before you release

it. I've found that you can learn a lot from past internal examples as well as by looking externally at what other companies are sharing about their learnings.

As data informed design is becoming an established process, I've found that an effective technique is to take past projects and then retroactively look at the data using a data informed framework. Of course this may not always be possible, because you may not have had all the proper tracking in place but many times the basic metrics will be tracked and available. You can look at old projects and see if there was any way of looking at the data ahead of time that might have affected the decisions you made on the project. Often the data is available to but the team just wasn't considering it.

This might be especially effective on projects where the outcome wasn't what you were expecting. For example, was there a project where you were hoping to see a big lift in acquisition and it didn't happen? Going back to the data, was there anything you could have looked at prior to making those changes that might have given you a hint as to the fact that it wasn't going to be successful? How can you build that knowledge into future tests? Or perhaps your users reacted very badly to a new feature you launched but really thought was going to make them happy… applying some survey techniques or user research in these instances (even though the feature has already been launched) will give you insight into what your users might have been able to tell you ahead of time if you had done this work pre-launch. When you can share these kind of insights back to the team (even if it does feel "too late") it makes people that much more enthusiastic to incoroporate a data informed approach earlier in the process the next time.

There are also a number of external resources as well where you can see what other companies have learned. People are constantly sharing information about what experiments they've tried and what they've learned from them. As you are considering different design and product decisions, looking to what may already be written or shared on the subject can be a great way to pre-inform or to accelerate acceptance for adopting a data informed design process in your company. For example, you might be considering

creating a side navigation system, but you find this article from the Next Web on side drawer navigation costs user engagement. [2] At the time, this was a very common navigational style and many companies were doing it. Sharing this data back to your team should give you good reason to consider running a test for your own navigational system and evidence that for this particular decision taking a data informed approach to the design would be valuable.

One thing to keep in mind is that the results that you see for some companies may not always be transferrable to your company. There might be other factors at play which give them the results that they get and so you can expect that blindly applying the same approach will give you the same results. This is why it is important to eventually establish your own internal process of data informed design – so that you can ultimately

[2] http://thenextweb.com/dd/2014/04/08/ux-designers-side-drawer-navigation-costing-half-user-engagement/

build up your own knowledge base of things that work best for your company and your product.

Pilot groups and allies

Whenever I'm looking to try something new at a company, I always try to start small by using a "pilot group". If your company or organization has never done data informed design before, you're likely to find that there are some folks who are eager to try something new and others who will have more resistance to it or not want to be bothered with the hassle of adopting a new practice. Generally, finding a few allies who are willing to try something out with you and who are also going to be more forgiving in the initial stages when you are likely to make a lot of mistakes can be really helpful. This will give you a chance to work out any kinks in the process with a small group of supportive people and then have more confidence as you try to roll it out to other designers or teams.

To find a good "pilot" project, there are a couple of characteristics to consider. Find something small so that you can ideally get a "quick win" and share the positive results of using data informed design to others. If the first project you pick to use data informed design takes you a few months just to establish what data you're looking for or to do the analysis on the results, it's very unlikely that you'll get a lot of enthusiasm from others to adopt something new. Find a project which has well defined success metrics. This will make it easier for you to explain what you've learned or not by applying a data informed process. A project that allows you to do a couple of quick iterations can also be good because it can demonstrate a data informed process over a couple of cycles. In the rest of this book, you'll find that we really try to encourage using data to do more than "just optimization" but when you're establishing the practice for the first time – these kinds of projects are perfect for warming up the larger organization to the concept of data informed design and to build confidence in adopting a new framework for design.

Finally a note on "allies". As mentioned above, some people will be naturally drawn to a data informed approach and others will not. I've often found that the best allies exist in other teams and that having many different voices who can advocate for a data informed process can be very powerful. Some of the best advocates I've found have been in the analytics team (if you have one) and marketing. Seeing support from the broader organization helps to make this more than "just a design thing" and helps to elevate it to the level of a company wide initiative. This also helps with the "universally accepted" aspect of culture which we opened this chapter with as well.

Ongoing education and establishing a rhythm around data

As we've mentioned before, creating an environment where it's clear that ongoing learning is supported can be one of the most effective tools to getting traction for a framework like data informed design. I've found that this is best done through a series of ongoing meetings, forums and occasional talks.

Lectures – this might be more applicable to larger teams - introducing a topic to a broad audience (say a mix of tech, product and design) can be an effective way to plant some of the seeds of data informed design with that group. In the past, the kinds of topics that I've used these lectures for have been:

- Introduction of company wide metrics
- Overview of best practices and introduction to data informed design processes
- Overview of different methods and techniques and when to use them

However, lectures can't be the ONLY means you use to introduce a new framework to an organization. You'll notice that most of the above are at a very superficial overview level. These can be a great introduction, but what's really needed are follow up discussions with active project teams at the right time in their process. So, while it's good to use an all hands or team meeting to talk about the ways that you might leverage early stage user research – it won't really feel concrete or useful until you can find a team that is actually at a stage in their project where they can leverage early stage research and speak directly to what techniques make sense for them at that point in time. This is one of the reasons that we'd really like you to look to the later chapters as something that you may not read in sequence, but instead can refer back to when appropriate for you.

Project review meetings

One of the most effective meetings for sharing knowledge is a recurring project review meeting. This might be an ongoing meeting between a broad set of product managers and designers to review a number of topics around ongoing work. The key is when you invite people that may not normally work day to day with each other – this way they can share what they are learning in each of their areas to the broader organization and it helps generally with building up that shared knowledge base of what is working and what doesn't work when it comes to your users. You might break this meeting into two sections:

1) Ongoing or upcoming work
2) Results from launched projects

The meeting should encourage a lot of discussion and debate about the projects that are being shared and it should feel more like a working meeting where everyone is vetting what is being presented. The people who presented should leave the room feeling more confident about the approach that they are taking and perhaps have some new insights as well. To get you started here are some of the kinds of things you might present and discuss.

On ongoing or upcoming projects your discussion might include the following:

- Hypothesis
    - Is the hypothesis valid?
    - Background for what generated the hypothesis and any prior work/data/research that influenced it

- Success metrics
    - Are they right ones?
    - Will these really measure the validity of the hypothesis?
    - What other things might you measure or use to measure success?
- Methodology
    - What techniques are you using to collect data for this project? (e.g. user research, AB testing, etc.)
    - What do you hope to learn?
- Design
    - How effective is the design at reflecting the hypothesis or variations of that hypothesis?
    - How does the design support what you want to learn from this project?
    - Note: This meeting will be different from a design review where you might be looking at consistency within your design language, etc. and giving more pure design feedback. Those meetings are still useful, but probably a separate meeting from this one.

For projects that have launched and where you are reviewing results, discussion might revolve around the following:

- Summary of methodology
    - What methods were used?
    - How effective were they at getting the insights you were looking for?
- Results and analysis
    - Was the hypothesis proven? Why or why not?
    - What did the team learn and what can be applied to other work that is going on ?
    - Did the results support any other larger trends that you might have seen before?
    - How do these results compare to prior work?
    - What are the next steps?

By reviewing work like this on a regular basis, people can actively learn about what works and doesn't work. I've found that it's always been more engaging and useful for folks to learn through case studies and having a meeting like this can encourage people to ask questions and discuss, , thereby learning together and actively learning while doing.

When the organization is newer to data informed design, having a meeting like this is a great way for the few who might be more comfortable with the concept to teach many. When the organization is actually fairly mature at using data informed methodologies, then I find that the discussion can actually be quite stimulating and really great at helping to push, uplevel and engage people even further.

Keeping track of track records

Another best practice which is especially effective with respect to AB testing is to keep track of which cells you (and your team) think will "win" and then comparing it to what actually happens. To cast your "vote" ahead of launching a test, ask all the members of the team to write down what they think the results will be and why they think those results will happen. When you get the actual results and can compare it with the votes that were cast prior to the test being launched then it gives you a chance to see how well you and your team were actually able to predict the impact to users.

Over time, it's always interesting to keep track of who in the team has the "best" product instincts. There's of course a danger in this tactic in that it can create a competitive environment, but the trick is to make it more about learning together and keeping each other honest. This ties back to the section in this chapter where we talked about self evaluation and holding yourself accountable. I've actually done presentations where we've used survey results or results from AB test to ask the audience to raise their hand or predict which version was going to win or what consumers might do. It's always enlightening for people to see just how poor some of their instincts may have been.

Developing a rhythm around data

Finally you might find that there are some very easy ways to set up a rhythm around data. In this sidebar, Leisa Reichelt talks about how getting executive buy in to come and watch 4 hours of user research every 6 weeks became a great way of establishing the value of user research to the rest of the organization. There may be other things that you can look to piggy back on to set up an expectation within your team about good habits that you can make part of a recurring rhythm.

For example rather than setting up user research as needed, you could set up a monthly user research session which either focuses on your baseline experience or that gets filled with whatever happens to be going on that week. You might also consider sending out a monthly update with insights from the various projects that you've been doing. By starting to set up expectations that data will be collected on a regular basis or looking to create dashboards that show how metrics might be changing over time given changes that you are making to the product, you can start to establish a working rhythm in your organization around data.

Summary

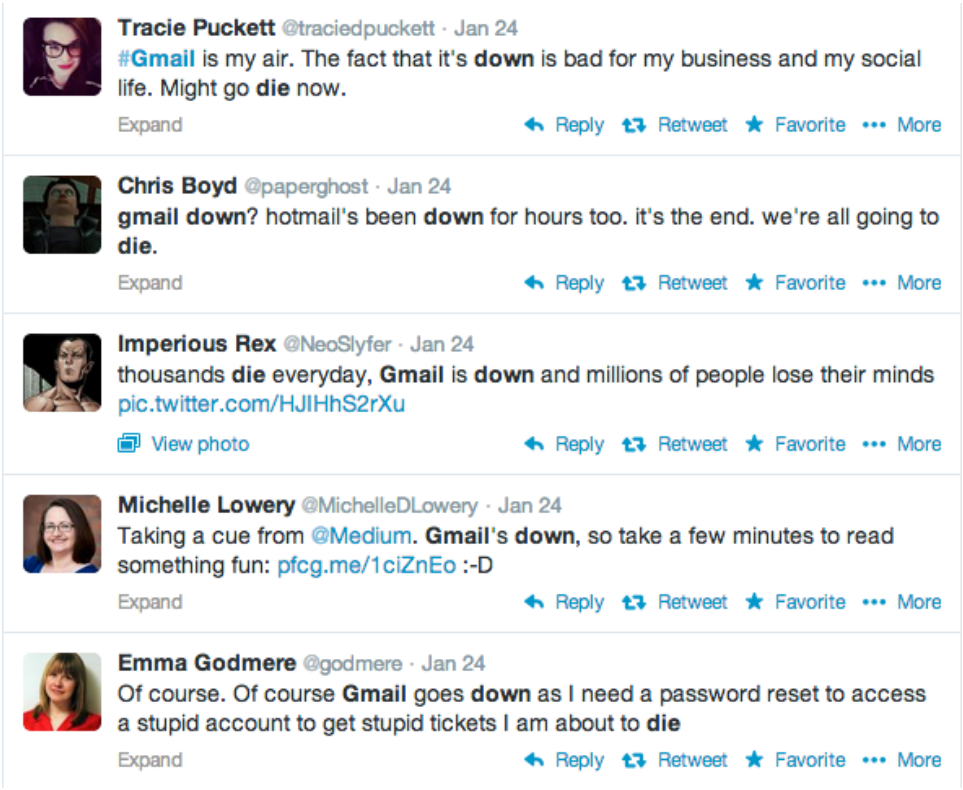
If you're interested enough in creating a lasting and deep culture around using a data driven framework to make product decisions, you'll also invest some time in building up a culture which can support it with the right processes as well as the right people. In the following chapters, as you learn more about data informed design, look for the concepts and ideas that resonate most with you and ask yourself some of the following questions along the way:

- What are the things that will be challenges for you as an individual to change?
- What are some of the things which will be challenging for your organization or team?
- What existing habits and cultural aspects are already in place that might make a good foundation upon which to build a culture of data informed design and decision making?
- Who do you already have in your organization that might prove to be really helpful for you to pair with as you try to establish these practices?

As we mentioned when we started this chapter, culture is really something that is unique to every organization and we would never dictate what the right solution might be for you. You may want to take some of the practical things we've suggested and put them in place right away, and you'll find that some of them will work well and last, whereas others will just never get any traction.

Perhaps in keeping with the theme of data informed design, I've always taken an experimental approach to organizational processes as well – trying out many different things, and then getting the feedback from my "users" (or team) which things seem to be effective or not. And again, similar to how you approach data driven design, see if you can find ways to measure your success. In the case of culture, it may not be as clear cut as "retention" or "acquisition", but you can find other ways to measure your success. How well has your "common vocabulary" taken hold? Are people actively asking to bring data into the design and product development cycle or do you need to push them to incorporate it? How much does data informed design get attributed for the success of your product and how much is it referred to by other organizations in your company?

As with other advice you'll see in this book around AB testing, it's important to have a certain amount of patience as well to see if the fruits of your labor are paying off or not. In my experience it can take anywhere from 6 months to 1 year to really see a change in the mindset of an organization take effect. Keep looking for those cultural success metrics to take hold, hopefully with some time and tweaking, you'll find that data informed design really does have a positive effect not just on how you build your products, but in your culture as well.



Why do people create necklaces and earrings of an error page for Twitter[8], but talk about death and suicide when gmail goes down?

It's pretty simple – they're two totally different kinds of businesses. Twitter is a user generated content site. When UGC sites go down, people's livelihood generally don't depend on it (especially for those in early stages), so you normally get a minority of people that have legitimate grievances with downtime.

SaaS products like gmail are products people rely on. When it goes down, it means someone can't so something that they've generally paid (or are getting paid) to do.

The moral here isn't that it sucks to work at Gmail and rocks to work at Twitter. The moral is that SaaS companies care about things like performance and user experience. Social networks care about completely different things, like sharing, return visits, or reducing spam content.

The kind of business you're in dictates the kinds of metrics you care about. We'll cover in detail the 7 kinds of businesses and the 3 stages of a company in Chapter 2+ so that you can focus on getting the right kind of data for your situation.

[8] https://www.etsy.com/ca/listing/60239227/hoist-fail-whale-necklace

THE **LEAN** SERIES ERIC RIES, SERIES EDITOR

Laura Busche

LEAN BRANDING

Creating Dynamic Brands to Generate Conversion

O'REILLY®

Lean Branding

Creating Dynamic Brands to Generate Conversion

Laura Busche

O'REILLY®

Beijing • Cambridge • Farnham • Köln • Sebastopol • Tokyo

Contents

PART FOUR: LEARN

CHAPTER 3

Brand Story

Throughout this chapter, we will look at the six essential parts of a lean brand story: positioning, promise, personas, personality, product, and pricing.

I've never met someone who did not aspire to be something more. Even Homer Simpson, with his absolute lack of will and ultimate disregard for the future, at times aspired to be a better husband, a fitter couch potato, a less miserable son to his old man. I've heard 5-year-olds state with absolute certainty that they will be president. Someone is sweating his head off right now at some ridiculously expensive gym because he aspires to be fitter. As you read this, someone is pulling an all-nighter studying to earn class valedictorian status—or maybe even reading this book to up her brand game (in which case I highly appreciate it!). I'm pretty sure someone is daydreaming to the sound of Bruno Mars's "Billionaire." Someone (perhaps you) aspires to build a successful product that customers open their wallets and hearts for. If that is indeed you, please hold on to this thought: *aspirations*.

Behind every great brand is a promise that fulfills its customers' aspirations. We are in the business of taking customers from A to B, where A is who they are today and B is who they want to be tomorrow. Consider the products you love—sports attire, note-taking apps, electronic devices, chocolate ice cream…this book—and how they've made you feel closer to whom you want to be.

When most people think about aspirations, they imagine long-term dreams or perpetual objectives. But in reality, aspirations come in all sizes, time

lengths, and levels of difficulty. After all, an aspiration is nothing more than a pursuit—an urge that influences our daily decisions. Whether that aim is to become a more organized worker, a more inspired creative, or the president of the United States is irrelevant.

> *All human aspirations are opportunities for brands to build relationships.*

Here are some sample aspirations to think about:

- Be *independent* and perceived as such by others. Become an autonomous individual by purchasing products and services that empower you to do more and better on your own.
- Be more *relaxed*. Live a less stressful lifestyle by purchasing products and services that help release different kinds of tension.
- Be *unique* and perceived as such by others. Express yourself and your worldview by purchasing products and services that let others know who you are and reflect your identity to the world.
- Enact a *new role* in life. Embody a "new persona" by purchasing products and services that help you attain a new professional or personal position.
- Engage in *better relationships*. Improve the way you connect with other human beings by purchasing products and services that strengthen your social circle and increase your sense of belonging.
- Be *more stable*. Avoid danger by purchasing products and services that increase your safety.
- Be *well-known*. Become accomplished by purchasing products and services that help you become more recognized and reputable.
- Be a *genuinely better* human being. Grow individually by purchasing products and services that boost your professional, spiritual, and emotional development.

Forget Everything You've Heard About Brand Stories

Our mission for this chapter is to understand whom we're selling to and learn the best way to show them who we are, what we offer, how we're different, and how we promise to help them. The good news is that there's no need to reinvent the wheel. Human beings have been learning via stories forever, and there's simply no better tool to send our brand message across.

For some reason, when they hear about "brand stories," some people imagine their company's CEO reading to a potential buyer sitting on his lap, in an altruist display of emotion and mutual love. *Rewind*. Yes, there might be some reading. And yes, we need the CEO on board. But, while emotion is a part of the whole thing, please remember this:

> *Your brand story's "happily ever after" involves open wallets.*

Seth Godin has an interesting thought on the matter:

> *Stories and irrational impulses are what change behavior. Not facts or bullet points.*

Some other people hear me say "brand story," and "storytelling" immediately comes to mind. I'd like to let you know that "telling" is not what we're doing here. We're aiming for "story-showing." Telling isn't enough. Brand communications are not about explain-time, justify-time, defend-time… blah-time. Please keep the idea of *show*time close to heart. This is why Chapter 5 is all about "story-showing"—feel free to take a peek if you're curious at this point.

But before we deal with ways to show our brand story, we need to create one.

I know social media, digital channels, videos, websites, and many other tools are appealing at the moment. I know that you think they might help your product scale instantaneously (miraculously); that they're just what the doctor ordered. *I also need you to understand that you better stay away from them unless you have a story to tell.*

Otherwise, digital channels just have a way of…well…swallowing you.

Remember this:

> *Digital channels are just tools to show (what should be) a meaningful brand story.*

Having a brand story is not optional. If you don't start writing it, please worry because—mark my words—*someone else will write it for you*. In my experience, only extremely disgusted or extremely pleased customers will put any effort into contributing to your brand story, so imagine how *that* would turn out without your attention. Brand storytelling is your unique chance to be persuasive and make the case for your product.

To take full advantage of this opportunity, we will answer five simple questions for our customers—though, as you'll see, there's really one big, profitable, underlying question (*Why should I buy from you?*):

Brand story part	Customer asks	Customer really thinks
Positioning	"How are you useful to me?"	***WHY** should I buy from you?*
Promise	"What do you promise to do for me?"	*Why **SHOULD** I buy from you?*
Personas	"What do I need/want from you?"	*Why should **I** buy from you?*
Personality	"Who are you?"	*Why should I buy from **YOU**?*
Product	"What will you offer, over time?"	*Why should I **BUY** from you?*
Pricing	"How much is this going to cost me?"	*Why should I **BUY** from you?*

First Things First: What's Your Name Again?

Someone has to come up with a name for the "fear of naming a new product." I am positive that this is an *actual* phobia. Side effects include headache, insomnia, anxiety, and utter bipolarity. I also blame this entire situation on superstitious nonsense.

Listen: I'd be a terrible marketer if I told you that your name doesn't matter. It does, deeply. What I have an issue with is the amount of time some people waste naming something that *isn't really anything yet*. Somehow, we've come to believe that even before the product has some sort of clear function (i.e., adds value), we will reach nirvana and come up with a *million-dollar name*. Said "million-dollar name" will then take a mediocre product from zero to millions within a day.

Wake-up call: *there is no million-dollar name*. There are, if you work hard enough, great *products* that make millions in the marketplace aided by an equally great name. Bottom line:

> *A great product deserves an equally great name that does it justice.*

That being said, there *are* some ways in which a fitting name can make the difference for your product. If you already have a name, feel free to jump to Part Three, where you'll learn how to measure its impact. If you are just getting started, you'll find the process much easier if you answer these questions (and please do):

1. Look around. What names are your competitors using to brand their products? (Write at least 20.)
2. What word or words encompass the *most important* thing your product is here to change? (Jot down a list of at least 20.)

Go literal: write down words commonly used in the industry that you're in, the type of product you're trying to sell, and the people who will buy it.

Sometimes we only think of nouns (i.e., things), but there's actually a vast list of verbs (actions) and adjectives (descriptors) that are great sources for names:

- Dropbox (verb + noun)
- Pinkberry (adjective + noun)
- Goodreads (adjective + noun)
- Instagram (adjective + noun)
- MySpace (adjective + noun)

Go figurative: use metaphors; find ideas that are related to the experience that you are creating.

3. Which of these words (or combination of words) could best convey what your product does? (Narrow it down to 10.)
4. Which of these options is most original and therefore recognizable in the marketplace?
5. Which of these options can be trademarked and does not interfere with another company's legal rights?

Gather Your Brand Ingredients

Remember the recipe we just learned about a few pages ago? That Lean Branding recipe included the 25 ingredients that we will be building throughout Part Two, and the time has come to find our brand story ingredients: *positioning, promise, personas, personality, product, and pricing.*

Positioning: How Are You Useful to Me?

We've already discussed aspirations: those possible selves that consumers are constantly striving for. We saw how brands can provide shortcuts to satisfy these aspirations and build relationships that go beyond a simple feature set. Our brand *adds value* when it helps a customer go from A to B, where B is whom they want to be. But just how do we go about making them understand and remember what we're here to do?

That's where positioning comes around. I know it sounds like a complex term, but it really isn't. If we are going to enter the marketplace to help customers go from A to B in their lives, we need to *position* our brand as an aspiration enabler. As a problem solver. As a propeller. Positioning is simply finding and taking a space within the marketplace that projects your brand as the "aspiration enabler" for a certain customer segment.

BRANDING ON STEROIDS

Naming Resources

Where to find name inspiration:

- *http://www.leandomainsearch.com*
- *http://www.panabee.com*
- *https://www.namefind.com*
- *http://www.namestation.com*
- *http://www.namemesh.com/*
- *http://www.domai.nr*

Where to figure out brand name availability (US trademark):

- United States: *http://tmsearch.uspto.gov*
- European Union: *https://oami.europa.eu/eSearch*

Where to figure out domain name availability:

- *http://www.godaddy.com*
- *http://www.namechk.com*

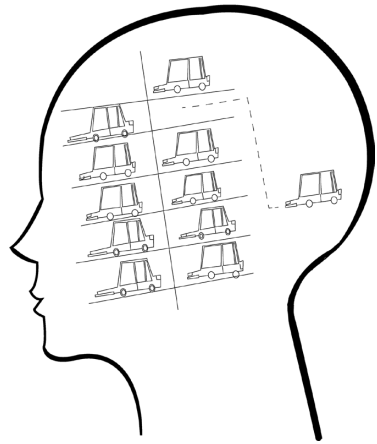
In a nutshell:

> *Positioning is finding the right parking space inside the consumer's mind and going for it before someone else takes it.*

These questions are key to building your brand's initial positioning:*

1. Who is your target customer?
2. Which of your customer's aspirations will your brand propel?

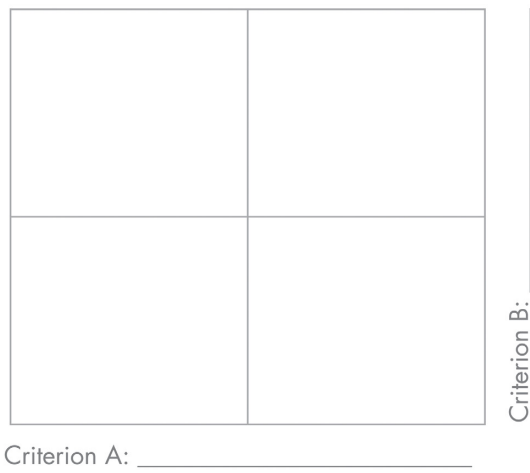
* It is important to make this distinction because, as you've learned before, customers cocreate this positioning with you once your brand is out in the marketplace.



3. What will make your customer be most compelled to buy from you?
4. Who is your main competitor?
5. What makes you different from this competitor?

As a first step, consider mapping out who your main competitors are and how your brand's offer compares to theirs. There is a very handy, visual tool to simplify this comparison. It is called a *positioning map*, and it will help you understand which brands you are competing with and what makes your offer stand out.

POSITIONING MAP



To build this map, select two criteria that your main competing brands and your own can be ranked on. For example, a fashion brand could select price and apparel formality as its criteria. Then, assign one of the criteria to the

horizontal axis and the other one to the vertical axis. Finally, map out the brands in your competitive space according to how they rank on both axes.

It is important to understand that this initial positioning is an ideal space that will not match consumer perception perfectly. This positioning map will change over time, and it is your job to make sure that continuous measurement guides any direction changes. Chapter 7 presents strategies to measure whether your positioning is resonating well in the marketplace.

The positioning statement

A positioning statement answers three main questions:

- Which space are you trying to occupy?
- What is the main aspiration that you are trying to satisfy?
- Who else is there (competing with you)?

No need to make our lives any more complicated (this is Lean Branding, remember?). Geoffrey Moore Consulting created a very effective template to create your positioning statement that has been used by companies everywhere:

For *(target customer)*

Who *(statement of the need or opportunity)*

The *(product name)* is a *(product category)*

That *(key benefit, compelling reason to buy)*

Unlike *(primary competitive alternative)*

Our product *(statement of primary differentiation)*

Fill it out. Done. Use it, use it, and abuse it. You can't go wrong.

Promise: What Do You Promise to Do for Me?

Keep it short and sweet. This is basically a fun-sized version of your positioning statement, emphasizing on your core value offer. It's the *one* thing you promise that will keep customers coming back, crafted into a memorable, short phrase that they can remember easily. It's your "Save money, live better" (Walmart), your "Remember everything" (Evernote), your "Eat fresh" (Subway)…you get the point.

Bob Dorf* has labeled this promise "Bumper Sticker" and has some great advice about how to make it shine. Take a look at the following "Get on It" sidebar. If you still need more ideas for your brand promise, check out the "Inspiration Hack" sidebar on the next page.

GET ON IT

The Bumper-Sticker Brand Promise

"A bumper sticker can't be more than a handful of words. Imagine a 12pt bumper sticker. Nobody in the car behind you could read the words on that bumper sticker unless they crashed into you! It's got to be a few words that summarize hopefully in a catchy, memorable, inviting way what it is that this company does that's special. It's really your brand promise all rolled into one mouthful of words."

—Bob Dorf

Personas: What Do I Need or Want from You?

Every brand story needs characters, and that's exactly what buyer personas are. We are going to create fake people with *very real* needs and aspirations to inspire everything our brand is, does, and communicates. This entire book is hanging on the idea that you are aware of who these characters are and are willing to constantly monitor how they act. So stay with me on these simple ways to find out who your buyer personas are and what they want. It's our only shot at making sure that what they want is *you*.

Most entrepreneurs will tell you that they "absolutely know the buyer." There are a few harsh truths that must be told at this point:

- Even if you think you know what buyers want or need, this is constantly *changing*.
- Even if buyers think they know what they want or need, sometimes they won't *say* it.
- Sometimes buyers have *no idea* what they want or need, but you can find out by observing.

---

* Bob coauthored *The Startup Owner's Manual* (K & S Ranch) with Steve Blank. A serial entrepreneur, he has founded seven startups for two home runs, two basehits, and three tax losses. He has invested in or advised more than a score of startups and teaches customer development at Columbia Business School.

INSPIRATION HACK

Brand Promises

Get inspired with these 10 brand promises that have endured time and our short (and getting shorter) attention spans:

- Office Depot: Taking Care of Business
- MGM: Means Great Movies
- MasterCard: There are some things that money can't buy. For everything else there's MasterCard.
- Airbnb: Find a Place to Stay
- Foursquare: Find Great Places Near You
- Las Vegas: What happens here, stays here.
- Disneyland: The Happiest Place on Earth
- Crest: Healthy, Beautiful Smiles for Life
- CrazyEgg: Visualize Your Visitors
- Zoho: Work. Online.

Go out and learn cautiously from what your buyers *say*, and rigorously based on what they *do*. This "going out there" is something anthropologists, designers, and professionals from many other disciplines know as *ethnography*.

NOTE

Marketers have been using focus groups and surveys for years, but besides not being time- or cost-efficient, they are often performed out of context. Meaning: is there a point in asking someone what she thinks by removing her from the place where she usually does her thinking and pretending she'll do *exactly* as she says? I won't bore you with method wars here, but bear in mind that we are looking for the most efficient way to learn about our buyers in order to make decisions that are not only informed, but also *timely*.

There are probably a million ways to conduct ethnographic research (i.e., going out there and observing people). For the purpose of remaining lean, let me introduce you to four that are very time/cost efficient.

Secondary research

Look out for data on your buyers from magazines, databases (if you have access to any), newspapers, industry white papers, and reputable sources that you can trust. This will give you a broad sense of how the market is doing and general trends to keep in mind as you observe your buyers.

In-depth interviews

Sit down with as many potential buyers as you can and ask them as many questions as you need to start discovering patterns among them. There are some very specific items we are looking for, and you will find a list of sample questions later in the chapter. Some people suggest you should also interview existing customers who both love and hate your product. In case you are just getting started, and you have no customers to conduct these with, arrange for interviews with potential customers. In Chapter 6, we will come back to this and apply it with existing customers.

Interviews were a vital part of the Apps.co entrepreneurship program in Colombia. Since teams were expected to go from idea to minimum viable product in eight weeks, the program demanded that a number of interviews be conducted during the first few weeks. These interviews would guide the development of the product and the brand that would project it into the world.

In an outstanding case of customer discovery, a mobile app called Bites pivoted from a food delivery intermediary into a tool to improve the on-site restaurant experience. After dozens of interviews with potential customers, the team realized that their market felt fascinated and moved by authentic food photography. They were eager to see pictures taken by other "foodies" and were more comfortable trusting their peers for restaurant recommendations. Perhaps repeating this exercise somewhere else on Earth would reveal different insights. And that, ladies and gentlemen, is the whole point behind conducting interviews while you are building your brand: going beyond your own assumptions.

Fly on the wall

I absolutely love this method. Picture a fly on the wall. That's pretty much what you'll be. This type of observation consists of becoming a part of any given environment for a few hours. You are trying to understand your buyers' context: what threats and opportunities can you spot? If you're interested in teachers, for example, visit a school for a given number of hours and record as many details as you can about the school day.

In 2012, my agency was hired to build a local politician's online brand presence. She had already been elected but was having trouble empathizing

with the citizens who hadn't voted for her. During a brainstorming session, we decided to use expressions from citizens themselves to communicate her brand message. We would address naysayers and continue appealing to sympathizers by using their *own* terms and language.

To do so, we followed the politician in every single public event for four months. We did not intervene or ask any questions during this time. Our team was just *present*, taking notes and analyzing what came out of citizens' mouths. Later, we would synthesize these insights and come up with messages that reflected the citizens' own concerns and satisfactions. Our social media engagement skyrocketed, and the conversation (which had remained stagnant for months) activated around the topics that citizens were actually interested in hearing.

Shadowing

Here's a classy term for some serious stalking activity—except you will formally approach someone to ask his permission to stick with him for a couple of hours. The idea here is to get a sense of what his particular routine is like, what are some needs and aspirations he might not be willing or able to declare verbally but are visible to you, and what channels would work best to reach him, among others. You must remain unnoticed and avoid disturbing his routine. This is *academic* stalking.

Take a few minutes to imagine the aspiration that your brand is here to facilitate. Are you trying to make it easier for customers with a fast-paced lifestyle to share their files? Are you trying to provide a healthy or environmentally friendly option in the midst of a toxic marketplace? Are you building an app to optimize the way I spend my time? Are you a political brand offering to bring prosperity and education? This answer is a vital starting point for the observations that you will take note of while shadowing a subject.

For Procter & Gamble, this answer was once "we are trying to offer a better way to clean a floor."* In 1994, the company hired an innovation and design consultancy called Continuum to figure out a way to reinvent the act of cleaning floors. To do so, researchers followed individual people as they cleaned their own kitchen floors. They found out, among others, that "mops worked mostly by the adhesion of dirt to the mop and people seemed to spend almost as much time rinsing their mop as they did cleaning the floor." To solve it, they designed a new product that would bring their newfound idea of "Fast Clean" to life. The rest is Swiffer history.

---

* *http://continuuminnovation.com/work/swiffer/*

As you follow your potential or existing customers, take note of the activities, environment, interactions, objects, and users involved in their routine. These five types of observations belong to a framework called AEIOU (after their initials) that was developed by a group of researchers at the Doblin Group in 1991 to make ethnography easier to record.*

Here is what some sample, raw shadowing observations might look like, as coded with the AEIOU framework. You can also use this framework to record your fly-on-the-wall observations:

Subject: John, potential user for our to-do list app **Started:** 03/05/2013 8AM **Ended:** 03/05/2013 11AM	
Type of observation	**Notes**
Activities	Getting ready for work Driving to work Picking up coffee before work Accomplishing daily tasks
Environment	Home: 2BR apartment, NW side of town. Office space in living room. Pinboard next to desk. Monthly calendar pad on top of desk. Coffee shop: dim lighting, slow jazz music. Office: personal cubicle, white light, notebooks scattered on desk. Monthly calendar on cubicle wall.
Interactions	Drinking coffee, talking to barista, printing a last-minute document, checking email on smartphone, driving compact car to metro station, parking car, taking metro to work. Checking email on desktop computer again. Adds some dates to wall calendar. Added sticky note to computer screen. Completing yesterday's pending tasks first. Tossing sticky note once task completed.
Objects	Monthly calendar pad on home desk, cappuccino with low-fat milk, Metro card, printer, 21.5" iMac computer, colored sticky notes, iOS smartphone, monthly calendar on cubicle wall.
Users	Wife, boss, barista

After following John for a couple of hours, you already have an interesting set of brand-related questions to continue validating:

- Should our brand message include a reference to "keeping all your to-dos in the same place" or "making your to-do list accessible"?

---

* Bruce Hanington and Bella Martin, *Universal Methods of Design: 100 Ways to Research Complex Problems, Develop Innovative Ideas, and Design Effective Solutions* (Beverly, MA: Rockport Publishers, 2012).

- Considering John's reliance on email, would placing ads on email platforms be an effective communication strategy for our brand?
- Would coffee and a desk generate associations with planning and to-do list creation, and therefore be effective images to build our brand's visual identity around?
- Given John's desire to visualize important deadlines in walls, should our brand message focus on screen-paper (online/offline) integration?

The importance of carrying a field guide

Some general guidelines are to make sure that you select the right people, prepare your interview questions beforehand (but be sensitive to open up to others), fill out a reasonably organized field guide, and keep your eye open for behaviors that deviate from the standard. Also, even if you're researching buyer personas for a very specific product, make sure that you look at the *whole* experience. You never know what new ideas may come up.

As promised, next is a sample field guide. Miscellaneous questions like "favorite social media platform," "car make," and "hair color" might make sense in specific cases (with car or shampoo brands, for example). Take this guide with you, be *efficiently* creative, and feel free to edit or add questions as needed.

SAMPLE FIELD GUIDE

NAME	
AGE	
MARITAL STATUS	
INCOME LEVEL	
LOCATION	
NO. OF CHILDREN	
TO DO LIST	*What must he/she accomplish every day, as related to the solution your brand is proposing?*
ASPIRATIONS	*What are his/her goals, as related to the solution your brand is proposing?*
PAINS	*What is it that he/she finds difficult to accomplish at the moment, as related to the solution your brand is proposing?*
CRITERIA	*What makes him/her complete the purchase of an existing solution, similar to the one your brand proposes? How much is he/she willing to pay?*
CHANNELS	*Which publications is he/she most likely to read? Where would it be smart for us to place our brand message?*

www.leanbranding.com

Now that you've interviewed and observed a considerable amount of potential buyers, summarize the patterns you have uncovered in—you guessed it—your first buyer personas. The main characters in your story. Also known as *the people we're going after.*

Use the following template to summarize your information:

Personality: Who Are You?

Ever been stuck talking (more like yelling) to a machine when you needed support from an actual human being? That's how customers feel when your brand personality isn't there or isn't coming through. Try this with me: imagine Apple and Microsoft as people. Now think about them going out on a date. If you can sustain their dialogue and picture them for over a minute, both of these brands have done their homework. When *they* tell you stories, you get "human" rather than "machine"—even though (sorry to break your heart) they are essentially machines.

> *People relate to people, and if your brand feels like "people," they'll relate to you, too.*

BRANDING ON STEROIDS

Social Media Research

Give it a try: use your prospective buyer's name and search for her problems using hashtags like #teacherproblems #ceoproblems #momproblems #teenageproblems #athleteproblems. What you'll find is pure gold. We used to *pay* for that kind of insight. Marketers used to (and still do) give people journals where they documented any given experience during their day. Forget journals: this is real-time, real-deal market research.

These are some of the tools that I use the most:

- Topsy
- Twitter Advanced Search
- Quora
- Google Plus
- Instagram
- Flickr

The American Psychological Association has defined personality as the "unique psychological qualities of an individual that influence a variety of characteristic behavior patterns (both overt and covert) across different situations and over time."* Our personality influences the way we think, behave, and feel with regard to everything that surrounds us. Similarly, brands must react in a marketplace where conditions are uncertain and ever-changing. Creating a flexible, yet strong, brand personality will help us adapt quickly to this environment. Lean brands sail out into the world to discover their customers and can rarely predict the outcome; however, they can always decide how to react to what they find. A brand personality guides us in this agile decision-making process.

For the purposes of this book, we will understand "brand personality" as the humane psychological qualities associated with our brand that dictate its interactions in the marketplace during different situations and over time. By linking your brand with several traits that are traditionally used

* Richard J. Gerrig and Philip G. Zimbardo, *Psychology and Life*, 16th edition (Boston: Allyn and Bacon, 2002).

to describe *human* personalities, you can build a more relatable story that consumers will engage with.

Having followed dozens of startups as they developed sound brand personalities, I have compiled a list of descriptors that can help you create yours. In Chapter 7, you will find out how to measure whether this personality is resonating well with customers and adapt it accordingly. Feel free to include and use other words that apply to your brand. Based on the positioning, promise, personas, and product you've described before, which of these qualities fit your brand best?

BRAND PERSONALITY PROFILE

Adventurous	Classy	Disciplined	Futuristic	Kind	Obstinate	Rebellious	Sociable
Affectionate	Clean	Discreet	Generous	Knowledgeable	Old-fashioned	Refined	Solemn
Agile	Clever	Disruptive	Gentle	Laid-back	Optimistic	Reliable	Sophisticated
Agreeable	Coherent	Dramatic	Grumpy	Liberal	Outgoing	Religious	Soulful
Alert	Compassionate	Eager	Handsome	Lively	Outspoken	Reserved	Stable
Altruistic	Competent	Easy-going	Happy	Local	Passionate	Resolute	Strong
Ambitious	Competitive	Eccentric	Hard-working	Logical	Paternal	Resourceful	Studious
Analytical	Confident	Efficient	Helpful	Loud	Patient	Respectful	Subtle
Argumentative	Conservative	Emotional	Hip	Loyal	Patriotic	Responsible	Systematic
Artistic	Consistent	Empathetic	Humble	Masculine	Peaceful	Restless	Tactful
Assertive	Controlling	Energetic	Idealistic	Maternal	Pensive	Rowdy	Talented
Astute	Cooperative	Enterprising	Impetuous	Mature	Picky	Safe	Thoughtful
Balanced	Courageous	Enthusiastic	Impulsive	Methodical	Playful	Sarcastic	Tidy
Brave	Crafty	Exuberant	Incisive	Meticulous	Polite	Sassy	Traditional
Calm	Crazy	Fashionable	Independent	Mischievous	Popular	Scientific	Trustworthy
Candid	Creative	Fearless	Indiscreet	Modern	Practical	Sensitive	Unassuming
Capable	Critical	Feminine	Ingenious	Modest	Precise	Serene	Unconventional
Careless	Curious	Fervent	Innocent	Motivated	Proactive	Serious	Urban
Caring	Deep	Fiery	Innovative	Mysterious	Proficient	Sexy	Versatile
Cautious	Defiant	Flashy	Insightful	Natural	Profound	Sharp	Warm-hearted
Charismatic	Delicate	Flirtatious	Inspiring	Naughty	Proud	Silly	Watchful
Charming	Determined	Formal	Intellectual	Neat	Provincial	Sincere	Wealthy
Chatty	Devoted	Frank	Interesting	Nostalgic	Prudent	Sloppy	Wise
Chic	Diligent	Friendly	Joyful	Nosy	Punctual	Smart	Witty
Child-like	Diplomatic	Funny	Keen	Nurturing	Reassuring	Snobby	Young

www.leanbranding.com

Good, so *what* are we supposed to do with this personality now? How does it shine through? How does it help us make more money? *Why* are we doing this? These are all great questions. Having a clear brand personality will make so many decisions easier that you will regret you hadn't defined one before. For example:

- Wouldn't it be easier to know whom to partner with once you know how your brand feels about life in general?
- Wouldn't it be that much simpler to choose a social media message once you know how your brand is supposed to sound?

Defining our brand personality gives us a better idea of how we should face the customer. It elucidates what is the *voice* telling the story. Keep this idea of brand voice close to heart; based on the personality we just described:

- What would my brand say and how? Along the same lines, how would my brand speak to customers during the different stages of their experience? More on this in the upcoming Product section.
- What does it hate?
- What does it absolutely love?
- What is its favorite drink or meal? Why?
- (We could go on forever, now that we have a personality.)

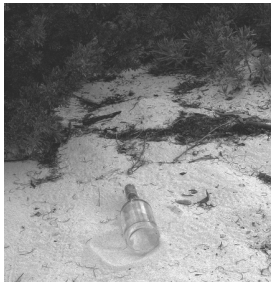
Consider the following example of a brand personality and how it translates into the brand's voice.

<table>
<tr><th>Brand personality</th><th colspan="2">Brand voice</th></tr>
<tr><td rowspan="4">Young, cool, unique, intelligent, confident, charming, funny, down-to-earth

Hates: arrogance, lack of originality, bad music

Loves: artists, designers, makers, good music

Favorite meal: chocolate anything

Favorite drink: happy-hour mojitos</td><td>To express encouragement</td><td>"Have a great Monday everyone! Three words about life: it goes on. (Also: Friday will come.) This is what XYZ HQ looks like this morning."</td></tr>
<tr><td>To express excitement</td><td>"We're thrilled to announce our newest feature: XYZ! Check it out and let us know what you think."</td></tr>
<tr><td>To express an apology</td><td>"Sorry, guys. We'll be back soon. Good things take time! In the meantime, here's a song that's keeping us going: XYZ."</td></tr>
<tr><td>To express gratitude</td><td>"Huge thanks to everyone who came out to our launch party yesterday! It's good to see that everyone survived. Well, almost everyone…"
  </td></tr>
</table>

Coming up next, you will see how this personality comes into play in different stages of interaction with your customer. Time to speak up!

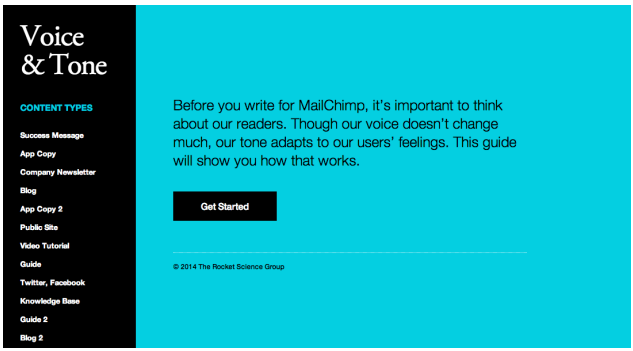
MailChimp's Freddie von Chimpenheimer IV

Meet Freddie von Chimpenheimer IV. You might recognize him as MailChimp's witty little character. Freddie is a clever chimp that becomes your concierge, coach, and friend as you navigate MailChimp's interface. More than serving, he is here to amuse you with unexpected jokes and short pep talks.

He doesn't sound like a vector cartoon anymore, right? That's because Freddie has a defined personality and voice that encapsulates the MailChimp brand story. Visit *www.voiceandtone.com*, and you'll find an interactive guide to what this story sounds like.

These are some of the jewels you'll find:

- "Freddie's jokes aren't intended to be useful or educational—they're simply a layer of humor. Be funny!"
- "Surprise and delight our users here. Catch them off guard in the best way possible."
- "Pat these users on the back for getting a campaign out the door. They're probably feeling happy and relieved—use casual language that encourages those feelings."
- "The app is MailChimp's heart, and tracking is usually the most exciting part for users. Our personality should come through in this step."



Product: What Will You Offer over Time?

Strong brands know that there's much more to it than "the sell." The term *product experience* does a better job at capturing what it is that our brand is really standing for. We're used to thinking that products (and services) are just a sum of features, but taking a closer look at the customer's experience reveals the truth: products are an augmented set of opportunities to add value that includes (of course) tangible features, but also service, visual identity, support, warranties, delivery, installation…the stuff people pay a premium for. The stuff strong brands are made of.

Products shouldn't just work well, they must unfold well.

Using Journey Maps to design your product experience

A useful way to visualize your product experience is to use a *Customer Journey Map*. These maps display the complete adventure that customers jump on the minute they consume your product. It is important to remember that consumption involves pre-purchase, purchase, and post-purchase stages, and consumers will be looking for you all along. A complete Journey Map includes these three stages. The main idea here is that customers are not just looking for value on the spot. These stages are all *brand touchpoints* in which you can add value, and it is essential to recognize what and where they are and how you'll respond to consumers during each of them. Think about a few stages that your own customer would go through as he consumes your product:

Before purchase

Maybe he has to check out a few other options before deciding whether to purchase your brand. What will you do during this stage to make sure that he learns about you? If this is a corporate purchase, does the buyer need someone else's approval? If it isn't, will he need his wife's approval?* Once he learns about you, how will you communicate what makes you different (and worthy of his decision)? (Chapter 5 will discuss communication strategies in depth.)

During purchase

The customer actually pays for your product. Is there something you can say or do to make this process easier or more rewarding? Is there a stimulus that you can provide to make the decision-making process simpler?

---

* These people standing between you and the buyer, affecting the buyer's decision, are known as *gatekeepers*.

After purchase

How does your product interact with the customer after he has purchased it? How does it resolve his initial aspirations? Perhaps your product involves an installation process. Even if it doesn't, consumers want to be able to communicate with you after their purchase and reaffirm that their choice was correct (read the upcoming "Dig Deeper" box to learn more). How will you make the customer feel that his purchase decision was right? If this is a corporate purchase, will other departments resent your customer's decision? May coworkers dislike the fact that your customer's purchase changed their workflow? What can you say or do to make sure that your product's integration with the customer's routine is seamless?

DIG DEEPER

A Word on Cognitive Dissonance and Why It Matters

Cognitive dissonance is a term widely used in consumer psychology to refer to what we commonly know as "having mixed feelings" about something. According to psychologist Leon Festinger, who coined the term in the 1950s, we are all looking for something called *internal consistency*. All this means is that we avoid displaying attitudes and opinions that create inconsistency within ourselves. Whenever our attitudes and opinions do conflict, we experience what is known as cognitive dissonance.

Now, this idea is essential in designing a pre- and post-purchase brand experience because consumers often experience periods of doubt, regret, and anxiety after they have acquired a product/service. As customers, we question whether the price reflected the product's quality, or whether the communicated feature set corresponds with the product's *actual* functionalities.

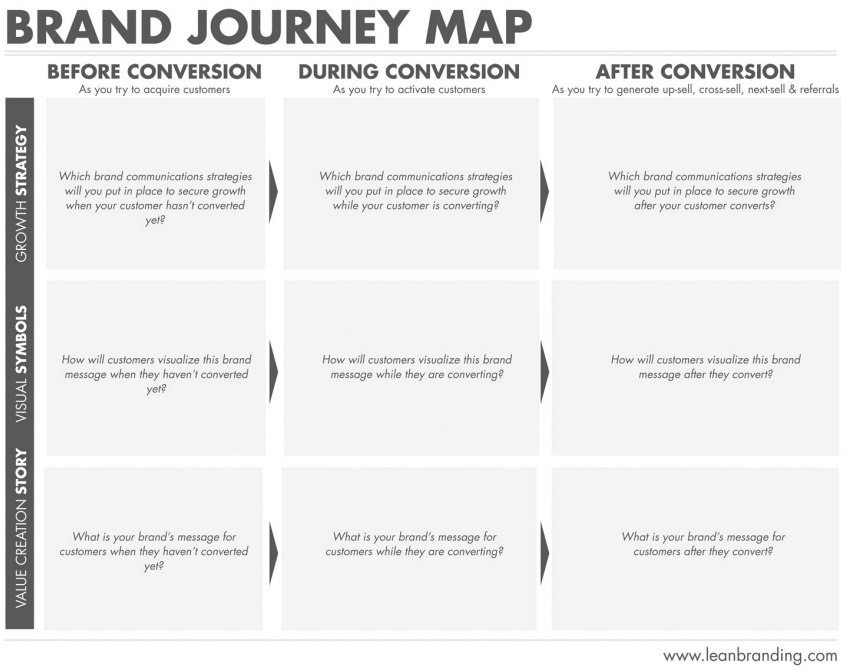
Cognitive dissonance can be dangerous, not only for your long-term relationship with a particular customer, but also for the potential ripple effect that may arise if the dissatisfaction grows. Think about aggressive messages in review platforms that may scare future customers away. Needless to say, brands need to work on preventing and remediating cognitive dissonance.

To prevent cognitive dissonance, answer the following questions:

- *Is your brand message honest from the pre-purchase stage?*
- *Are you sharing sufficient and objective information about your product's performance before the sale?*
- *Are you reassuring your customer after closing the sale (post-purchase stage)?*

Please note that "purchase" may not be a relevant conversion action for you. In this case, think about signups, registrations, votes, or whichever act *does* embody conversion in your business model.

Back in Chapter 1 we discussed the importance of understanding that brand and product don't compete. Brand *is* product, and everything else conforming to the unique story that consumers create when they think of you. Therefore, we can now think of your product experience in terms of all the brand elements that must come together when someone consumes your offer. To make this easier for you, the following Brand Journey Map template includes a series of stages and boxes to fill out as you design the experience that you are trying to offer.



Mapping the *brand journey* makes visualizing this process easier. I urge you to take a few minutes to draw your own path to identify where and how your brand story will be told.

In each step, try to picture what aspect of your brand experience is interacting with your customer. I understand this might be a little abstract, but check out the next "Inspiration Hack" sidebar to find a few examples of journey steps that would work for a web-based solution.

INSPIRATION HACK

Beginning Your Brand Journey

Here are some ideas for stages that you can include within your Brand Journey:

- Signup
- Calls to action
- Tutorial to help the customer get started (also known as *onboarding*)
- Referrals
- Typical product use
- Logging out
- Notifications
- Confirmations
- Receiving a periodic email
- Upgrading to a premium version
- Getting help

Pricing: How Much Is Your Solution Worth?

At this point, you've created a brand promise and positioning, defined buyer personas, and mapped an engaging product journey infused with an equally appealing brand personality. Good job! Now it's time to reap the rewards.

But just how will you go about figuring out what this reward should be? How will you fix a price that appeals to your buyers and keeps your company's finances healthy? When is it too much and when are you underestimating what your solution is worth to the buyer?

We could spend days discussing pricing strategies from multiple standpoints. There are at least a dozen different pricing models that you can consider putting in place, but we will look at three of the most widely used:

Cost-based pricing

> You set a price based on production cost considerations and profit objectives. Every company gauges these production costs differently, but an essential question remains: *How much should we charge for this product or service if it costs us X to produce and we want to earn Y on top of that?*

Value-based pricing

You set a price based on the customer's perception. During this pricing process, you basically figure out how much value your customer places on the product or service that your brand offers. *How much is she willing to pay for this product?*

Competition-based pricing

You set a price for your product or service based on your closest competitors' strategies. When you have little or no competition (in a disruptive category), you can look at substitute products. *How much is my closest competitor charging for a similar product or service? Do I want my price to be at, above, or below theirs?*

More often than not, a brand's pricing strategy is a combination of two or more of the models just presented. For example, a company like Samsung may concurrently consider production costs, its target customers' willingness to pay, and other manufacturers' strategies when deciding on a price for its products. Other widely used pricing strategies include:

Penetration pricing

Setting a low price to grab a large share of the market when you are launching.

Skimming or "creaming" price strategy

Setting up a high price initially to recover what you have invested in developing the product.

Freemium

Offering one version of your product or service for free and a more advanced version for a fee.

Premium pricing

Setting up a deliberately high price because it allows your brand to appeal to a specific market segment with a certain purchasing power.

Psychological pricing

Setting a price having considered how the actual numbers play a role in a consumer's decision-making process. For example, you may validate that charging $4.99 instead of $5.00 helps boost sales. This tactic relies on the idea that customers will (involuntarily) round down and perceive a lower price, which in turn triggers a purchase.

If you decide to use *value-based pricing*, keep in mind your potential buyer's answer to these questions from our Personas section (earlier in this chapter):

> *Purchase criteria and motivations: What makes him complete the purchase of an existing solution, similar to the one your brand proposes? How much is he willing to pay?*

In Chapter 7, when we measure our brand story ingredients, you will learn about a pricing concept called *willingness to pay*. This indicator will allow us to *approximate* our buyers' ideal price point(s) and approach our brand's pricing from a value-based perspective.

Bringing It All Together: Your Brand Storyboard

At this point, you have built a positioning, promise, personas, personality, product experience, and pricing for your brand. As we approach the next steps in this process, it helps to continuously visualize your entire brand story.

While I worked with tech startup teams from every background imaginable, I noticed that storytelling doesn't come easy for many of us. That's why I designed this simple tool:

BRAND STORYBOARD

Once upon a time...	He/she always...	But always had a problem...	He/she tried to solve it...
But he/she wished that...	**Until one day...**	**Unlike his/her solution, this...**	**His/her wish came true: to...**

www.leanbranding.com

By completing each of these scenes, you will be answering some of the key questions behind powerful brand storytelling. Let's go over some considerations as you fill out the storyboard:

Once upon a time

In this scene, you will describe the buyer personas introduced earlier in this chapter using images or text. Who is the main character in your brand story?

He/she always

Define some of the main tasks that your customer is regularly involved with. What does she do every day? What are her main responsibilities in life and work, as related to the product or service that you offer?

But always had a problem

State the main issue that your customer faces when trying to complete her tasks. What is the unsatisfied need or aspiration in this story?

He/she tried to solve it

If the previous problem is real, your customer is probably already solving it. What are some alternate solutions to the issue at hand? How is your customer managing to partially satisfy this aspiration?

But he/she wished that

Outline the flaws in the solutions your customer is currently using. Despite purchasing these other products or services, your customer is still unsatisfied. What are existing solutions lacking?

Until one day

Describe how your customer will most probably learn about your brand. What happened on the day she first heard about you?

Unlike his/her solution, this

List some of the aspects related to your product experience that set you apart from competitors. How does your offer differ from your customer's current solution?

His/her wish came true, to

Clearly define the aspiration that your brand fulfills. What is your customer's "wish come true"?

Now that you have filled out each of these scenes, consider some of the implications of each of your answers. *Why* do they matter? *How* can we use them? Why are they important for brand building and conversion? The following storyboard provides some thoughts:

ONCE UPON A TIME...	HE/SHE ALWAYS...	BUT ALWAYS HAD A PROBLEM...	HE/SHE TRIED TO SOLVE IT...
In this square we are defining the target audience for all our communications efforts. Advertising, content, and other types of campaigns should be designed and directed to this segment.	*These are the times and places where the need that your brand satisfies is most active. They are good spaces for brand placement and advertising.*	*Pains and aspirations that you must reflect in communications pieces, so that consumers feel identified with your message and it can resonate ("That's me!").*	*Learn more about these competitors and alternate solutions. Study their flaws and potential improvements. Consumers expect you to differentiate from them by bringing something new to the table.*
BUT HE/SHE WISHED THAT...	**UNTIL ONE DAY...**	**UNLIKE HIS/HER SOLUTION...**	**HIS/HER WISH CAME TRUE: TO...**
Explore ways in which you can solve these pending issues in your own product/service. Your competitor's flaws are areas of opportunity. You can address these existing consumer concerns to position your brand as the optimal solution.	*In this square you defined different scenarios where consumers first learned about your brand. Was it online? Through a video? By word-of-mouth? These are ideal channels to start sharing your brand story.*	*Reflect these key differentiators in your brand story, symbols and strategy. Make sure that your core value offer is contained in your brand promise.*	*Use this fulfilled aspiration in your communications pieces to reinforce your brand's role in the customer's satisfaction. Represent this aspiration in visual symbols like your brand's video and imagery as you see fit.*

Recap

As you'll remember from Chapter 1, we are in the business of taking customers from A to B, where A is their position today and B is the place they want to be tomorrow. Building a dynamic brand story helps drive this message home for the consumer. While digital communication channels are important, they are just *tools* to show (what should be) a meaningful brand story.

Many different ingredients are involved in creating your brand story. In this chapter, we introduced six of them: your positioning, promise, personas, personality, product experience, and pricing. A brand's positioning conveys which space you are trying to occupy, the main aspiration that you are trying to satisfy, and an idea of who else is competing for that space. Your brand promise is basically a fun-sized version of your positioning statement, emphasizing on your core value offer. It should answer the question, "What's in it for me?"

Personas are "fake" people with *very real* needs or aspirations that inspire everything our brand is, does, and communicates. By crafting our brand messages around these personas, our story will become more humane. This is important because people relate to people, and if your brand feels like "people," they'll relate to you, too.

Strong brand stories satisfy consumer aspirations with a well-designed, holistic product experience. Products shouldn't just work well, they must *unfold well*. Finally, defining our brand personality gives us a better idea of how we should face the customer. It elucidates the *voice* telling the story.

THE **LEAN** SERIES

Jeff Gothelf with Josh Seiden

LEAN UX

Applying Lean Principles to Improve User Experience

O'REILLY®

Eric Ries, Series Editor

Lean UX

Applying Lean Principles to Improve User Experience

Jeff Gothelf

Josh Seiden, editor

O'REILLY®

Beijing · Cambridge · Farnham · Köln · Sebastopol · Tokyo

Contents

SECTION III: MAKING IT WORK

CHAPTER 5

MVPs and Experiments

All life is an experiment. The more experiments you make, the better.

Ralph Waldo Emerson

With the parts of your hypothesis now defined, you're ready to determine which product ideas are valid and which ones you should discard. In this chapter, we discuss the Minimum Viable Product (MVP) and what it means in Lean UX. In addition, we'll cover:

- Determining product focus (delivering value or increasing learning?) using MVP
- Using prototypes and prototyping tools
- Running experiments without prototypes

About MVPs and Experiments

Lean UX makes heavy use of the notion of MVP. MVPs help test our assumptions—will this tactic achieve the desired outcome?—while minimizing the work we put into unproven ideas. The sooner we can find which features are worth investing in, the sooner we can focus our limited resources on the best solutions to our business problems. This concept is an important part of how Lean UX minimizes waste.

Your prioritized list of hypotheses has given you several paths to explore. To do this exploration, you are going to want to create the smallest thing you can to determine the validity of each of these hypothesis statements. That is your MVP. You will use your MVP to run experiments. The outcome of the experiments will tell you whether your hypothesis was correct and thus whether the direction you are exploring should be pursued, refined, or abandoned.

The Focus of an MVP

The phrase MVP has caused a lot of confusion in its short life. The problem is that it gets used in two different ways. Sometimes teams create an MVP primarily to learn something. They're not concerned with delivering value to the market; they're just trying to figure out what the market wants. In other cases, teams create a small version of a product or a feature because they want to start delivering value to the market as quickly as possible. In this second case, if you design and deploy the MVP correctly, you should also be able to learn from it, even if that's not the primary focus. See Figure 5-1.

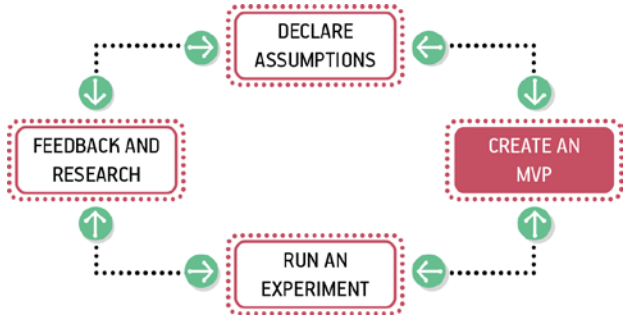


Figure 5-1. *You will create MVPs after you've defined and prioritized a set of hypotheses.*

Let's take as an example a medium-sized company with which I consulted recently. They were exploring new marketing tactics and wanted to launch a monthly newsletter. Newsletter creation is no small task. You need to prepare a content strategy, editorial calendar, layout and design, as well as an ongoing marketing strategy. You need writers and editors to work on it. All in all, it's a big expenditure for the company to undertake. The team decided to treat this newsletter idea as a hypothesis.

The first question they had to answer was whether there enough customer demand for a newsletter to justify the effort. The MVP they used to test the idea was a signup form on their current website. The signup form promoted the newsletter and asked for a customer's email address. This approach wouldn't deliver any value to the customer—yet. Instead, the focus was on

helping the team learn enough to make a good decision about whether to proceed.

They spent half a day designing and coding the form and were able to launch it that same afternoon. The team knew that their site received a significant amount of traffic each day: they would be able to learn very quickly if there was interest in their newsletter.

At this point, the team made no effort to design or build the actual newsletter. That would come later, after the team had gathered enough data to make a GO/NO-GO decision. After the team had gathered enough data, and if the data showed that their customers wanted the newsletter, the team would move on to their next MVP, one that would begin to deliver value and learning. They planned to experiment with MVP versions of the newsletter itself that would let them test content strategy, design, and other newsletter features.

Creating an MVP

When you start planning your MVP, the first thing you have to do is consider what you're trying to learn. It's useful to think about these three basic questions:

1. Is there a need for the solution I'm designing?
2. Is there value in the solution and features I'm offering?
3. Is my solution usable?

Although you can build an MVP to help you answer any of these questions, the first question is probably best answered with traditional design research methods. (In the next chapter, we discuss Lean approaches to this research.) But for the second and third questions, using an MVP adds a lot of value.

If you're trying to answer question two, you will likely find yourself creating an MVP that is optimized for learning. If your question is about the usability of your solution, you will want to emphasize value delivery in your product. This step will allow you to "release" a product into the market and "observe" users interacting with it in realistic contexts.

Here are some guidelines to follow if you're trying to maximize your learning:

Be clear and concise

Spend your time distilling your idea to its core value proposition and present that to your customers

Prioritize ruthlessly

Ideas, like artifacts, are transient. Let the best ones prove themselves.

Stay agile

Information will come in quickly, so make sure that you're working in a medium that allows you to make updates easily.

Measure behavior

Build MVPs that allow you to observe and measure what people actually *do*, not just what people say. In digital product design, behavior trumps opinion.

Use a call-to-action

You will know people value your solution when they demonstrate that they are using it. A call-to-action is a clear phrase, sometimes complemented by an image, that asks the user to take a specific action: "sign up" or "buy now." Giving people a way to opt in to or sign up for a service is a great way to know if they're interested.

Here are some guidelines to follow if you're trying to deliver value to your customers:

Be functional

Some level of integration with the rest of your application must be in place to create a realistic usage scenario.

Integrate with existing analytics

Measuring the performance of your MVP must be done within the context of existing product workflows.

Be consistent with the rest of the application

To minimize any biases toward the new functionality, design your MVP to fit with your current style guide and brand.

Of course, you'll find that you're trying to learn and deliver value at the same time. Keeping these guidelines in mind as you plan your MVPs will help you navigate the trade-offs and compromises you're going to have to make.

Regardless of your desired outcome, build the smallest MVP possible. Remember that it is a tool for learning. You will be iterating. You will be modifying it. You may very well be throwing it away entirely.

And keep one last thing in mind: in many cases, your MVP won't involve any code at all. Instead, you will rely on many of the UX designer's existing tools: sketching, prototyping, copywriting, and visual design.

Prototyping

One of the most effective ways to create MVP's is by prototyping the experience. A prototype is an approximation of an experience that allows you to simulate what it is like to use the product or service in question. It needs to be clickable (or tappable). At the same time, your goal should be to expend as little effort as possible in order to create the prototype, which makes your choice of prototyping tool important.

Choosing which tool to use for your prototype depends on:

- Who will be interacting with it
- What you hope to learn
- How much time you have to create it

It's critical to specify the intended audience for your prototype. Knowing your audience allows you to create the smallest possible prototype that will generate meaningful feedback from this audience. For example, if you're using the prototype primarily to demo ideas to software engineers on your team, the prototype can largely overlook primary areas of the product that aren't being affected by the prototype, such as the global navigation. Your developers know those items are there and that they're not changing, so you don't need to illustrate those items for them.

Stakeholders, often less familiar with their own product than they'll ever admit, will likely need a greater level of fidelity in the prototype in order to truly grasp the concept. To meet the various needs of these disparate audiences, your prototyping toolkit should be fairly broad. You'll want a broad range of methods to communicate your ideas. Let's take a look at some ways to create prototypes and the pros and cons of each.

Low-Fidelity Prototypes: Paper

Made of the most accessible components—paper, pens, and tape—paper prototypes (Figure 5-2) allow you to simulate experiences in a quick, crafty, fun manner. No digital investment is necessary. Using tactics such as flaps to show and hide different states on a page or even creating a "window" for a slideshow of images to move through gives the team a sense of how the product should function. You'll be able to get an immediate sense of what is available in the experience and what is missing. Paper prototyping can give you a sense of how the workflow is starting to coalesce around the interface elements you've assembled. Paper prototyping is especially helpful with touch interfaces that require the user to manipulate elements on a screen.

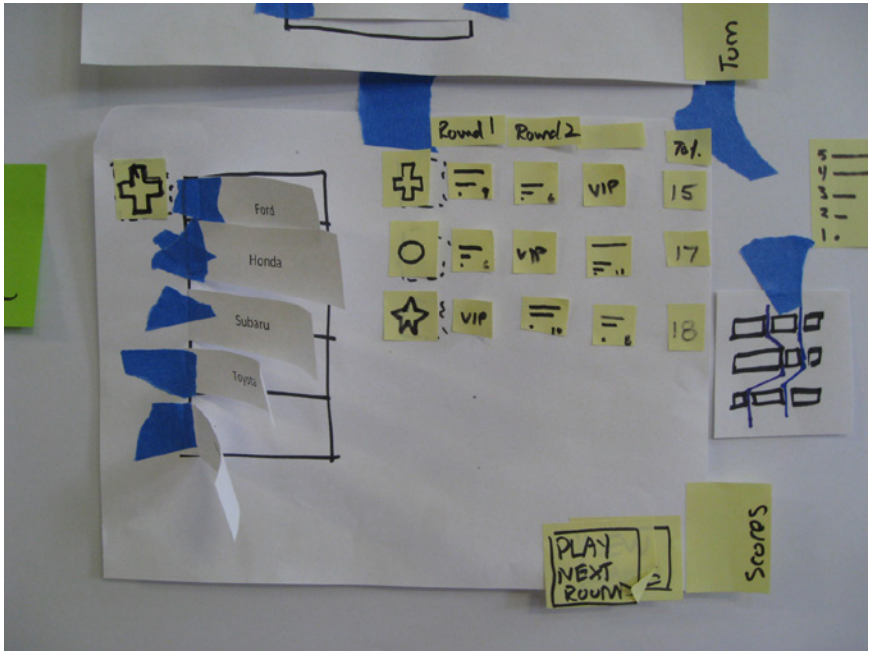


Figure 5-2. *Example of paper prototype.*

Pros

- Can be created in an hour
- Easily arranged and rearranged
- Cheap
- Can be assembled with materials already found in the office
- Fun activity that many people enjoy

Cons

- Rapid iteration and duplication of the prototype can become time-consuming and tedious
- The simulation is very artificial, because you're not using the actual input mechanisms (mouse, trackpad, keyboard, touch screen, etc.)
- Feedback is limited to the high-level structure and flow of the product

Low-Fidelity Prototypes: Clickable Wireframes

Creating an experience with clickable wireframes (Figure 5-3) lets you take a prototype to the next level of fidelity. Your investment in pixels provides a bit more realistic feel to the workflow. Test participants and team members use digital input mechanisms to interact with the prototype, which offers better insight and feedback about the way they will interact with the product at the level of the click, tap, or gesture.

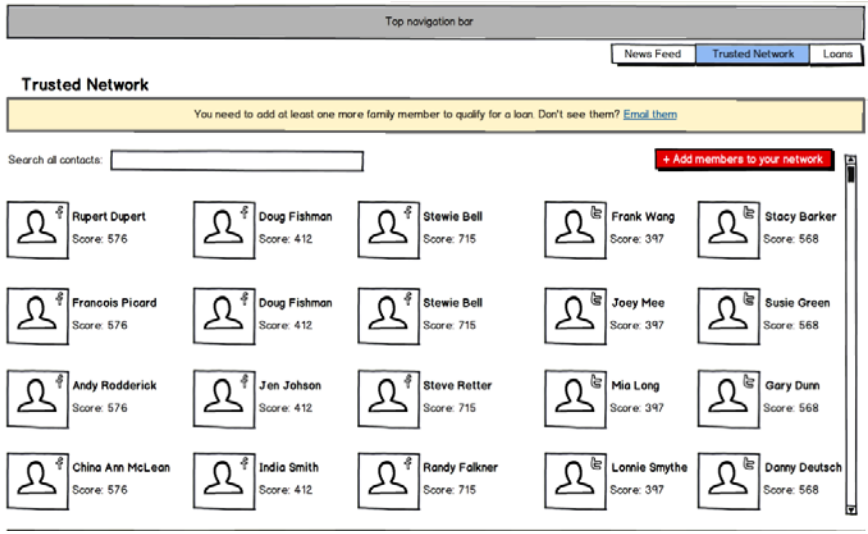


Figure 5-3. *Clickable wireframe prototype example.*

Pros

- Provides a good sense of the length of workflow
- Reveals major obstacles to primary task completion
- Allows assessment of findability of core elements
- Can be used to quickly create "something clickable" to get your team learning from your existing assets instead of forcing the creation of new ones

Cons

- Most people who will interact with these assets are savvy enough to recognize an unfinished product
- More attention than normal is paid to labeling and copy

Tools for creating low-fidelity clickable wireframes

Here are some of the tools that work well for this type of prototyping:

Balsamiq

An inexpensive wireframing tool that provides "sketchy"-looking output. It's the closest thing to digital sketching of interfaces and has a robust community of support. Its limitations are what make it powerful; you can't spend your time tweaking the finer points of the interface, so you spend more time churning through revisions. The ability to link pages together easily makes it a great early prototyping tool.

Microsoft Visio

This program, the granddaddy of wireframing tools, can still be used to link its screens together to create something clickable. It's hard to work quickly with Visio, though: the general challenges of using this product make it less and less attractive as more modern products, both desktop and web-based, enter the market.

OmniGraffle (Mac only)

In many ways, this is the Mac equivalent of Visio, though it is easier to use, has more robust features, and provides better-looking artifacts. You can use images in your drawings, so you can create good-looking artifacts. Still, its core power is in diagramming, not in simulating workflows and interaction.

Microsoft PowerPoint

In a pinch, PowerPoint can still be relied on to fake some level of interactivity. You can use the native drawing tools to draw wireframes and link them together, or you can import mockups, wireframes, or screenshots that you've created in another tool. By clicking sequentially through your slides or by using linked hotspots, you can provide a bare minimum level of fake interactivity. On the Mac, Keynote can be used in the same way. You can also buy libraries of images from Keynotopia that let you assemble realistic-looking mockups. Maintaining these prototypes can end up being very time-consuming.

Fluid Designer/Pop Prototype on Paper

These mobile prototypting tools (and others like them, which are emerging very rapidly) allow you to quickly build prototypes that run on a smartphone. You import images (or photograph sketches) and link them quickly using hotspots. You can simulate simple workflows very quickly.

NOTE

I am aware that there are many options on the market for creating wireframes and prototypes. The list of tools I mention in this section is in no way meant to be comprehensive. In fact, it is highly recommended that you try out as many of these tools as you can. See how well they fit the way you and your team work together. Most products have a free trial so you can give the product a field test before committing.

Mid- and High Fidelity Prototypes

Mid- and high-fidelity prototypes (see Figure 5-4) have significantly more detail than wireframe-based prototypes. You'll use these to demonstrate and test designs that are fleshed out with a level of interaction, visual, and /or content design that is similar to (or indistinguishable from) the final product experience. The level of interactivity that you can create at this level varies from tool to tool; however, most tools in this category allow you to represent pixel-perfect simulations of the final experience. You will be able to create interface elements such as forms, fields, drop-down menus that work, and form buttons that simulate submit actions. Some tools allow logical branching and basic data operations. Many allow some types of minor animations, transitions, and state changes. Additionally, the cost of creating this level of fidelity is significantly reduced with the use of these tools.

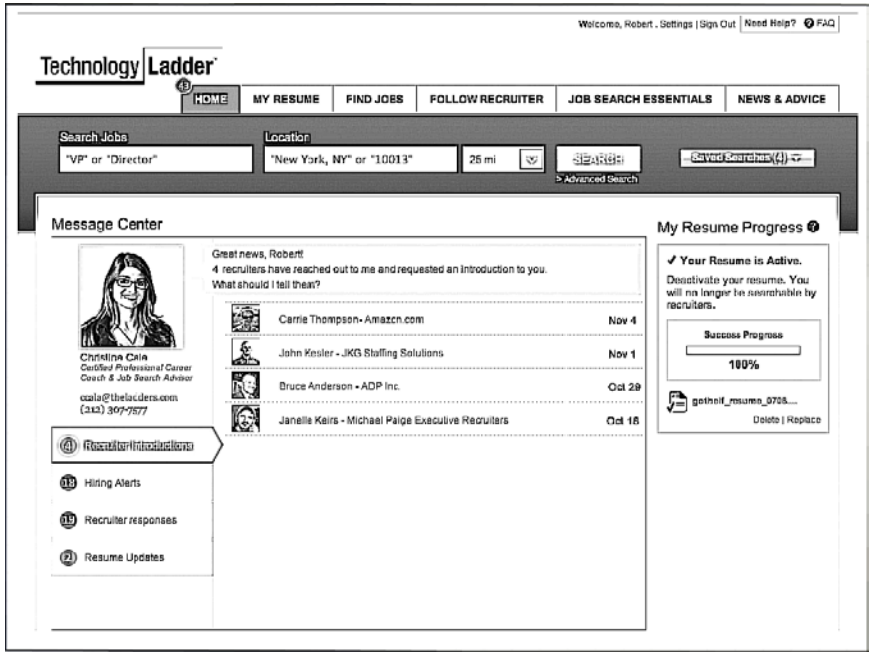


Figure 5-4. *Example of a mid-fidelity prototype.*

Pros

- Produces high-quality and realistic prototypes
- Visual design and brand elements can be tested
- Workflow and user interface interactions can be assessed

Cons

- Interactivity is still more limited than fully native prototypes
- Users typically can't interact with real data, so there is a limit to the types of product interactions you can simulate
- Depending on the tool, it can be time-consuming to create and maintain these prototypes; maintaining a high-fidelity prototype and keeping it in sync with the actual product often involves duplicate effort

Tools for creating mid- and high-fidelity clickable wireframes

Here are some of the tools that work well for this type of prototyping (again, this is only a very partial list):

Axure RP

This increasingly popular prototyping tool allows you to create realistic web pages with screens and forms, and to submit workflows. Axure mockups run in any browser and do an excellent job of simulating web pages. Because it imports images well and supports native HTML user interface elements, it is a very effective mid-fidelity prototyping tool (though you can use it for both low- and high-fidelity prototypes as well). It has good conditional logic, so you can mock up a good range of interactions. A growing community of support is sprouting up around Axure, and many interaction designers have begun using it as their primary tool. Its ability to generate specifications from the prototype is an added bonus for organizations that still make those demands of their teams.

Adobe Fireworks

An old Macromedia acquisition, Fireworks tries to blend the best of Adobe Illustrator with the best of Photoshop and mashes it up in a stew of pseudo-interactivity that makes it a powerful prototyping tool when visual fidelity is important. You can create screens and manage varying states of specific elements. You can add working form components. You can link elements via simple hotspots. You can create custom asset libraries that make the reuse of interface elements efficient and encourage use of the tool.

Coded Prototypes

Coded prototypes offer the highest level of fidelity for simulated experiences. For all intents and purposes, people interacting with this type of prototype should not be able to distinguish it from the final product unless they bump up against the limits of its scope (i.e., they click on a link to a page that was not prototyped). Coded prototypes exist in the native environment (the browser, the OS, on the device, etc.) and make use of all of the expected interactive elements. Buttons, drop-down menus, and form fields all function as the user would expect them to. These prototypes take input from the mouse, keyboard, and screen. They create as natural an interaction pattern as possible for the prototype's evaluators.

Hand-coded and live-data prototypes

There are two levels of fidelity for coded prototypes: hand-coded and live data. The hand-coded prototypes look and function like the end product, but don't account for any kind of real-time data input, processing, or output. They are still just simulations. The live-data prototypes will connect with real-time data and process user input. These are often deployed to real customers and offer a level of analytical insight into customers' usage of the prototype that is not available from hand-coded prototypes. They can also be used when A/B testing certain features or changes to the current workflow.

Pros

- Potential to reuse code for production
- The most realistic simulation to create
- Can be generated from existing code assets

Cons

- Team can get bogged down in debating the finer points of the prototype
- Time-consuming to create working code that delivers the desired experience
- Tempting to perfect the code before releasing to customers
- Updating and iterating can take a lot of time

What Should Go Into My Prototype?

You've picked the tool to create your MVP and are ready to get started. There is no need to prototype the entire product experience. Instead, simulate the most important part of the experience for your customer and your business. Focus on the core workflows that illustrate your MVP.

Focusing on the primary workflows of your MVP gives the team a sense of temporary tunnel vision (in a good way!), allowing them to focus on a specific portion of the experience and assess its validity and efficacy.

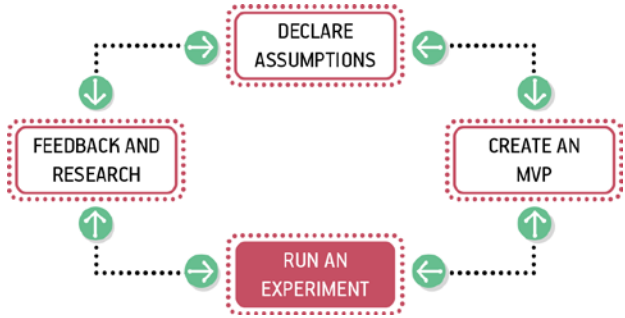


Figure 5-5. *Where prototyping fits in the Lean UX cycle*

Demos and Previews

Test your prototyped MVP with your teammates, stakeholders, and members of other teams. Take it to the lunch area and share it with colleagues who work on different projects. Ensure that people within the company are providing the team with insights into how well it works, how they'll use it, and whether it's worth additional investment. Let stakeholders click through it and give you their insights and thoughts.

Prototypes help show the project's stakeholders that progress is being made. If your team has a demo day (and if it doesn't, it should), bring the prototype there to show progress on the project. The more exposure the MVP gets, the more insight you'll have as to its validity. Next, take your prototype to customers and potential customers. Let them click through the experience and collect their feedback.

Putting It All Together: Using a Prototype MVP

Here's how one team I worked with recently used a prototype MVP. In this case study, the team was considering making a significant change to their offering. We used a prototype MVP to support the research and decision-making process.

This established startup was struggling with their current product—an exclusive subscription-based community for group collaboration. It had been in market for a few years and had gained traction, but adoption had reached a plateau—new users were not signing up. Realizing that a radical change was in order, especially in light of growing competition, they were considering revamping their business model and opening up the product to a significantly broader market segment. Their concern was twofold:

- Would current users accept this change, given that it would alter the exclusive nature of the community?
- Would the new market segment even be interested in this type of product?

The team was worried that they could take a double hit. They feared that existing users would abandon the product and that there wouldn't be enough new users coming on board to make up for the shortfall.

I worked with the team to define our plan as a hypothesis. We laid out the new market segment and defined the core set of functionality we wanted to provide to that segment. It was only a subset of the ultimate vision, but it could be articulated in five wireframes.

We spent a week creating the wireframes using Balsamiq to ensure that our developers, marketers, and executives were on board with the new direction. We showed the wireframes to current customers (twice!) over the course of these five days and ended up with a clickable prototype—our MVP.

The timing for our experiment was fortuitous: there was a conference full of potential customers scheduled for the following week in Texas. The team flew down to the conference and walked the halls of the convention center with the prototype on our iPads.

The mockups worked great on the iPads: customers tapped, swiped, and chatted with us about the new offering. Three days later, we returned to NYC with feedback written on every sticky note and scrap of paper available.

We gathered the notes into groups, and some clear themes emerged. Customer feedback made us realize that although there was merit to this new business plan, we would need further differentiation from existing products in the marketplace if we were going to succeed.

All told, we spent eight business days developing our hypotheses, creating our MVP, and getting market feedback. This work put us in a great position to refine the product to fit our market segment more effectively.

Non-prototype MVPs

For many teams, the default approach to creating an MVP is to create a prototype—to immediately begin designing and writing code. It's easy to understand this approach: we are trained to test our designs and our code, so when we think about validation, we naturally think about creating a product mockup to test. There are many occasions when this step isn't necessary and can even be harmful, though. As valuable as prototyping is, it isn't the only way forward.

Sometimes it makes sense to create an MVP that doesn't simulate your product and instead lets you test something related to your product. For example, when your team is trying to determine the value of a new feature or product, it often makes sense to use a non-prototype MVP to learn whether you're on the right path.

The mantra to keep in mind when creating non-prototype MVPs is this: *you can always go leaner.* To plan your MVP, ask yourself the following questions:

1. What am I trying to learn?
2. What are the main signals I need from the market to validate my hypothesis?
3. Are there any other signals I can test for that will serve as indicators for my main signal?
4. What's the fastest way for me to find this information?

As an example, let's answer these questions for a solution that an ecommerce company wants to test:

1. *What am I trying to learn?* We are trying to learn whether this new ecommerce solution will increase purchases.
2. *What are the main signals I need from the market to validate my hypothesis?* The main signal we're seeking from the market is an increase in completed purchases.
3. *Are there any signals I can test for that will serve as indicators for my main signal?* Instead of completed purchases, can we test for customer intent and use that as a proxy for purchases?
4. *What's the fastest way for me to find this information?* Let's send out an email to a subset of our users offering a few items for sale and count click-throughs for that call-to-action. This will help us determine interest and intent to purchase.

Types of Non-Prototype MVPs

Let's take a quick look at some techniques for creating non-protoype MVPs.

Email

Email is a very powerful tool when it comes to learning about your customers. Open rates, click-throughs, and task completion rates for recipients all provide insight into whether your idea has value.

Google Ad Words

A very inexpensive experiment to run is to purchase Google Ad Words advertisements that target searches relevant to your business. By monitoring what people are searching for, you'll start to get feedback on what kind of language resonates with your audience. By measuring click-throughs, you can see how much interest there is in the words and messages you propose.

Landing Page

A landing page for click-through traffic from Google ads can further validate your thinking. A landing page is the online equivalent of a Wild West movie studio set. It's just a facade of your service, built with a very specific and obvious call-to-action on it. Whether it's Sign-up, Buy Now, or Share-With-A-Friend, every user who completes the task on your landing page counts as validation of your product idea.

The button to nowhere

A feature can be tested on your site by adding a button to the interface that touts the new functionality. That button does nothing more than measure the number of times it's clicked. Each click indicates a customer's desire for that feature. With enough measurable interest, further development of the feature can continue. Of course, you should give the user some explanation of why the feature isn't working. You can use this further interaction as a chance to capture an email address or another bit of feedback.

Hybrids and Creativity

When I talk to teams and entrepreneurs, I'm often impressed by how creative they can be in their approach to creating MVPs. Designing tests is a creative process, and you should use the methods listed in this chapter as inspiration for your creativity. The best approach for you will often be a mashup of many approaches.

Here's an example of how Cheryl Yeoh launched CityPockets using a hybrid approach called a Concierge MVP to find out whether her idea solved a real problem and if there was enough demand to justify building it for real.

Cheryl Yeoh started CityPockets from the hypothesis that people had trouble managing, keeping track of, and redeeming all the daily deals and coupons they purchased online. She interviewed customers to validate that indeed there was a need, but she wasn't sure if her solution—an online wallet for all of these coupons—would bring the kind of value these customers needed. To validate her hypothesis, she launched an MVP version of CityPockets.com that featured only a front end. Building the back-end processing and integration she would need was going to be costly; she didn't want to spend her money unless she was sure her customers would find her service valuable.

Instead of building a back end, Cheryl assigned a unique email address to each customer who signed up for the service. She instructed her customers to forward all of their coupon emails to that address. Behind the scenes, Cheryl was manually entering every coupon into a database. She set an arbitrary target outcome for herself: 500 emails a day. If customers were sending her 500 emails a day, she felt confident concluding there would be enough demand for the service to merit further investment. She would be ready at that point to build a back end to take over the processing.

This approach—though it involved some design and coding—left out the heavy lifting. Instead, it let Cheryl focus her investment on the smallest possible set of features she needed to support her learning. At the end of the day, this is the essence of the Lean UX approach. Design only what you need. Deliver it quickly. Create enough customer contact to get meaningful feedback fast.

Conclusion

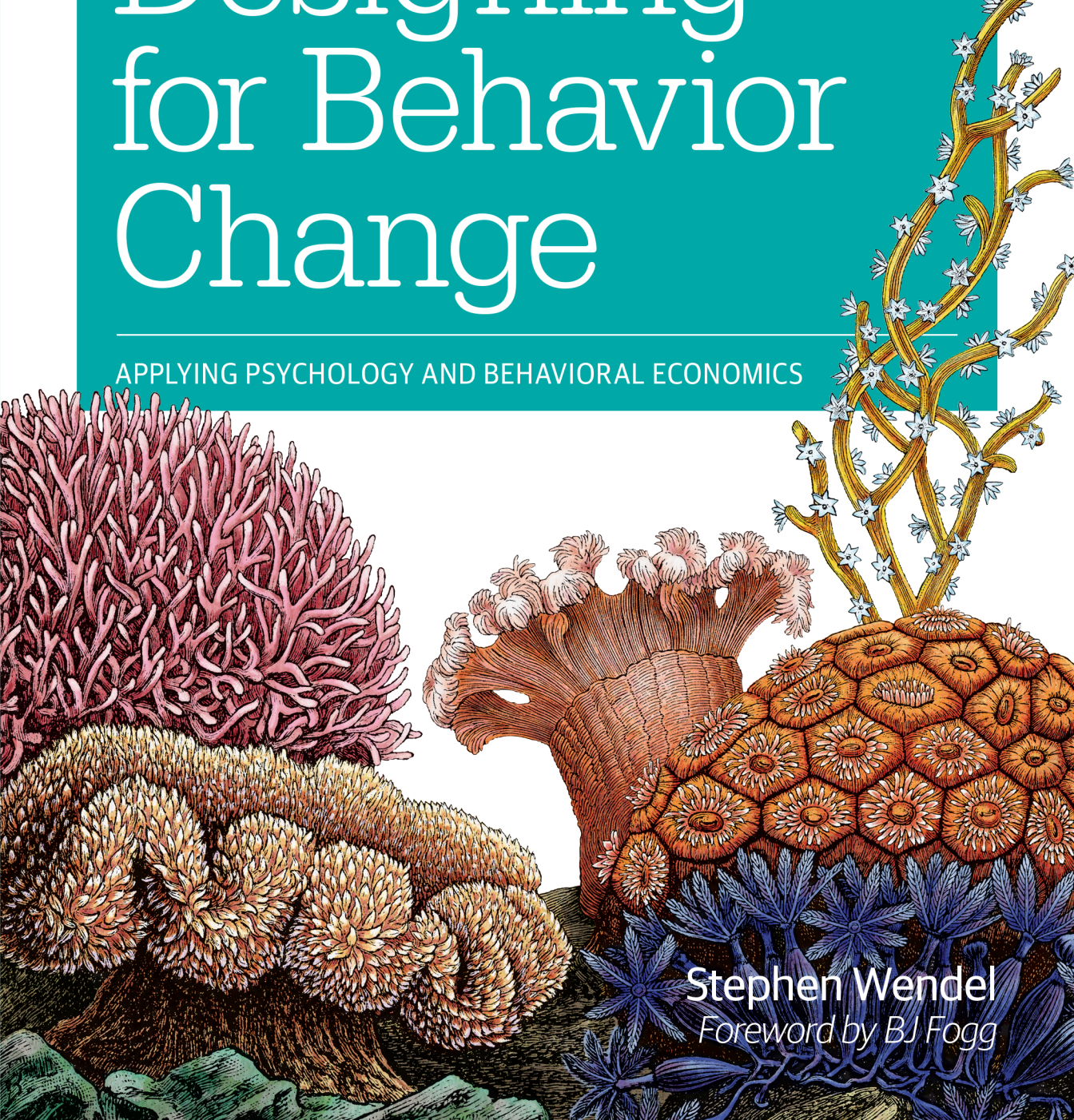
In this chapter, we defined the Minimum Viable Product as the smallest thing you can make to learn whether your hypothesis is valid. In addition, we discussed the various forms an MVP can take, took a closer look at prototyping, and discussed tactics for learning that don't require building prototypes.

In Chapter 6, we take a look at various types of research you can use to make sure that your designs are hitting the mark. We also take a look at how your team can make sense of all the feedback your research will generate.

O'REILLY®

Designing for Behavior Change

APPLYING PSYCHOLOGY AND BEHAVIORAL ECONOMICS



Stephen Wendel

Foreword by BJ Fogg

Designing for Behavior Change

Applying Psychology and Behavioral Economics

Stephen Wendel

O'REILLY® Beijing · Cambridge · Farnham · Köln · Sebastopol · Tokyo

[Contents]

[3]

Strategies for Behavior Change

A Decision or a Reaction: Three Strategies to Change Behavior

How can a product help its users pass all the way through the Action Funnel and actually take action? There are three big strategies that a company can choose from, to change behavior and help users take action. Two of them come straight from the research literature and from the difference between deliberative and intuitive actions. The third is less obvious, but immensely powerful—it's called cheating.

The conscious, deliberative route is the one that most of us are familiar with already—it entails encouraging people to take action, and them consciously deciding to do it. Users have to pass through all five stages of the Action Funnel, and often spend considerable time on the conscious evaluation stage.

The intuitive route is a bit more complex. Recall from Chapter 1 that our lightning-fast, automatic, and intuitive reactions arise from a mix of various elements: associations we've learned between things, specific habits we've built up, our current mindset, and a myriad of built-in shortcuts (heuristics) that save our minds work but can lead us astray. Of these, habits are the most promising route to developing a sustainable path to behavior change because there are clear, systematic ways to form them.[49] And once they are formed, they allow the user to pass effortlessly through two of the stages of the Action Funnel—the conscious evaluation and the assessment of the right timing for action.

The third strategy takes a lesson from Chapter 1, that our minds are usually on autopilot, to the extreme: it decreases users' need to act altogether, so they simply need to give consent if they wish the target action to occur. This strategy—to "cheat"—I'll argue is the most effective and desirable of all.

In the following sections, I'll go into detail about each of these strategies, and when each of them is most appropriate, but in short, here are the guidelines:

Cheat

: If what you really care about is the action getting done, and it's possible to all but eliminate the work required of the user beyond giving consent, then do it.

Make or change habits

: If the user needs to take an action multiple times (like eating better or spending less), and you can identify a clear cue, routine, and reward, then use the "habits" strategy. Also use this strategy if the user is fighting an existing habit—to cleverly undercut it, rather than using brute force to stop it directly.

Support conscious action

: If neither of the other two is available, then you must help the user consciously undertake the target action. There are ways to make this process nicer and easier, but it's still the hardest path to follow.

In each case, the individual makes a conscious choice; what's different is *what is being chosen* and *what happens afterward*. In the first strategy, the person chooses whether to give consent to the action occurring on her behalf. In the second strategy, the person chooses whether to set up the conditions for habit formation (or for stopping an existing habit), and chooses whether to repeat the behavior until the habit is formed (or broken). In the last strategy, the person chooses whether or not to take the original target action. If the action is to be repeated, so is the choice. There's no (ethical) way to avoid having the user consciously choose whether or not to act, but products can change the nature of that choice by selecting among these three strategies.

How can you help users take better pictures with your camera? Provide a manual ("support conscious action"), provide a frequently used and easy menu system ("build habits"), or set intelligent defaults ("cheat").

These *behavioral strategies* provide high-level direction for how the product should be designed: how it accomplishes the process of behavior change. *Behavioral tactics* (such as comparing users to their peers, highlighting the pain of losing an opportunity, or priming them to think about particular topics) don't provide much overall guidance on how the product should work. Instead, they can be slotted in at various junctions in the product to make each piece of the product more effective. Throughout this book, we'll talk about both high-level strategies and lower-level tactics. This chapter is all about strategy, though, so let's get started with the first one: cheat.

Strategy 1: Cheat!

While you can make an action rewarding, easy, familiar, socially acceptable, or any of the other things we talked about in Chapter 1, the activity still involves *work* for the user. Ideally, the company should find ways to shift the user's burden onto the product, by identifying clever

ways to make active participation by the user unnecessary beyond giving informed consent. That's what I call cheating—substituting the user's nasty problem with a much simpler one: deciding whether he wants the product to take the action for him. As you'll see, this strategy is only available in certain cases, but when it is feasible, it is immensely powerful.

Exactly how a company can "cheat" depends on whether the target action is undertaken once or infrequently (like buying running shoes) or repeatedly (like going running each morning). I'll talk about each of these two situations in turn.

Strategies to Cheat at One-Time Actions

DEFAULT IT

To default an action, the company first finds a way to take the action on the user's behalf. Then, it gives the user a choice about whether the product should take the action on his behalf, where the default answer is "yes." The user can say "no" if he so chooses.

Most defaults are invisible—you don't even think about them as defaults; they just happen. In fact, we're not used to seeing the defaults that are all around us, and so we rarely think of it as a solution. To that point, the most common reaction I get to proposing defaults is, "That's great, but there's no way that will work here. You can't default this behavior." Well, maybe. Here are some examples to show how defaulting works in real life:

Behavior change sought: have users save for the future.

Two of the greatest success stories in the recent history in helping users save money are 401(k) auto-enrollment and auto-escalation.[50] (For non-US readers, 401(k)s are retirement savings plans provided by an employer to employees.) Under auto-enrollment, individuals who are eligible to participate in their company's 401(k) plan are defaulted into contributing to the plan, but are given the option to not contribute if they wish. Similarly, auto-escalation automatically increases the contribution rate over time, but the individual can opt out at any time.

The initial action that users take is often quite minimal—signing their name on a package of new-employee documents—and afterward, contributions to the 401(k) plan are automatically deducted from their paychecks and placed into their retirement account

on their behalf. Instead of requiring that an individual choose to contribute to the retirement plan each month (or choose to find the HR representative with the necessary paperwork required to enroll in the plan), this process effectively removes the work required by the user.

401(k) auto-enrollment is a powerful example of increasing savings, but it also can skirt the line between *voluntary* behavior change and trickery. Some employers strive to inform employees about their retirement plans and default contributions. In other cases, the employees don't know about their accounts until they leave their job and get a check—which they quickly spend on non-retirement needs, since they weren't informed and invested in the process in the first place.

The impact of defaults is significant in this case: defaulted (auto-enrolled) plans have nearly *twice* the participation of non-defaulted plans (Nessmith et al. 2007).

Behavior change sought: have users take high-quality photos, rather than crappy ones.

High-end camera manufactures have a problem: many consumers want lots of features, but those same features make the camera sensitive to user mistakes and result in bad pictures.

Good cameras have a simple solution that help people take quality pictures, but still provide power options (and a premium price): the cameras have default settings that are dirt simple and would provide a good picture in most scenarios. In addition, they have all of the fancy bells and whistles that make the product more attractive and expensive than a bargain-basement camera.

Similar defaults are common in computer software ("Would you like the standard install or the scary customized one?")—the options are there, but the software makers have provided intelligent defaults so most people don't have to worry about them and install the software without getting themselves in trouble.

Impact of the defaults: apparently, cameras still can't help us take *interesting* pictures. More seriously, though: do any mass market cameras *exist* anymore that don't have intelligent defaults for things like contrast, white balance, and F-stop?

MAKE IT INCIDENTAL

If the action can't simply be defaulted away, there's another clever option—have the action come along for the ride with something else that users are going to do anyway. In other words, don't have them think about doing the action at all. Make the action happen automatically when the user does something else—something that is inherently more interesting or engaging—but leave the option for the user to decline the action if he so chooses.

Here are two examples:

Behavior change sought: improve people's intake of vital vitamins and minerals.

OK, before I go into the solution, what's the most effective way to improve the amount of vitamins and minerals people get? Convince them of the benefits? Pay them to eat well? Run a public campaign with celebrities endorsing vital minerals? How about this: put it in the food people already eat—with their consent, and without removing other food options. For example, put iodine in salt.

Iodine deficiency is the leading preventable cause of mental retardation (McNeil 2006).[51] It causes stunted growth, infant mortality, lower IQ, goiter (big lumps in the neck), and more (International Council for the Control of Iodine Deficiency Disorders 2013). Two billion people suffer from insufficient iodine. Iodine costs virtually nothing to produce and add to salt.

The story of iodized salt also shows that defaulting can't be allowed to turn into coercion, either practically or ethically. At various times, people around the world have objected to iodine being added to their salt without their consent, causing these iodization campaigns to fail (and iodine deficiency to continue). When there is no way to opt out (non-iodized salt) and no consent, it's not "defaulting"—it's just an ethically problematic mandate. There must be consent among the population *first*.

Impact of making iodine incidental: in many of the places where iodized salt has been used (with consent), the problem of iodine deficiency simply ceases to exist; that's the ideal outcome of any behavior change strategy. In the United States, iodine deficiency is rarely an issue anymore, except where it hasn't been made incidental.

Behavior change sought: have people (voluntarily) contribute money to savings.

One solution in this case is a savings lottery, aka prize-linked savings accounts. A prize-linked savings account is like a lottery in which a person can "buy" multiple tickets.[52] Each ticket is a contribution to their savings account. Like any lottery, there's a jackpot—a big pool of money that one (or more) winners get. Unlike a normal lottery, the participant doesn't lose the cost of their ticket: it's just deposited into their savings account (Tufano 2008). There's a significant upside, but little downside, to participating.

Prize-linked savings makes contributing money to a savings account incidental because some users have a strong preference for playing lotteries (Filiz-Orzbay et al. 2013); the fact that they don't lose the money they use to play is a nice added bonus, but incidental.

Impact of making it incidental: prized-linked savings programs have been highly popular around the world for centuries, starting in Britain (Murphy 2005). They have recently gained traction in the United States through the tireless work of Doorways to Dreams, a Massachusetts-based NGO.[53, 54]

And there are many more examples that we rarely think about in our daily lives. If you want your toddler to take a pill, you crush the pill up and put it in some juice that he likes. The toddler doesn't care or know (and if he doesn't know, he can't complain) about the pill; it's incidental. The juice is what matters.

Strategies to Cheat at Repeated Actions

With repeated actions, you can use both of the last two approaches (defaulting and making the action incidental). For example, with prize-linked savings programs, the savings lottery can be repeated each month to encourage sustained savings contributions. Each time the person acts, savings is incidental. Similarly, each time the person uses an application, she can encounter the same (configurable) defaults.

In addition to these two approaches, another approach becomes possible with repeated actions: you can turn the repeated action into a one-time action by *automating the act of repetition*.

AUTOMATE THE ACT OF REPETITION

Taking an action repeatedly is inherently more difficult than taking that action once, even when the person learns how to do it better over time. So, why not turn a repeated action into a one-time action?

In this scenario, the individual takes a one-time action to set up or accept the automated process, and then the rest is handled without her intervention by the product itself. The principle is simple, and is very similar to defaulting a one-time action: use behind-the-scenes magic to shift the work from the user to the product.

Some great examples of automating repeated behaviors in the health space are exercise trackers that people carry with them throughout the day. These include devices such as the Nike+ FuelBand, Jawbone Up, and Fitbit One; and apps (e.g., RunKeeper) that use GPS or a phone's accelerometer to accomplish this without a separate device (see Figure 3-1). These apps and devices automatically log and compare exercise against a user's target. They've successfully taken something that is annoying but beneficial (logging exercise in a journal, comparing it to one's daily goals), and made the work magically disappear. Once *exercise tracking* was automated away, companies could focus on more interesting (and user-beneficial) target actions—like helping users *exercise more*.

Another example of automating behavior comes from the personal finance space, with software that automatically categorizes transactions and tracks spending—such as HelloWallet (where I work), Mint, and numerous bank websites. In the "old" days (i.e., the 1980s), if you wanted to know how much you had in your checking account, you had to track your spending and balance your checkbook (remember checks?). When ATMs become popular in the 1990s, you also had to track your cash withdrawals. If you had a credit card, it would send you a monthly statement, but before that arrived, you were out of luck.

With personal finance applications, tracking expenses can happen automatically. Each individual transaction is automatically logged, categorized, and, where relevant, compared against a goal or budget item. As with many other forms of automation, once the action is automated for the user, the product team is then free to focus on more interesting and difficult-to-change behaviors—like helping users stay within their budget. But that wouldn't be feasible for most users if they are wasting their time tracking their spending first.

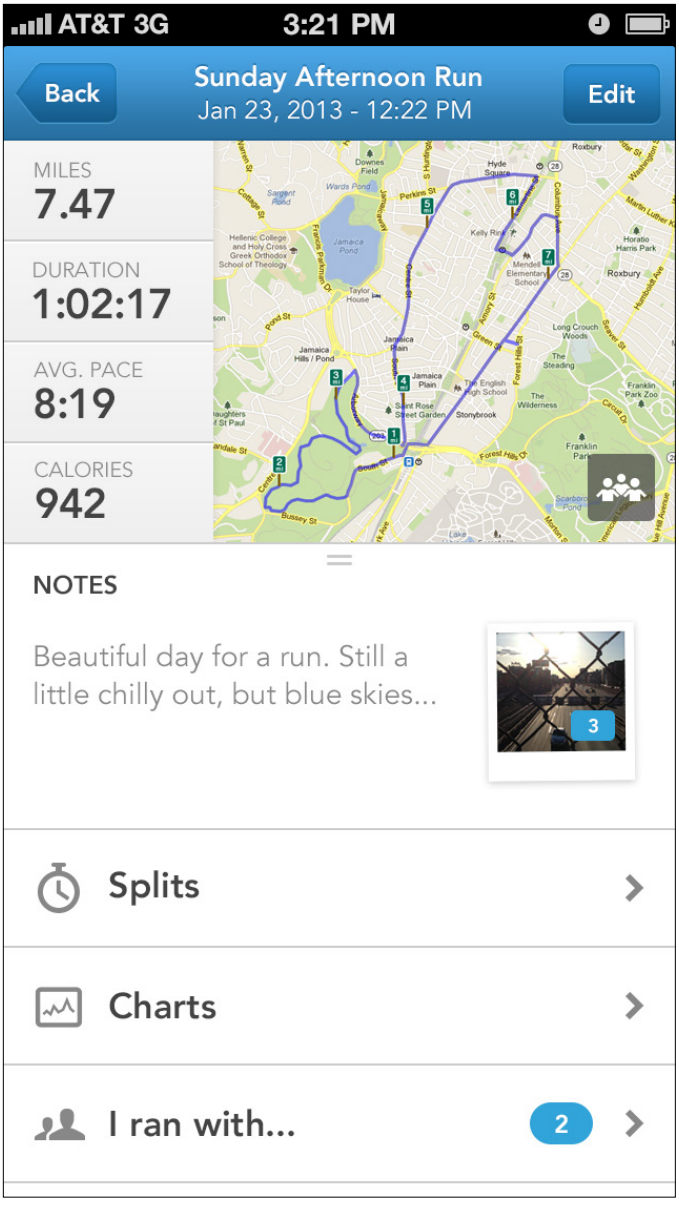


FIGURE 3-1.
RunKeeper, an example of automated exercise tracking

The most powerful combination of all is to combine automation with defaulting—automation makes it a one-time action, and defaulting makes it little more than an acceptance of that automation. I didn't go into detail about this earlier, but 401(k) auto-enrollment is such an example—the savings contributions are automatically deducted, and the default is to enroll in the program.

But Isn't Cheating Well, Cheating?

Before I move on to other behavior change strategies, I'd like to confront an implicit assumption that I've seen in many do-good products—that doing good requires making our users work hard. If we, as people designing for behavior change, want to help people take an action, we should be pushing people to tackle that hill! We know it's hard, they know it's hard, and that's what makes it worthwhile, right?

Well, no.

If the goal is to make people healthier, and the action is consistent with that goal (say, by making the food that people already eat magically become healthier but taste and cost the same), and automation doesn't have nasty side effects, does it inherently matter if the user doesn't have to work hard for it? This probably raises all sorts of hackles. In the case of magically healthier food, I can hear my own inner do-gooder say, "Well, that misses the point—we want people to make wise choices, learn about the wonders of nutrition, be grateful for all the energy we put in to help them, etc.!"

That's why it's vital to be clear about the *end goal* of the product. For example, educating people about health is a laudable goal. But do we really only want to educate people? Or, do we educate in order to help people change their eating habits, which then makes them healthier in the long term? If we could jump ahead, solve this *very particular problem*, and move on to something else, wouldn't that be a good thing? Maybe making food healthier helps with the goal of decreasing vitamin deficiencies, but it doesn't solve the issue of cardiovascular disease. Great—once the food solution is in place, then you can devote your energies to the next problem: helping people decrease cardiovascular disease.

Any product will have multiple aims. But there should be one clear thing that you gauge its success against—a final outcome or goal (that one thing can be a composite of multiple smaller things). I'll talk about how to identify and fine-tune the product's goal in a later chapter, but let's assume you know what it is. When you're clear about what exactly is being sought, go for it. Even if it feels like cheating because it doesn't make people suffer. There are no martyrs in beneficial behavior change. The point of making work magically disappear is that you can move on and help your users with other, more intractable problems.

There's good behavioral science behind this point too. In short, our self-conceptions are constantly adapting based on our own behavior. We often forget or ignore the reasons why we do things and develop a story of who we are based on what we observe about our own behavior (Wilson 2011).[55] For example, if we successfully contribute money to a retirement plan, even if we were defaulted into it, we suddenly feel that that's something we can do—we're savers! The pride that people feel at saving money through automatic enrollment is real, and should not be discounted. That self-conception as a saver then has knock-on effects for other related behaviors—we're prepped for future action.

A classic study in this field is Freedman and Fraser (1966), in which the researchers started by asking homeowners to put a small sticker in their window encouraging safe driving. Weeks later, this randomly selected group was far more likely to accept a large lawn sign about safe driving than other homeowners; a whopping 76% of them accepted the large sign, compared to 17% who hadn't been asked to show the small sticker. In other studies, homeowners were also more likely to accept *other* non-driving related lawn signs. The homeowners started to see themselves as people active in their community, which had broad effects on their behavior.

There are cases when this doesn't work, of course. When people don't know the action occurred at all, then the self-conception doesn't change—but in that case it isn't a voluntary behavior change at all: it's behind-the-scenes trickery.

Cheating at the Action Funnel

Remember the Create Action Funnel from the last chapter? It's difficult for a user to pass all of the way through the funnel from the initial cue to a conscious choice to act with sufficient urgency. The *cheating* strategy takes the funnel, and changes its meaning. With a conscious choice to take a hard action, "success" occurs when the user passes through the funnel. When the product cheats, "success" occurs when the user agrees to the action occurring, but *doesn't* pass through the funnel to stop it from occurring.

Strategy 2: Make or Change Habits

Habits are widely used in products that change behavior. For example, the Nike+ FuelBand builds the habit of checking progress throughout the day with a simple cue (the band itself), routine (checking NikeFuel points), and reward (seeing the points increase). That habit of checking progress is an essential part of the feedback loop used to encourage additional exercise. The mobile phone application Lift builds user-selected habits by cueing people each day to take their action, and then recording it and rewarding them.[56] Before talking about how to build them, let's look at the role they play in behavior change.

Habits Simplify Behavior

Our minds are built to form habits because they are essential—they free our conscious minds from handling mundane details, for everything from thinking about how to prepare our breakfast in the morning to how to properly greet a friend we meet frequently. Without habits, our conscious minds would be overloaded with the minutiae of daily decisions—it would be like we were always moving to a new house, a new job, a new city, and trying to consciously maneuver our way around, every single day of our lives.

Before a habit is formed, however, the user still needs to choose to act. Just like with automation and intelligent defaults, using habits doesn't remove the need for the mind to think; it simply shifts the task to a simpler problem. With habits, the simpler problem is to help the person "get into the groove"—to start taking the action, so that the mind can make it habitual. I'll talk about how to consciously *start* taking an action in the next strategy; for now, we'll focus on how to build the habit given the user's initial commitment to act.

From a behavior change perspective, habits are both a boon and a bane. If a product helps the user form a habit, then the person can act on autopilot. The product team, and the user, can move on to tackling harder problems—just like the "cheating" strategy discussed earlier. But "bad" habits work on autopilot just as much as "good" ones. The next two sections talk about how to form (hopefully "good" habits) and clever ways to attack existing "bad" ones.

How to Build Them

Chapter 1 described the two basic types of habits: habits created out of simple repetition (cue-routine, cue-routine, etc.), and habits that have the added feature of a reward at the end (cue-routine-reward) that drives the person to repeat the behavior. Your product's users could form habits by simple repetition, but then the burden of work and willpower is all on their side. When designing for behavior change, add a reward at the end to help bring people back while the habit is forming.

The cue-routine-reward process is depicted in Figure 3-2. Charles Duhigg popularized the process in the *Power of Habit* (Random House, 2012), building on an old tradition in applied behavioral analysis.[57]

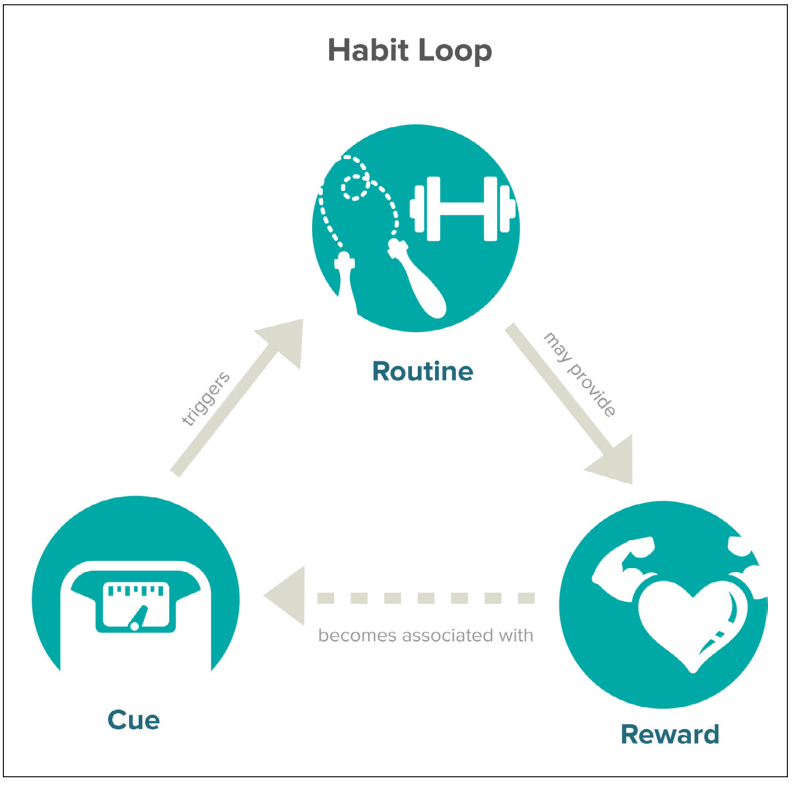


FIGURE 3-2.
The cue-routine-reward process described by Duhigg (2012). For example, seeing the scale in the morning triggers the exercise routine. The immediate reward is a pleasant muscle burn.

To build habits with a product, here is a straightforward recipe:

1. Identify a routine that should be repeated dozens of times, without significant variation or thought each time.
2. Identify a reward that is meaningful and valuable for the user.

3. Identify a clear, unambiguous, and single-purpose cue in a person's daily life or in the product itself (an email, an alert, etc.).
4. Make sure the user knows about the cue, routine, and especially, the reward.[58]
5. Make sure the user wants to and can undertake the routine (i.e., the user needs to make the conscious choice to act).
6. Deploy the cue.
7. Either facilitate the routine or, at least, seamlessly and immediately track whether the routine has actually occurred.
8. Have the product *immediately* reward the user once the routine has occurred. That allows dopamine in the brain to reinforce neurons associated with the cue and routine before the memory fades.
9. Repeat steps 6–8, tracking completion times and rates, and adapting the process until it's right.

There's a lot of nuance there, of course.

First, the *cue* really needs to be single-purpose and unambiguous (i.e., after the habit is formed, the cue is linked to the specific routine and nothing else), because you want to avoid the mind having to think about what to do when the cue occurs. Fogg and Hreha (2010) argue that the triggers (i.e., cues) can be:

- Directly tied to another event (e.g., looking at the bathroom mirror first thing in the morning is connected to picking up your toothbrush)
- At a specific time of day every day or every week

The trigger/cue can be "internal" (boredom or hunger) or "external" (seeing the clock first thing in the morning, or getting an angry email). Internal triggers are great, since they are inherent in the human condition; however, lots of other things in one's life compete for the same triggers (which makes them not single-purpose and thus ambiguous). External triggers can be just as effective, if wisely constructed.

Second, while the *routine* must be structured so that it can occur effectively without thought, it need not be "stupid" or "simple." Good driving, for most people, is a (complex, impressive) habit. Remember how hard it was to learn to drive? Remember all of the thought that was required just to start the car and get it going? Yet, after learning, we

avoid getting too close to other cars while on the road, we coordinate what our eyes see with what our hands do to steer, and so on. The reason is that driving uses a set of hierarchical habits—large, complex habits built out of thousands of small, routinized behaviors that are cued from the environment and linked to one another in succession. Each piece is structured so that it can be consistently executed after the cue without conscious thought.

Routines that can be made into a habit often will have a strong and clear feedback loop (i.e., after the action is taken, the reward is immediate and unambiguously tied to the success). Habit formation is not a conscious event, though we can consciously put ourselves in situations where we'll learn them.

Third, the *reward* need not be offered every time, as long as it is still clearly tied to the routine. Random rewards are quite powerful in some circumstances. In the operant conditioning literature, habits with random reinforcement take the longest to *form* but also take the longest time to *extinguish* once the reward is no longer given. Gambling provides the ultimate random reward—and once you have the bug, it's difficult as all heck to get rid of. One reason that random reinforcement is so powerful is that our brains don't really believe in randomness. We look for patterns everywhere. So, part of the desire driving a random reward is our brains trying to find a pattern (ever talk to a gambler who has "a system"?).[59]

And finally, a key part of using products to build habits is experimentation and fine-tuning. Your product is probably going to get it wrong the first time—the cue won't be clear or won't grab the user's attention, the user may stop caring about the reward, or the context for the routine might change, and conscious thought is required.[60]

Changing Existing Habits

This book is about helping users take action. Sometimes, though, that may require intentionally stopping a habit, instead of just adding new behaviors. For example, at some point, improving fitness through exercise means not just exercising more, but also sitting less. And that means overcoming an existing habit.

Unfortunately, it can be extraordinarily difficult to stop habits head-on. Brain damage, surgery, even Alzheimer's disease and dementia sometimes fail to stop habits, even as other cognitive functions are severely impaired (Eldridge et al. 2002). BJ Fogg, for example, argues that stopping existing habits is the hardest behavioral change task to undertake (2009b).

Why are habits so difficult to change? First, it's because habits are automatic and not conscious. Our conscious minds, the part that would seek to remove them, are only vaguely aware of their execution (see Dean 2013); we often don't notice them when they occur, and we don't remember doing them afterward. Across dozens of studies on behavior change interventions, researchers have found that the conscious mind's sincere, concerted intention to change behavior *has little relationship to actual change in behavior* (Webb and Sheeran 2006).

Second, it's because habits never truly go away—once a habit is formed (i.e., the brain is rewired to associate the stimulus and response), it doesn't normally un-form. It can remain latent or unused, but under the right circumstances, that circuitry in the brain can be activated and cause the habitual behavior to reappear.[61]

Another way of thinking of habit cessation is this: if stopping bad habits were easy, we wouldn't need so many darned books on everything from stopping smoking to dieting.[62] Nevertheless, one can draw lessons from the literature on habit formation and change—which can save product teams needless pain and suffering. There are five main options that product teams can take to handle an existing habit:

1. Avoid the cue.
2. Replace the routine.
3. Cleverly use consciousness to interfere.
4. Use mindfulness to avoid acting on the cue.
5. Crowd out the old habit with new behavior.

In each case, the person doesn't engage in a direct confrontation to simply suppress the habit. That takes constant willpower, which is finite and unsustainable in most cases.

OPTION 1: HELP THE PERSON AVOID THE CUE

The cue signals the brain to engage in the problematic behavior; one way to stop a habit is to avoid the cue (Wood et al. 2005). For example, in addiction counseling, counselors advise addicts to change their environment so that they don't encounter the things that remind them to act. If you always stop for a drink when you see the bar on the way home, then change your route home so you don't see the bar anymore.[63]

Designing a *product* to help people avoid cues is especially tricky. First of all, most cues for bad habits are, by definition, outside of the behavior change product. People use the product in order to change the habit—the product didn't cause the bad habit. So, the product must help the person avoid the cues themselves: the product must provide guidance and instruction. And the individual must first know what the cues are—and be able to successfully avoid them.

Second, because the routine is outside of the product, the application usually won't know if the person has engaged in the behavior. It's up to the user to report falling off the wagon—which is doubly difficult. External monitoring systems are required—like the breathalyzers that alcoholics install in their cars to stop them from driving drunk. Much more is required in the case of chemical addictions like alcoholism, but we can learn from these efforts as we design products to stop less intractable habits.

While this route is clearly challenging, there are products that have successfully done it. One example is Covenant Eyes[64]—software that helps people who are struggling with sexual addiction or want to avoid the temptation before a habit is formed (see Figure 3-3). It helps users avoid cues (by filtering out sites with explicit content) and/or automatically monitors web usage to inform accountability partners of when the person does access pornography.

FIGURE 3-3.
Covenant Eyes, an application to stop the habit of viewing sexual material online, via filtering and automatic monitoring

OPTION 2: CHANGE THE HABIT INTO SOMETHING ELSE

The other strategy that products can use to change a bad habit is to transition an existing cue and reward to a different (more beneficial) behavior. In *The Power of Habit* (Random House, 2012), Duhigg describes two elements that are needed: routine replacement and a real belief that the habit can change.

Routine replacement works by hijacking the cue and the reward, and inserting a different routine between them. He uses the example of taking a snack break when you're not really hungry. The cue may be that you're having a down moment at work or watching a commercial on TV. The reward would be the relief of (momentary) boredom and the pleasant crunching sensation of the snack. To hijack this process, one needs to:

1. Identify the trigger, and the reward (if appropriate).
2. When the trigger occurs, consciously engage in a different routine that provides a similar reward (like doing a crossword puzzle when bored during commercials).

3. Continue that conscious switching of routines until the new habit is instilled.

The process of consciously replacing routines is also known as "competing response training." It is used in the treatment of people with Tourette's syndrome (involuntary tics), and has shown dramatic results in experimental testing (Piacentini et al. 2010; Dean 2013).

For especially difficult habits, like smoking and drinking, swapping in a new routine isn't enough, though. The new reward is never quite like the old one. Swapping can handle everyday behavior, but when times are tough, people can be immensely tempted to "fall off the wagon." Something else is needed to get through those dark times and back to the day-to-day humdrum that they *can* handle. That something else can be faith that the hard times will pass. It can be a religious faith, a personal faith in themselves, or a faith in others that pulls them through. Either way, it's an internal narrative that things will get better.

How does routine replacement work in practice? One of two ways. First, you can ensure that the product itself is present at the moment when the cue normally occurs. At that moment, it would remind or entice the user to do the new routine instead of the old one. After the routine is done, it would reward the user—or encourage him to reward himself.

The other route is trickier and is needed when the product *isn't* present when the user encounters the cue. As with avoiding the cue (described in the previous section), the product must advise and prepare the individual for the moment of temptation, and find some way of tracking what action the person took. ChangeTech.no has an intensive program of support and tracking that accomplishes this, with over 400 points of contact with individuals during their smoking cession program. And, its method has shown positive results in randomized control trials (Brendryen and Kraft 2008).[65]

An example of in-the-moment hijacking of habits that we're all familiar with is shopping in brick and mortar stores with a smartphone:

- *Cue*. See a camera, computer, or something else you like.
- *Old routine*. Pick it up, go to the cash register, buy it.
- *New routine*. Look it up on the phone, compare price (usually lower), and buy it.

- *Reward*. Feel great about saving money, imagine yourself using the cool camera, receive item, and so on.

This habit hijack is killing brick and mortar stores. It's not a "beneficial behavior change," but it's the same underlying process.

OPTION 3: USE CONSCIOUS INTERFERENCE

Our big brains are really good at blocking our own autopilot; properly deployed, they can interfere with habits in progress *without requiring direct willpower to overcome the action*. Thinking = bad, for a habit at least. In sports, masters of their game sometimes "choke" because they consciously cut into a process that normally runs on autopilot, and this happens in any field of mastery (Baumeister 1984; Gallwey 1997). To interfere with a habit: think about it. Look especially for what triggers it. Then closely examine the routine that's normally automatic—just by thinking about it (consciously), we can interfere with its smooth execution.

Products that do this should be present at the time of action and can grab the user's conscious attention to his or her behavior. The Prius is well known for functioning this way. The car's consumption monitor provides ongoing, immediate feedback about the car's gasoline consumption. This in-the-moment feedback can break people out of their existing driving habits by making them consciously aware of what's going on, causing them to use less gasoline, aka "the Prius Effect."

In order for this approach to work, like all habit intervention (and habit formation) approaches, it must be voluntary. If someone doesn't care about mileage or finds the car's consumption monitor annoying, he won't listen to it. It starts with the conscious choice to act.

OPTION 4: USE MINDFULNESS

Another, subtle way to overcome bad habits is by employing mindfulness. Mindfulness is a concept used in Buddhism to refer to awareness of the present moment and its experiences, without judging or trying to control them. It's a mental state of openness and acceptance of events and sensations as they occur. Mindfulness-based therapies are increasingly popular in the treatment of mental conditions, such as acute stress, anxiety, and depression (Hofman et al. 2010). Similar to mindfulness meditation in Buddhism, these therapies entail an intentional focus on the present moment without interference or judgment (Shapiro et al. 2006).

By bringing the cues that trigger habitual behavior into conscious awareness, it's possible for one to be aware of the trigger without acting on it. For example, mindfulness has been shown to limit undesired, but habitual, binge drinking (Chatzisarantis and Hagger 2007). A number of apps, such as Headspace,[66] support mindfulness to reduce stress or increase focus, though do not target habit change in particular.

OPTION 5: CROWDING OUT OLD HABITS

Another way of approaching habit change is to crowd out the old habit with new behaviors—a method that combines option 1 and 2 (and sometimes 3). In this method, you *focus on doing more of what you want instead of less of what you don't want.*

For example, think about someone who is in bad shape, spends lots of time watching TV, and has bad eating habits. The person starts to go to the gym to exercise more (creating a new habit). As the person goes to the gym, he meets new people, and enrolls in exercise and cooking classes with them. Slowly, the amount of time available to watch TV decreases. The person simply isn't at home as much, which leads him to avoid the old cues to watch TV. Also, because of the cooking class, and new ways of eating and cooking, there simply isn't the hunger and opportunity to use his old eating habits; they are slowly being replaced.

Naturally there are multiple forces at work in his life, such as changing self-identity, and changing social norms. However, as the structure of his daily life changes, the old habits fade away—not through a direct assault, but because other things are taking up his time and satisfying his hunger pangs. This only works if he gets far enough down the path of habit change—and doesn't quit going to the gym soon after signing up, as so many other people do. The initial choice to push ahead, before the habit is formed, is a conscious one.

Strategy 3: Support the Conscious Action

You probably noticed that both of the previous two strategies involved removing user work and simplifying the problem. But the simplified problem still requires a conscious choice to act. That's unavoidable, and it's even a good thing. The conscious mind must be engaged at some point. Ideally, that interaction entails informed consent—and the product automates or defaults the rest. Or the conscious choice can be to start down the path of habit formation or habit change.

The explicit strategy of making a conscious choice over each part of the action should generally be avoided because it requires additional effort on the part of the user. All else being equal, more effort means less chance of acting. But sometimes, a head-on approach is used (or required). That means passing through all five stages of the Action Funnel (cue-reaction-evaluation-ability-timing), in order to execute the action itself.

The conscious choice to act is a primary focus of this book, and Chapters 6–8 cover it in great detail. They describe how to support users to take conscious action—whether the action is the original behavior that the company sought to address, or a restructured action like giving consent to an automatic action taken on their behalf.

A Recap of the Three Strategies

This chapter discusses the three primary strategies for behavior change. In each case, there is a choice that individuals make, but the nature and subject of that choice changes. Here's a quick recap of when to use each one, and how they fit into the larger picture of behavior change:

Cheat

- *What it is*. Help the user avoid the work of the action altogether by making the outcome occur by default when the user interacts with the software or when the user takes a *different* action, or fully automating a repeated behavior after consent is given.
- *What is consciously chosen*. Whether or not to give consent to the action occurring on the user's behalf.
- *Examples*. 401(k) auto-enrollment; substituting healthier ingredients into the food people already eat.
- *Use this strategy*. When you can replace a hard action with informed consent. This is not appropriate for overcoming ingrained habits, nor is it appropriate for cases in which the user needs to personalize the action to specific needs—that requires conscious, active involvement.

Make (or change) habits

- *What it is*. Help the user avoid conscious effort and thought by making the desired action an automatic response to a trigger. Or when changing habits, cleverly attack the habit's structure to hinder it from occurring.
- *What is consciously chosen*. Whether or not to set up the conditions for a habit to form (or be broken).
- *Examples*. At the supermarket, go down the produce aisle before the canned foods aisle; walk once a day.
- *Use this strategy*. Whenever the user wants to undertake a behavior that is done multiple times, in a consistent context. Also use this when trying to overcome existing habits, with the tricks described as part of the strategy, rather than using brute force to consciously override habits.

Support the conscious action

- *What it is*. Help the user think about the action, and take the necessary steps (consciously) to make it happen.
- *What is consciously chosen*. Whether or not to take the target action.
- *Examples*. Educating people to get a good mortgage; encouraging people to sign up for (and attend) a yoga class for the first time.
- *Use this strategy*. Whenever the first two strategies aren't feasible, especially when the action is complex, novel, and requires the user to make numerous small choices that can't be defaulted.

On a Napkin

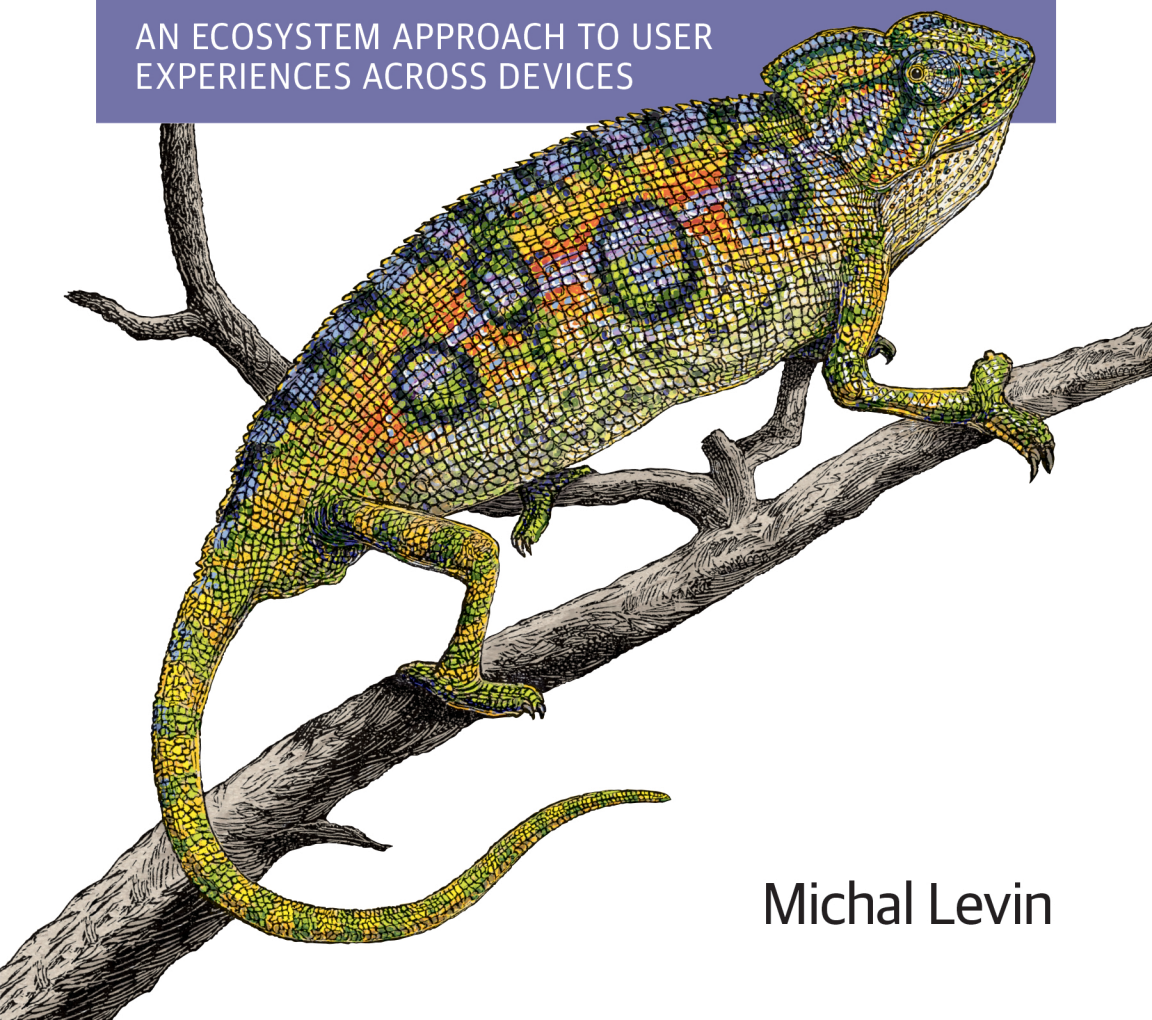
- Look for technical solutions to remove user work wherever possible; it's often much more effective to engineer a solution than it is to change behavior.
- Three technical solutions are automating the action behind the scenes, using intelligent defaults, and making the behavior a side effect of something else the user is doing.
- Habits are immensely powerful ways to lock in repeated behaviors. They require an unambiguous cue, unvarying routine, and a meaningful, immediate reward.

- When possible, avoid trying to stop an existing behavior. The product should build new ones instead.
- If stopping a habit is required, help users avoid cues, replace the routine, crowd it out, or draw conscious attention to the trigger and routine via mindfulness.
- No matter what, some type of conscious choice is required for voluntary behavior change—the strategies presented here are all means to simplify the choice and the work required of the user.

O'REILLY®

Designing Multi-Device Experiences

AN ECOSYSTEM APPROACH TO USER EXPERIENCES ACROSS DEVICES



Michal Levin

Designing Multi-Device Experiences

An Ecosystem Approach to User Experiences Across Devices

Michal Levin

O'REILLY® Beijing · Cambridge · Farnham · Köln · Sebastopol · Tokyo

[contents]

[1]

An Ecosystem of Connected Devices

> What does it mean to design a product in a world where people own multiple devices and use them interchangeably? This chapter describes how we got to this multi-device era, and introduces a new ecosystem design framework to help us both navigate and influence this new reality.

We have entered a world of multi-device experiences. Our lives have become a series of interactions with multiple digital devices, enabling each of us to learn, buy, compare, search, navigate, connect, and manage every aspect of modern life.

Consider the hours we spend with devices every day—interacting with our smartphones, working on our laptops, engaging with our tablets, watching shows on television, playing with our video game consoles, and tracking steps on our fitness wristbands. For many of us, the following are true:

- We spend more time interacting with devices than with people.
- We often interact with more than one device at a time.

The number of connected devices has officially exceeded the 7 billion mark, outnumbering people (and toothbrushes) on the planet.[1] By 2020, this number is expected to pass 24(!) billion.[2] This inconceivable quantity not only attests to the growing role of these devices in our digital lives, but also signals *an increasing number of devices per person*. Many individuals now own multiple connected devices—PCs, smartphones, tablets, TVs, and more—and they are already using them together, switching between them, in order to accomplish their goals. Ninety percent of consumers use multiple devices to complete a task over time—for example, shopping for an item might entail (1) searching and exploring options at home on the PC, (2) checking product information and comparing prices in-store using your smartphone,

and (3) writing product reviews on a tablet.[3] Eighty-six percent of consumers use their smartphones while engaging with other devices and during other media consumption activities, as shown in Figure 1-1.

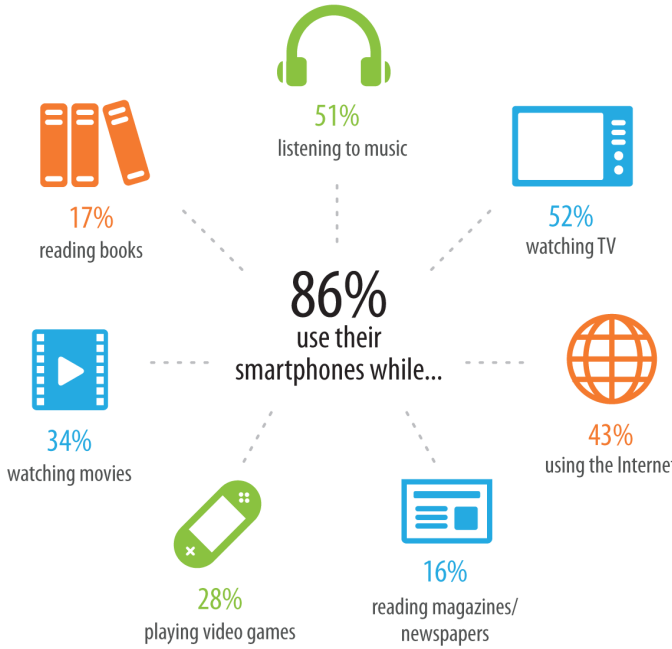


FIGURE 1-1
Multitasking view—activities take place and other devices are used simultaneously with the smartphone.[4]

While each device plays an important role in many of our daily activities, *their real power exists in how they are used together with other devices.* This multi-device usage sets the foundation of a product *ecosystem.*

The Concept of an Ecosystem

Biologists use the term *ecosystem* to describe interconnections within our natural world—a community of living organisms (plants, animals, and microbes) in conjunction with the nonliving components of their environment (elements like air, water, and mineral soil), interacting as a system. An ecosystem essentially describes a network of interactions—among organisms, and between those organisms and their environment—which together create an ecology that is greater than the sum of its parts.

In looking at the world of online apps and electronics today, we can see a type of ecosystem emerging. In this system—this climate of multiple devices—we see smartphones, tablets, laptops, TVs, and other connected devices all interacting with one another and wirelessly sharing data. These interactions are shaped by the different ways in which individuals use the content and services that flow between devices, in different contexts, en route to their goals (Figure 1-2).

FIGURE 1-2
An ecosystem of connected devices, including those that are already available today and a few that will join soon.

The growing number of connected devices, especially mobile ones, is progressively changing the way people perceive, experience, and interact with products and each other. Our mission as designers and product creators is to understand the different relationships between connected devices, as well as how individuals relate to them, in order to create natural, fluid multi-device experiences that allow these dynamic changes. These experiences should focus on how the (increasing) set of connected devices can best serve users' needs as they move between activities and contexts throughout the day.

You can choose to build your product as an isolated cell on each device, replicating the same basic experience, and thus offering independent access to *everything*, *everywhere*, *anytime*. Or you can foster an ecosystem approach that captures the dynamically changing needs and contexts that accompany shifting devices, putting the emphasis on delivering the *right thing* at the *right place* at the *right time*.

The latter—the context-driven approach—is where I believe we'll find our biggest opportunities.

The 3Cs Framework: Consistent, Continuous, and Complementary

In this era of proliferating connected devices, one goal is becoming clear: clients want to see their products on as many screens as possible. At the very least, we'll need to get a product up and running across the basic, core set of screens already deeply embedded in our daily lives: the smartphone, tablet, PC, and TV.

How do we handle this design complexity, given the myriad devices on the market already (and those that are on the way)? How should we approach a multi-device experience design without overwhelming our users, and being overwhelmed ourselves?

When I was struggling with these questions a while back, I was inspired by the multi-screen patterns work done by Precious in 2011.[5] The company's pioneering work established a conceptual model for approaching the emerging multi-device world. However, as I deepened my work in this space, I noticed several challenges that arise when we try to make clear distinctions between the different patterns; this is due in part to particular pattern overlaps and semantic blurs across them. To cut through the confusion, I adopted a framework, or set of building blocks, that has proven to be durable and immensely relevant for approaching ecosystem design. I call the framework *3Cs*: *consistent*, *continuous*, and *complementary*.

In *consistent design*, the same basic experience is replicated between devices, keeping the content, flow, structure, and core feature set consistent across the ecosystem. Some adjustments are made to accommodate device-specific attributes (mainly screen size and interaction model), but overall the experience can be fully consumed, in an independent manner, on any device.

Spotify is a good example of consistent design. It enables users to discover and listen to their favorite music from everywhere—their computers, their mobile phones, their tablets, and their home entertainment systems.

While consistent design provides access to everything, anywhere, anytime—a first important step in bringing value to users through multi-device use—it often doesn't capture the full potential of an ecosystem. Consistency overlooks several significant factors involved in the user's experience: context (delivering the *right thing* at the *right time*),

multi-device relationships (ways devices can supplement and support one another), determining the best device for the task, and scaling the experience to a fully connected world (that goes beyond smartphones, tablets, PCs, and TVs).

To accommodate these needs, we need two additional design building blocks—continuous and complementary design.

The hallmark of *continuous design* is that the experience is passed on from one device to another, either continuing the same activity (watching a movie, reading a book) or progressing through a sequence of different activities, taking place in different contexts but all channeled toward achieving the same end goal (like getting ready to go to work in the morning).

The hallmark of *complementary design* is that devices complement one another (with relevant info/functionality), creating a new experience as a connected group. This experience can encompass two forms of device relationship: collaboration and control.

Google Chromecast, which turns a smartphone or tablet into a remote control for the TV, is one example of such experience design. These devices can then be used to browse the content on TV, control playback, and adjust volume.

The 3Cs provide a framework for thinking about how users accomplish a single goal using multiple devices. Rather than providing a technology-focused framework—stemming from device form factor, size, and resolution—the 3Cs focus on *people*, looking at the relationships between individuals and their devices, and how the latter can support them along their task flow en route to their goal.

Using an analogy from the music world, let's imagine that devices can be used in any of the following ways to elevate a user's experience:

- As a solo instrument, where each performs the entire piece from start to end (consistent)
- As a step in a sequence, by splitting the music into pieces and playing them with other instruments, one after the other (continuous)
- As part of an ensemble, where instruments play together in a coordinated manner to create a harmonized music piece (complementary)

In this musical group, *you* are the conductor...and sometimes even the composer. You decide which devices take part in the multi-device composition, what role each one plays, and how they all orchestrate the complete ecosystem experience. How?

In the following chapters, we will review many different multi-device experience examples through the lens of the 3Cs, demonstrating different relationships between devices, and how they can serve individuals in their tasks. As you will see, the 3Cs help you decide what flows and functionality should be featured when, and how the experience elements should be distributed across devices.

Bear in mind that each design approach is useful for addressing certain user contexts and use cases. For that reason, no single approach is useful in all situations, and quite often the best design employs several approaches (more on that in Chapter 5).

By mapping the variety of contexts across an experience, and then framing the roles each device plays in the overall ecosystem, we can create a clear narrative and mental model for that multi-device experience. Once *we* have this clear understanding in mind, we can translate it to design decisions for each device and for interaction points between devices. In doing so, we can help users navigate (and make sense of) the increasing complexity involved in having more and more devices, guiding them toward a more effective, productive, and delightful multi-device usage.

A Glance at the Past: Portable Computer Ads from the 1980s

Figures 1-3 and 1-4 are two portable computer newspaper ads that illustrate well how far technology has come in less than 30 years.

FIGURE 1-3

Hewlett-Packard, 1984 ad for The Portable computer—a nine-pound computer, with 656 KB memory; by comparison, smartphones today weigh on average 130 grams (0.286 pounds) and offer 1–2 GB RAM and 16–32 GB internal memory.

FIGURE 1-4

IBM 1984 portable personal computer ad ("The first IBM PC you can pick up and take with you"). This computer weighed 30 pounds, and was the approximate size of stationary computers today.

But how did we arrive at this multi-device era? Why are its opportunities so unique?

Single-Device Design Is History (Don't I Know It!)

As a UX designer, I can remember how not that long ago we all labored over the personal computer (PC) when designing a mass-market product. Our parameters were bounded: a single device, stationary, in landscape format, with relatively high resolution, and based on mouse and keyboard interaction.

Life was definitely simpler in many ways, but even back then, we had our share of design challenges for these shifting environments: ever-changing screen resolutions; multiple browsers with no common standard; and various technical limitations in terms of interaction design, latency, and implementation.

While those challenges represented the growing pains of a technology evolution, our current era is a technology *revolution*.

By *multi-device revolution*, I'm referring to the fast-growing number and diversity of connected devices—from smartphones, tablets, PCs, and TVs, through smartwatches, smartglasses, wristband activity trackers, smartfridges, and connected toothbrushes, to (soon enough) any physical object that can be connected to the Internet via sensors (what is known as the *Internet of Things*, which we'll discuss in Chapter 6). However, this revolution is not characterized just by new screen sizes, input methods, form factors, or increasing processing power. It also introduces new ways these multiple devices enable us to connect, operate, interact, work, and affect our surroundings—ways we didn't have before.

If we look back, we can see that three processes in particular were supremely influential in changing the multi-device landscape from a UX perspective: smartphones as a commodity, burgeoning application stores, and the emergence of tablets on the market. These processes signify the critical transformation from a single-device–focused design to the need to adopt an ecosystem approach to multi-device design.

HELLO SMARTPHONES

In June 2007, Apple launched its first iPhone device, a game-changing move in the mobile industry. A brand new, shiny device replaced the old tactile phone. It came with a slick, touch-based interaction, a rich feature set, a better graphical display, and processing power exceeding that of *Apollo 11* when it first went to the moon.[6] A year later, the App Store launch truly brought this device to life, with thousands of apps for entertainment, content, media, and other services (as discussed in the next section). As a side effect, iPhone owners consumed significantly more data than traditional cell phone users, which in turn encouraged mobile providers to offer subsidies on the device, given their compensating revenue from the increased data rates from these customers. All these factors contributed to accelerated adoption of the device.

The Android platform, introduced officially in October 2008, significantly reinforced smartphone adoption, with new kinds of devices from various manufacturers being offered at a lower cost. Soon enough, the face of mobile devices changed completely.

This change sparked (among other transformations) a sharp UX paradigm shift from the dominant design practices of the time, due to the move from tactile-based devices with a full physical keypad to touch-based displays where the entire interaction takes place on the screen itself.

Table 1-1 outlines the main differences in key user experience design aspects, in terms of both industrial design and interaction design.

TABLE 1-1. Comparison between the prevalent mobile device design before the iPhone was launched (left) and after (right)

	BEFORE	AFTER
Industrial design and form factor **(sample of devices)**		
Interaction model	Tactile	Touch
Interface manipulation	Indirect manipulation Users interact with physical keys below (and next to) the screen, through which they control the elements on the screen.	Directly on screen Users interact right with the elements on-screen.
Physical keys	Full physical keypad (usually 21-key–based, including navigation and action keys) and additional side buttons (like power on/off, volume up/down).	One home button on iPhone, three keys in Android, and additional side buttons (like power on/off, volume up/down).
Touchpoints	Supports a single keypress at a time	Supports touching multiple points on the screen simultaneously
Gestures	Supports only keypresses (no gestures)	Supports both button taps on-screen and gesture-based interactions (like swipe, pinch, rotate, and spread)
Orientation	Portrait only	Portrait and landscape
Focus and selection states	Two separate states (can have different visual treatments, and trigger different events)	Both merged into selection state (have a single visual effect, and can trigger only one event for selection)
Short and long press	Supports both	Supports both

These changes required everyone involved in product development and/or consumption to alter the way they approached mobile devices, and to establish new mental models, use habits, and best practices around the new mobile experience.

But the changes didn't stop there.

HELLO APP STORES

In July 2008, Apple launched its App Store for iPhone, and the mobile space was open for business. This move gave third-party developers the long-awaited freedom to build powerful mobile applications for everyone: no longer was the device strictly governed by mobile operators or handset manufacturers, but rather applications were developed by the people, for the people. Everyone could potentially build a mobile application, or develop a mobile version for an existing desktop product. The smartphone was finally recast from being a fancy, high-priced enterprise device (mostly used for calling, texting, and some browsing) into a multi-functional device offering a rich, dynamic world of content, features, and services.

In the first weekend after the App Store was launched, 10 *million* applications were downloaded. In less than a year, 1.4 billion apps had been downloaded, and more than 200,000 apps were available in the App Store.

Soon after Apple's App Store opened, many competitors followed suit, further enhancing smartphone adoption. In October 2008, Google introduced its Android Market. By August 2010, 1 billion apps had been downloaded from the Android market; in 2009 BlackBerry App World, Nokia Ovi Store, and Palm App Catalog opened; in 2010 Microsoft joined the party with Windows 7 Phone marketplace, and so did Samsung with its apps marketplace. Today, even the Nintendo DSi and Sony PlayStation Portable have application stores.

"There's an app for that" has quickly become the new catchphrase, serving as yet another signal of the profound effect (depicted in Figure 1-5) application stores have had on mobile development and design.

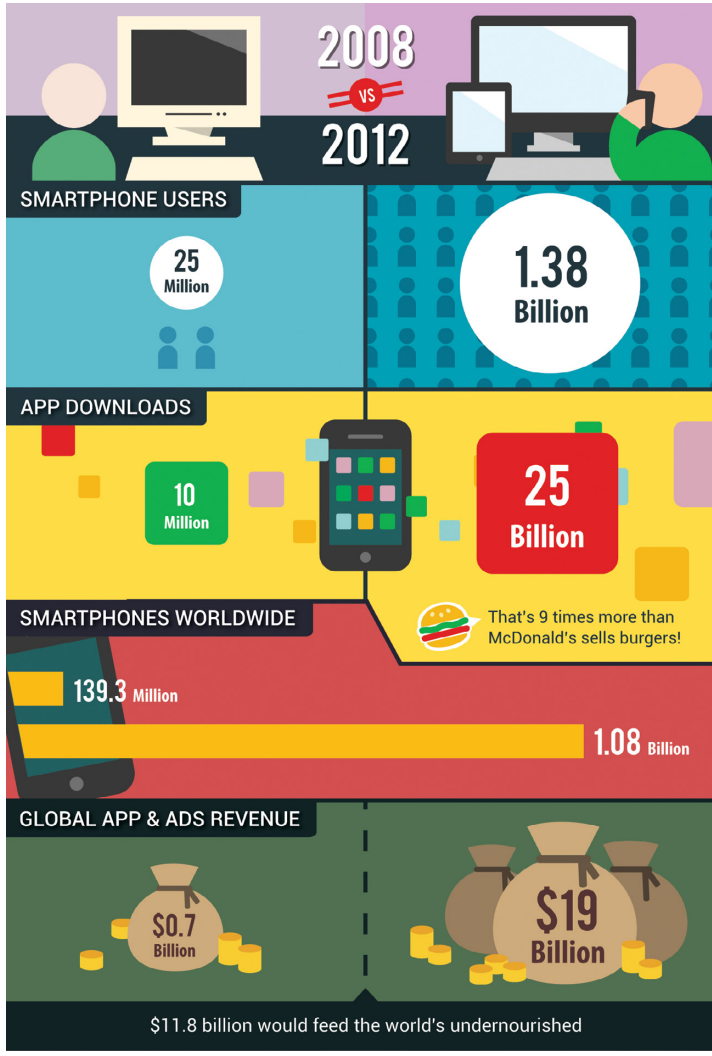


FIGURE 1-5.
Infographic comparing key mobile stats between 2008 and 2012. Can you believe this all happened in less than five years?[7]

In less than five years, mobile phones have practically become an extension of ourselves. Users *love* their smartphones, and use them everywhere—at home, work, the family dinner table, a restaurant, a store, a coffee shop, the doctor's office, the airport, the movie theater, the bathroom, on public transport, on a date, and even at church. If that's not enough, a recent study showed that 15% of iPhone owners said they would rather give up sex than go for a weekend without their iPhones.[8]

Four percent actually reported using their phones *during* sex (probably a topic for yet another book), and 65% reported they couldn't *live* without their devices. (If that's not love, I wonder what is.)

With the rapid success of the smartphone, its wide adoption, and its deep assimilation into our daily activities, it was just a matter of time until new devices joined the party. And they did.

Discussion: On Native Apps, Web Apps, and Everything in Between

For a few years now, whenever someone has considered building a mobile app, one of the debates that has immediately come up is which development path to choose. Native app? Web app? Or maybe a hybrid app?

Before going into the pros and cons of each approach, let's first understand what each of these app types means:

Native apps

These apps live on the device itself. Accordingly, they are specific to the device platform on which they reside (e.g., iOS or Android), and require separate development per platform. These apps need to be downloaded and installed through an application store, and can then be accessed via a dedicated icon on the home screen. Native apps can access and use all the device's sensor data (GPS, accelerometer, camera, etc.), as well as benefiting from native platform components like UI elements (buttons, sliders, tabs, and other controls), interaction patterns (gestures, transitions), and core features (contact list, call log). These apps can also run in the background, work offline, and integrate into the central notification system. Most of the mobile apps today are native.

Web apps

Also commonly referred to as HTML5 apps, these apps run in the web browser. These "apps" are effectively websites (typically written in HTML5) that users access just like any other website—through a URL entered in the device's browser. Upon first connecting to the website, users can install the app—an action that puts the app icon on the home screen (providing similar access to native apps). The difference is that in this case, the icon is a shortcut to the designated URL.

As web apps are run by the browser, they do not require a separate code base per platform; a single instance can be used across platforms.

Visually, web apps can look very much like native apps, and they can incorporate gestures and transitions (as supported by HTML5). At the same time, HTML5 apps do not (yet) match native apps' quality of experience (in terms of performance, smoothness, gesture slickness, etc.), and certain native OS features are (still) unavailable from the browser, such as the central notification system, some sensor data, and advanced gestures.

Hybrid apps

As the name implies, hybrid apps are a combination of native apps and web apps (see Figure 1-6). They are essentially HTML5 apps wrapped inside a native container. They live on the device itself and run inside a native wrapper, which uses the device's browser engine to render the HTML5 code. Just like native apps, they are installed through the application store.

This hybrid structure aims to take advantage of the best of both worlds: keeping a single code base for all platforms (like web apps), and having access to extended device capabilities such as the accelerometer, GPS, camera, and device storage (like native apps).

Development frameworks such as Sencha, PhoneGap, Titanium, and others provide tools to build hybrid apps that can work on iOS and Android, as well as BlackBerry, Windows Phone, and more.

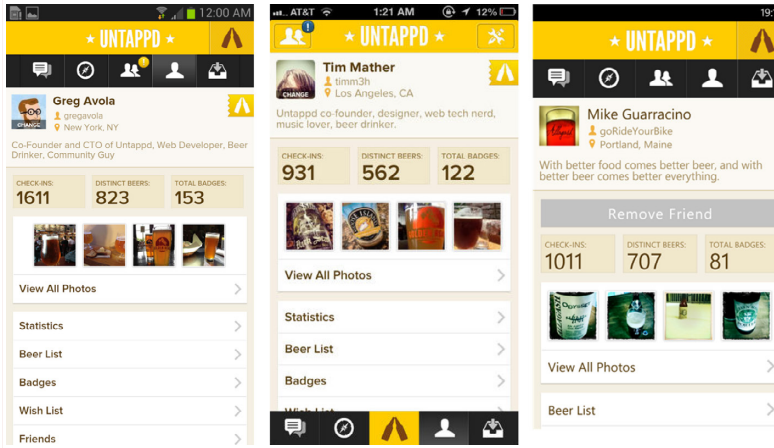


FIGURE 1-6

Untappd (created by PhoneGap) is an example of a hybrid mobile app. It is shown here (left to right) on Android, iOS, and Windows Phone, respectively. Untappd has a native app look and feel, and it even follows each platform's specific UI principles (e.g., the tabs location), strengthening platform affinity.

Deciding which development path you should take depends on multiple factors related to your company, your product, your users, and your product timeline. To help you with the task, Table 1-2 lists nine questions that will guide you through selecting the approach that best fits your needs.

TABLE 1-2. Comparison of native, web, and hybrid apps—the green bar indicates an advantage for that development path, while red highlights a disadvantage

	NATIVE APP	WEB APP	HYBRID APP
Need for speed	Faster graphics and performance	Slower	Slower
Need for device sensors (GPS, camera, etc.)?	Native apps provide access to all the device's features	Access to device is limited from the browser	A web-to-native abstraction layer allows extended access to the device's sensors
Limited development resources?	Requires separate, specialized development per platform	Single HTML5 code base across platforms	Single HTML5 code base across platforms
Crunched for time?	Slower implementation due to the need to develop multiple app variants and get them approved in the app stores	Fastest implementation and deployment	Faster implementation and deployment
Prioritizes premium design? Complex UX design?	Provides slicker, higher-quality, and more personalized UX. Allows use of device-specific gestures.	HTML5 is still more limited in UX capabilities. Might have a very similar look and feel, but inferior interaction.	HTML5 is still more limited in UX capabilities. Might have a very similar look and feel, but inferior interaction.
Large amounts of data transfer?	Faster data transfer	(Still) slower due to reliance on network speed	(Still) slower due to reliance on network speed
Expecting frequent updates?	Requires maintaining multiple specialized mobile variants and distributing the updates through the app stores	App can be changed and pushed out instantly to the entire user base	Requires changing only a single code base, but the distribution is still through the app stores

	NATIVE APP	WEB APP	HYBRID APP
Need the app to work offline?	Supports offline use	Cannot work without connectivity	Supports offline use
Plan to monetize content?	Much better in-app purchasing system also tied to other users' purchases on the platform	No structured support. Can offer a shopping cart on the website, but still doesn't compare to the native app.	No structured support. Can offer a shopping cart on the website, but still doesn't compare to the native app.

Note that the assessments in Table 1-2 are based on the current state of technology and may change with time, narrowing down some of the gaps that exist today and possibly expanding others.

In any case, remember that users don't really care if the app is native, web, or hybrid. What they *do* care about is being able to find the app they need, and that app offering a fast, slick, effective, and delightful user experience that helps them in their activities.

HELLO TABLETS

In April 2010, Apple expanded its family and introduced the iPad to the world. On the first day alone, 300,000 iPads were sold, and sales reached 14.8 million during the first year. As with its ancestor, the iPhone, it was really the App Store that made the iPad truly magical, and sales continued to skyrocket, with cumulative totals reaching 67 million units in just two years.[9]

In parallel, Google stirred up the tablet market even more, releasing Android OS for tablets in 2011 and gaining 39% market share within less than a year.[10] In less than three years, tablets reached 10% market penetration, becoming the fastest-growing device in the history of consumer electronics.[11]

The tablet, similarly to the smartphone, not only created new use modes and behavior patterns around it, but also changed the ways we use existing devices in what I call the *interaction effect*: the usage patterns of one device change depending on the availability of another device. This change can manifest in using the devices more, using the devices less, or using them differently—in conjunction with one another, for example.

In the case of tablets, their increasing use led to a decline in other media and device usage; for instance, 20% of tablet owners are using print magazines less, 25% are using fewer print books, and 27% are using print newspapers less often. Furthermore, desktop computers, laptop computers, ereaders, and portable media players are also used less.[12]

But there was another interaction effect created by tablets, one that demonstrates the strength of (and need for) multi-device experiences. According to a 2011 Neilsen Company survey, 70% of tablet owners used their tablets *while watching TV*—a use scenario that constituted the largest share of their total time with the tablet (30%), as Figure 1-7 illustrates.

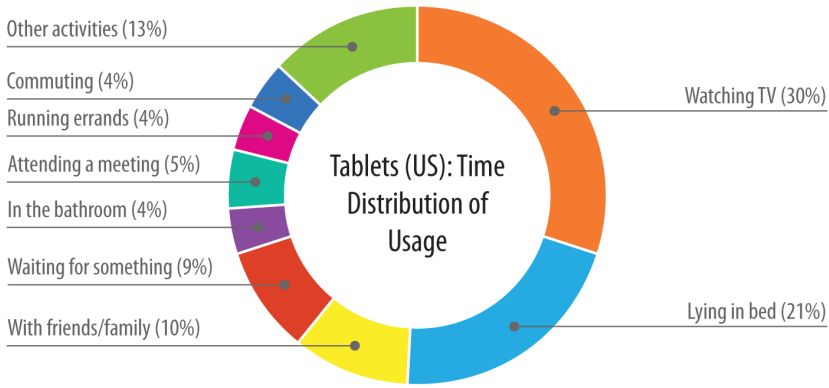


FIGURE 1-7.
Time distribution of tablet usage in the US market.[13]

This made the TV the device most used in conjunction with the tablet, pulling it into the ecosystem as one of its core devices (as well as granting tablets the title of "TV buddies"). Not only that, but this joint usage involved users engaging with the tablet for TV-specific activities during a show: looking up information related to the TV program they're watching (29%), searching product information for an ad (19%), and looking up coupons or deals related to a commercial (16%).[14] These behaviors emerged organically, through users searching for information using a web browser, before there were any specialized apps for that purpose. Can you imagine the UX opportunities embodied in building multi-device experiences that *are* specifically tailored to this use case? Apps that offer users relevant, real-time information and

activities while they're watching TV? Today, we already have an abundance of such apps (also referred to as *second-screen experiences*), and we will further analyze some of them in the next chapters.

At a higher level, this strong tie between the TV and the tablet, as well as other interaction effects taking place between the ecosystem devices, is a highly important data point when we're considering multi-device experiences.[15] These interaction effects provide insights into users' evolving habits, use patterns, and mental models. As a result, they can (and should) help drive product and design decisions regarding key questions like which devices should take part in the experience, how the experience should be distributed across devices, what the integration points between devices should be, and where (if at all) offering a collaborative device interaction is beneficial to users.

It's an Ecosystem!

The three processes just described—smartphones becoming a commodity, application stores gaining speed, and tablets joining the party—all contributed to the emergence of an ecosystem, with four devices at its core: the smartphone, tablet, PC, and TV.

In this ecosystem, a few important principles emerge that will accompany us throughout the book:

- We are in the midst of an important behavioral shift to a multi-device model; product design is no longer just about the desktop platform because there's a prosperous ecosystem of connected devices that complements it, and that continues to grow.
- These connected devices can form a multi-device experience as a connected group (rather than just a set of silo devices). In other words, the ecosystem experience can employ any of the three design approaches—consistent, continuous, and complementary—or a combination of them.
- The ecosystem is not bound to the four core devices. These devices are currently the most commonly used, and thus serve as the basis; however, as more connected devices are introduced, they can join the ecosystem, further expanding the contexts of use and device relationships it accommodates.

- The more we embrace the potential of an ecosystem by adapting the experience per device and building the required bridges between them (acknowledging the different use cases in varying contexts), the more we can simplify the experience on each device, and provide an overall holistic experience that is greater than its parts.

Bear in mind that the multi-device era is still in its early stages. Thus, use patterns across devices are just starting to take shape, and even the ones that seem to stabilize will probably change soon enough, as more devices join the ecosystem and introduce new, disruptive ways to connect and interact with the environment and one another.

At this point in time, the most important goal we should focus on is *learning*. We should explore and experiment with building multi-device experiences that can continuously drive us to create better products, that are more intimately tailored to individual users' changing needs. In this process, we need to encourage the open mind, inquisitive spirit, and broad thinking that are instrumental in taking the leap from a single-device approach to an ecosystem one. As you will see throughout the book, this new multi-device world opens up many new opportunities to innovate—not only by looking into future needs and use cases that will naturally arise, but also through rethinking some of our existing design approaches to current challenges. The latter is where much of the power lies: disrupting widespread perceptions and assumptions regarding what is possible in light of the new ecosystem possibilities we have.

Summary

- Particular conditions brought about the multi-device era and the unique factors that differentiate it from anything we've seen before.
- The multi-device era introduces an ecosystem, similar to that of the natural world. In the multi-device ecosystem, a variety of devices interact with one another as an ecology, and their interactions are shaped by how individuals use the devices in a variety of contexts en route to completing their information and entertainment goals.

- The three key processes that signify the critical transformation from a single-device model to an ecosystem approach were the entry of smartphones as a commodity, the burgeoning application market, and the success of tablets.
- The 3Cs framework consists of three principal approaches—consistent, continuous, and complementary—for handling the design complexity introduced by the numerous devices on the market (and those yet to be invented).

NOTES

1. Cisco.com, "Cisco Visual Networking Index: Global Media Data Traffic Forecast Update: 2012–2017," February 6, 2013, *http://bit.ly/1c0HYnW.*
2. "GSMA Announces That the Proliferation of Connected Devices Will Create a US$1.2 Trillion Revenue Opportunity for Mobile Operators by 2020," PR Newswire, October 10, 2011, *http://prn.to/19cuKWx*.
3. Google, "The New Multi-Screen World: Understanding Cross-Platform Consumer Behavior," August 2012, *http://bit.ly/1gVzizF*.
4. Google/Ipsos OTX MediaCT, US, "Our Mobile Planet: United States," May 2012, *http://bit.ly/1bQ26bo*.
5. Christophe Stoll, "Multiscreen Patterns," May 26, 2011, *http://bit.ly/1fmA87Y*.
6. Richard Stengel, "Making Sense of Our Wireless World," *Time Magazine*, August 27, 2012, *http://www.time.com/time/magazine/article/0,9171,2122243,00.html*.
7. "Number of Smartphones Sold to End Users Worldwide from 2007 to 2012 (in Million Units)," Statista, February 2013, *http://bit.ly/Jx8aM9*; "Strategy Analytics: Worldwide Smartphone Population Tops 1 Billion in Q3 2012," Business Wire, October 17, 2012, *http://bit.ly/1kSjiPn*; Rob Thurner, "The Latest App Download Statistics," *Smart Insights*, March 29, 2012, *http://bit.ly/1hZhib8*; Jeff Sonderman, "Mobile App Revenue Exceeds Ad Revenue," December 4, 2012, *http://bit.ly/1jf2w1o*; Chris Quick, "With Smartphone Adoption on the Rise, Opportunity for Marketers Is Calling," Nielsen, September 15, 2009, *http://bit.ly/1jf2AhI*; "A History of App Stores: Apple, Google, and Everyone Else" (infographic), WebpageFX, August 15, 2011, *http://bit.ly/1bNWZIC*.
8. Sam Liard, "Are You Addicted to Your Smartphone?" Mashable, September 5, 2012, *http://mashable.com/2012/09/05/addicted-smartphone/*.
9. Sam Costello, "What Are iPad Sales All Time?" About.com (based on Apple Quarterly Finance Report), October 23, 2013, *http://abt.cm/1dmA32t*.
10. Robin Wauters, "Android Reaches 39% Tablet OS Market Share 2012," TechCrunch (based on research by Strategy Analytics), January 26, 2012, *http://tcrn.ch/19Swgrc*.
11. Michael DeGusta, "Are Smart Phones Spreading Faster Than Any Technology in Human History?" *MIT Technology Review*, May 9, 2012, *http://bit.ly/1fEBvj0*.
12. "In the U.S., Tablets Are TV Buddies While e-Readers Make Great Bedfellows," Nielsen, May 19, 2011, *http://bit.ly/1khyS9G*.
13. "Double Vision—Global Trends in Tablet and Smartphone Use While Watching TV," Nielsen, April 5, 2012, *http://bit.ly/1cAsIuu*.
14. "State of the Media: U.S. Digital Consumer Report, Q3–Q4 2011," Nielsen, 2012, *http://bit.ly/J5amtF*.
15. Chapter 7 focuses on multi-device data analytics.

[3]

The Continuous Design Approach

> With continuous design, the second approach in our 3Cs framework, the multi-device experience flows from one device to the next. It is the *experience* that accompanies the user in his set of activities, within different contexts, en route to his information or entertainment goal. In the following set of examples, we'll look at the principles of this approach, as well as its use cases, design lessons, and main benefits to users.

What Is Continuous Design?

Continuous design addresses a user flow that runs along a set of contexts, during which devices "pass the baton" to one another until the user reaches her information goal or completes the desired activity (see Figure 3-1).

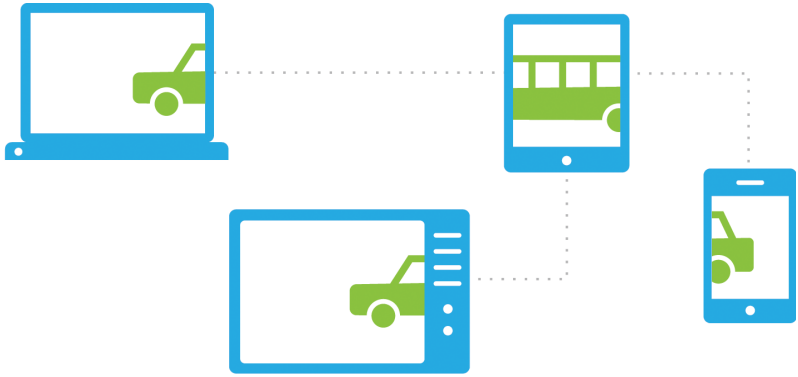


FIGURE 3-1

A continuous approach provides an end-to-end user experience that is distributed across multiple devices. With this "passing of the baton" approach, each device picks up where the previous one left off.

There are two chief experience types that continuous design addresses:

Single activity flow

A single activity—reading a book, watching a movie, or writing a document—typically requires a significant period of time to complete. These kinds of activities, therefore, tend to progress through several contexts (e.g., a laundromat, the airport, or a coffee shop line). And they include the opportunity for a user to use different devices (e.g., an iPad at the laundromat, laptop at the airport, and iPhone in the coffee shop line) and have each pick up the experience where the previous device left off. So, for example, you might start reading *Moby Dick* at home on a Sunday on your tablet and read the next chapter on your smartphone while waiting at the doctor's office.

Sequenced activities flow

Some task flows are composed of sequences of different activities that users need to complete in order to reach the end goal. One good example is cooking. This task is made up of several steps: (1) searching for recipes and deciding on the desired ones, (2) buying the groceries, and (3) cooking. Each step is typically done in a different location, at a different time, with different devices being available and/or most convenient for each activity. Still, all these steps progress the user toward the same end goal. Travel rentals (like with Airbnb) are another good example. They involve pretrip activities like (1) deciding where to stay, (2) looking for a place, and (3) booking. Then there are multiple activities during the trip itself, such as (1) getting to the apartment, (2) settling down, (3) finding places in the area, and (4) keeping in touch with the property owner. Finally, the flow could also involve post-trip activities, such as (1) leaving a review and (2) possibly listing one's own property for rental. Each activity within this broader end-to-end flow differs from the others in type, location, context, duration, and best (available) device for the task, but they all clearly contribute to accomplishing the overarching task.

It's important to note that activity duration is a key factor in determining the continuity flow across devices. Here are three ways in which it does so:

Multiple sessions

The lengthier the activity, the more likely it is that it won't be completed in a single engagement. Instead, users will complete a lengthy activity in multiple passes (planning a trip, for example). In many cases, this involves the user switching between different devices depending upon the context.

Shifting contexts

The lengthier the activity is, the more likely it is that the contextual environment will change. This happens when users terminate a session and pick it up later (the taxi ride is over, so they close down the website they were browsing), even if they hadn't planned to, either on the same device or a different device more suitable to the new context (e.g., revisiting the website later at the hotel).

Subtasks

The lengthier the activity is, the more likely it is that it can be broken down into a set of granular subtasks. For example, with traveling there's the planning, booking, and the trip itself, and the latter can also be split to subtasks—flying, picking up the rental car, getting to the hotel, visiting attractions, and so on. These subtasks, which can be very different from one another, may take place in different contexts (location, time, social setting, and more), and thus be best served by different devices.

Let's look at some examples to clarify these two forms of continuity using the core user screens: smartphone, tablet, PC/laptop, and TV.

Single Activity Flow

Three good examples of the single activity flow are Apple AirPlay, the Amazon Kindle, and Google Drive, discussed next. They demonstrate how this continuous experience type works well for both consumption and creation of content.

STREAMLINING THE VIEWING EXPERIENCE: APPLE AIRPLAY

With Apple AirPlay and Apple TV, users can wirelessly stream movies, music, and photos from any iOS device—iPhone, iPad, iPod, or MacBook—right to the big screens of their televisions (see Figure 3-2).

In other words, they can "hand off" any content they began consuming on their mobile devices to their TVs, and experience higher-quality images and audio than on a smaller device.

FIGURE 3-2

Apple AirPlay technology enables a continuous watching experience across devices to accommodate changing contexts. Users can begin watching a movie on their iPhones or iPads and continue on the big-screen TV right where they left off.

How is the AirPlay/Apple TV compatibility useful? In many ways:

- A user can start watching her favorite blockbuster on her iPhone during the train commute after work, and continue the movie from the exact point she left off on the big HDTV at home, in a smooth, seamless way.
- Teenagers can start watching a TV show on an iPad in their room while their parents watch the evening news on the living room TV. When the news broadcast is over, the kids can finish watching their show on the TV, taking advantage of the bigger-picture viewing experience. They can hang on to the iPad (or grab their iPhones) and use one of those devices for research while watching TV. (We'll look at this experience flow more closely in Chapter 4, on *complementary* design.)
- Morning-music people can listen to their playlists on their MacBooks as they wake up and start their morning routines. After getting showered and dressed, they can continue listening to their music over breakfast by "passing" the music to the TV downstairs, with better sound.

Question to the Reader

Can you think of other use cases for which being able to start consuming media (movie, music, photos) on one device and continue on another streamlines the experience flow?

Try approaching this question from the following pain points and contexts:

- Not enough devices. What if there's one TV in the house, and each family member wants to watch a different show?
- Not enough time. What if you have just an hour, but the movie you want to watch runs two hours?
- Being on the move. What if you have a busy schedule that requires you to frequently move from one place to the next, like organizing the kids and dropping them off at school, having breakfast with a friend downtown, running a series of work meetings (at the office and outside), and going on an evening run at the gym?

The continuous design approach—in all the aforementioned scenarios, as well as the examples we will see next—improves, streamlines, and facilitates an existing set of behavior patterns and needs people already have. In other words, the strength of continuous design is tied to the variety of new smart devices entering the market, which are available in increasingly more contexts, and it should be used to better support users in their activities.

Watching a movie from start to end, even when changing locations, isn't a new need. The fact is that until technology allowed it, people accommodated the need to finish a movie with suboptimal solutions like:

- Watching the entire movie on the same device, even if there was a better device available at some point during the activity
- Waiting until they got home/until the news broadcast was over to watch the entire movie on their HDTVs
- Starting to watch a movie on DVD on their laptops, and then continuing at home using their DVD players, after manually setting the movie to the point where they left off

The new smart devices that people increasingly have within reach give designers the opportunity to rethink existing user flows and use cases, and improve the activity flow among different contexts. We need to start asking ourselves how these new connected devices, especially mobile ones (tablets and smartphones today, and watches, glasses, bracelets, shoes, and many more tomorrow), can be used to help people accomplish their tasks. How can these devices optimize quality, speed, ease of use, and user delight?

Offering the entire or an identical UI on each and every device (i.e., consistent design) is not only suboptimal, but also becoming practically impossible as devices diversify in size (specifically, getting smaller and smaller—up to not having a screen at all), form factor (like wearables), ergonomics, and technological capabilities. The need to define the roles of different devices along the user flow, understand the use patterns around multiple devices, and identify the contexts, actions, and use cases where certain devices are more suitable or convenient than others is now a necessity as we design for multi-device experiences. Remember that down the road, when the Internet of Things disseminates with connected walls, ceilings, desks, bus stations, and whatnot—we will need to consider (again) the following questions:

1. How will these developments impact people's behaviors?
2. How will these new capabilities change the definition of "best available device (or display)" per context?
3. How can we further optimize users' experience along varying contexts using the connected devices?

SEAMLESS CONTENT CONSUMPTION EXPERIENCE: AMAZON KINDLE

The Amazon Kindle ecosystem (Figure 3-3) is another example of a continuous single activity flow. It takes this concept one step further, though, with its Audible.com integration.

FIGURE 3-3

The Amazon Kindle product ecosystem, which enables a continuous reading experience across a variety of devices.

At its core, the Kindle ecosystem allows users a seamless reading experience across multiple Kindles and Kindle apps by way of its Whispersync technology. It automatically syncs books across apps (including bookmarks and annotations), so users can begin reading on one device and pick up where they left off on another device.

This is great for a user sitting at a café waiting for a friend without having the Kindle device at hand—the user can simply pull out her smartphone and continue reading a novel from where she left off while she waits.

But, as I said, the Kindle ecosystem takes things a step further.

By partnering with Audible.com, Amazon expanded the range of contexts that make up the Kindle's continuous experience. Audible's Whispersync for Voice integration allows users to switch between

reading and listening seamlessly, so that there's no need to put the book down even when on the move (running, cooking), or lying in bed at night with the lights turned off. This example shows that a single activity flow doesn't necessarily need to be limited to the same exact continuous action (reading), but it can include additional related actions (like listening) that are all part of the underlying user's activity goal—consuming content. Not only that, but extending the set of contexts in which the product can be used, as demonstrated through the Kindle/Audible integration, enables greater, more frequent use of the product by users, which in turn enhances habit formation.

The Kindle ecosystem exemplifies two important principles:

- **Rethink what's possible.** The constant technological progress continuously affects people's behaviors, forming new needs and habits. Over time, these changes flex the contexts and ways people already use (or can use) your product. Keeping a finger on the pulse and stepping back to reassess the product landscape are instrumental in tapping into these trends and extending the product's use. In this example, the Kindle was able to break out of the (still) ingrained perception of "books = reading," broadening the scope to content consumption. Framing it this way immediately surfaced new opportunities—like audio—that support a richer, more "sticky" product ecosystem.
- **Together, everyone achieves more.** We designers and product creators don't necessarily need to build the entire ecosystem on our own. Powerful multi-device experiences can be the result of integrated solutions, which usually benefit both product owners and consumers. (More about this topic in Chapter 8.)

CONTENT CREATION AND EDITING FLOW: GOOGLE DRIVE

So far we've seen examples of content consumption flows. However, the single-activity continuous experience can also be applied in content creation and editing flows. Google Drive (Figure 3-4) is a helpful model for this functionality. It provides free online storage for a variety of file types—documents, photos, and videos—and enables users to access them from any device.

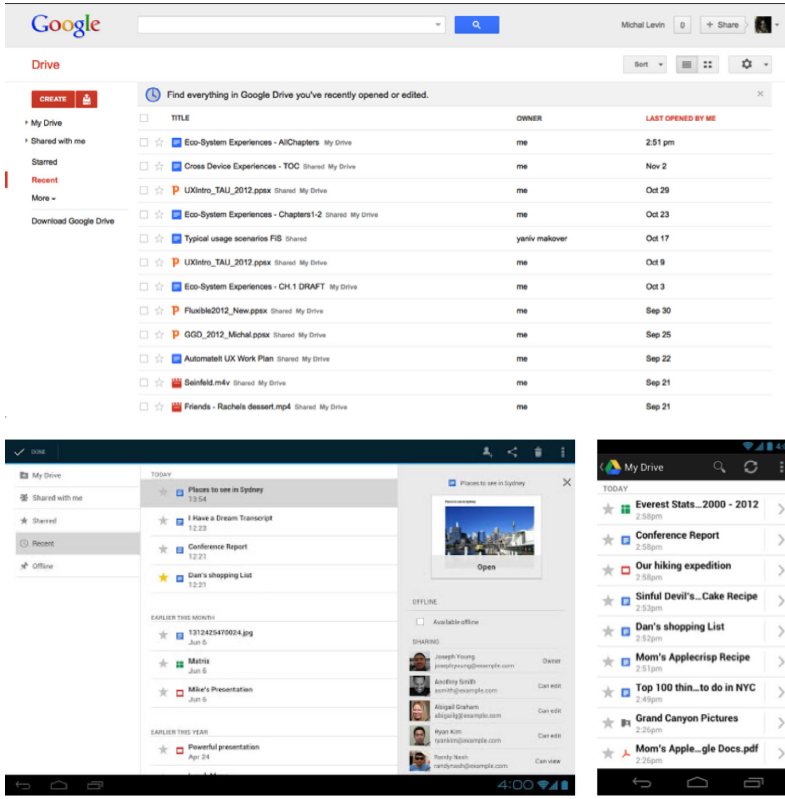


FIGURE 3-4

Google Drive design on three main devices: desktop (top), tablet (bottom-left), and smartphone (bottom-right). All users' files are accessible across all devices, enabling a continuous content creation process between them.

In addition to being able to access content across devices, users can create and edit content across their devices as well. A user can begin writing a document on his home computer, for example, and continue editing it on a tablet right where he left off during his train ride to work. Cloud computing and real-time synchronization between devices are the key enablers of continuous flow of content creation.

THE LINK BETWEEN CONTINUOUS AND CONSISTENT EXPERIENCES

If you look carefully at the three examples just discussed, you'll notice that across the devices, the user can consume the entire span of content on *any* device.

This should sound familiar, as it follows the consistent design approach discussed in Chapter 2.

The difference between those consistent experiences and the continuous experiences discussed in this chapter is that the latter incorporate an additional layer—a context-sensitive one—that adds continuity to the experience. By saving the "last visited" point for content items, users can resume their activity from the point at which they left off (on multiple devices). This can take place automatically upon the user's entering the app (jumping the user right to the last visited spot), or be triggered manually by user action; this is a design decision that depends on the use case at hand.

This consistency/continuity combination in the single activity flow highlights two important fundamentals of the 3Cs framework:

- The 3Cs are design approaches that should be treated as experience building blocks. You can have an experience that is based solely on one of them, or you can integrate several approaches within the same ecosystem experience. In fact, most multi-device experiences follow the latter line, as we will discuss in more depth in Chapter 5.
- The 3Cs framework is focused on the way *people relate to the task flow* and how the different devices support them in achieving their goals—meaning you could argue that the single activity flow is actually a consistent experience. The UI design is practically the same across all devices, with one functional difference in the behavior of content items visited before. You might ask, couldn't that be regarded as merely a consistency adjustment between devices?

  While they might seem similar on the surface, from the user's perspective there is actually a world of difference between the consistent experience flow and the single activity one. In the consistent experience, design adjustments aim to accommodate *device characteristics* (mainly screen size and input method). In the continuous experience, the difference in behavior has nothing to do with the form factor. It's a flow change that accommodates *people's needs*, supporting them in a continuous activity. The fact that it ends up manifesting in a similar way in the UI layer doesn't mean it stems from the same root—just like a computer failing to load can be the result of hardware malfunction, a software configuration problem, or a virus, for example. This distinction is important, as it affects

the way you approach the situation to begin with. So, while the single activity flow relies on consistent design at its core, it's still a distinct experience that provides a continuous flow for users.

Discussion: The Importance of Sign-in

Knowing your users is the most valuable asset you can have for designing powerful, engaging, easy-to-use products. The more you learn about what they do, want, and need—what matters to them and what doesn't; what their preferences, interests, habits, and behavior patterns are; and how all these are affected across different contexts—the better, more personalized experiences you can deliver.

When you're designing multi-device experiences, this need to know your users skyrockets, especially given the dynamic nature of these experiences and the key role context plays. In this ever-more-complex connected world, there are many unknowns and hardly any proven best practices, and use patterns among devices are still in early formation stages. Both designers and users are still figuring out their way through all these connected devices, and there's a whole lot of learning to do in order to create meaningful experiences that advance people to their goals in an optimal way.

Until we get to the stage where we can easily identify the individuals using devices through implicit signals that don't require user action (like individual use patterns, physical/biometric signals, proximity, and leveraging multi-device connections and leveraging multi-device connections—like SlickLogin and Knock), getting them to sign in is an extremely important step in building knowledge around who they are and their usage profile across devices.

The sign-in process (let alone registration) is still a significant conversion roadblock. Users abandon products when they are confronted with a sign-in prompt—whether because they've forgotten their usernames or passwords (or both), they've gotten stuck trying to recover those components, or they're just reluctant to identify themselves due to privacy or security concerns. Add to that enhanced security measures (like two-factor authentication), and getting over the sign-in hurdle becomes even more challenging. Having users do that multiple times across several devices makes the task much, much harder.

And yet, given the critical value of the sign-in step in product flow, sign-in design is one area of the experience design pillars you shouldn't overlook. Here are a few good things to start thinking about sign-in design:

Consider single sign-on (SSO)

This simplifies the sign-in process by using a single account (e.g., Facebook, Google+, or Twitter) to access many different applications (see Figure 3-5). SSO offers several advantages:

- Shortens the sign-up process by requiring fewer details and clicks
- Reduces the number of usernames and passwords people need to maintain
- Gives you access as a product developer to information about users, their friends, and their activity on that service. They will need to approve this access, of course, and in some cases the information might be basic (e.g., name, age, and profile picture), but it's still valuable data that you won't need to collect yourself.[1]

An important design consideration to keep in mind is that with every account login option you add to the form, you complicate your design in terms of decision junctions and visual load.

Furthermore, the more login options are offered, the harder it is for users to remember which of the accounts they used to log in to your service.

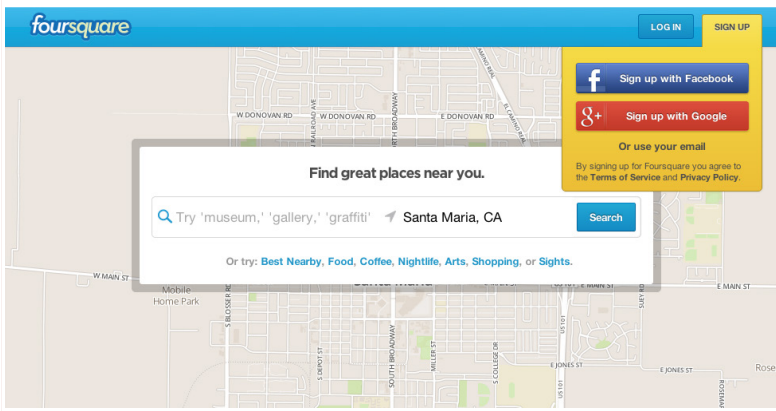


FIGURE 3-5

Foursquare sign-up form promoting SSO with two services, Facebook and Google+. Users can still choose to sign up using their email.

Use simple, engaging forms

Whether you offer SSO and/or your own sign-up process, simplify the form as much as possible:

- Ask for the minimum necessary information (usually email/username and password). Even if you really want to get users' birthdays or genders, wait and ask for that later, after they get a better grasp of the value you're giving them. Users will be more inclined to share information with you at that point.
- Don't make users re-enter information (password, email, or any other field).
- Include a "Stay logged in" option, and have it checked by default.
- Offer a "Forgot?" gateway if people can't remember their usernames/passwords.
- Provide useful, actionable feedback. For example:
  - If a username is taken, offer similar available usernames for people to choose from.
  - Don't overcomplicate the password restrictions, and explain them clearly from the beginning. If people get it wrong, specify the exact error. If all goes well, indicate that as well and make them a little bit happier.
  - Give clear, focused error messages rather than generic ones.
- Be clear about privacy and other conditions.
- Pay attention to tone of voice and copywriting; they matter. The text you supply can provide useful assistance throughout the process, as well as making the experience more fun and lightweight (see Figure 3-6). If you're contemplating several alternatives, try running A/B testing to see which performs best.

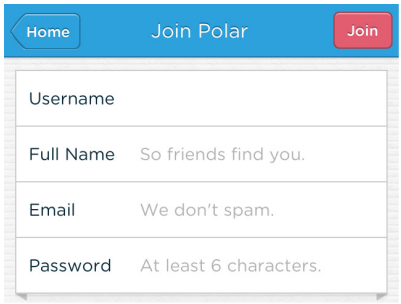


FIGURE 3-6
The Polar mobile app sign-up form, which uses a clear and conversational tone of voice.

Incorporate gradual engagement

An effective strategy to encourage people to take action in your product is to let them get a feel for its benefits before asking them to invest effort (or provide personal data)—similar to how movie producers release trailers to attract viewers to "invest the effort" (and money) and come to the theater. Giving a little teaser to users can help get them on board more easily (see Figure 3-7). I find this approach very useful as a general UX principle, and it definitely applies to sign-in specifically. Luke Wroblewski calls this sign-in flow approach *gradual engagement*—get people to engage with a product and try out its core features before asking them to sign up. This way, people see the benefits of the product through hands-on interaction, following which they are more willing to join in and provide personal information. The activity and information recorded during this interaction can be added to the user's account later, once she signs in.

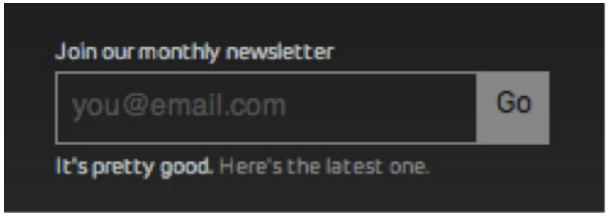


FIGURE 3-7

Sugru lets you look at the most recent issue of its newsletter before you actually join (via stepoffthis-utlingmachine, littlebigdetails.com).

The challenge with giving away something while asking users to sign in is finding the sweet spots in the product flow to trigger the sign-in request. These spots vary depending on your product type.

For example, in the Polar app (shown in Figure 3-6), you can vote on polls without signing up, but to access your activity log, create a poll, or see your profile, you must be signed in. On many ecommerce sites, users are asked to provide sign-in details during checkout (after they've browsed the site and found items of interest). For utility tools, such as design software, asking users to sign in when they're trying to save the work is usually a good practice.

In any case, gradual engagement can—and should—be combined with the previous guidelines as part of your sign-in design strategy. They handle different aspects of the flow, and together can make for a better, more streamlined experience.

Promote the advantages of being signed in in a clear, concise, and attractive manner

Make sure to incorporate the multi-device experience in your message so that people get educated about the broader ecosystem value early on (more on ecosystem education below). You can do this as part of an introductory experience: a marketing page, a warm welcome tour, or some other introduction method. Another approach is to integrate the advantages of sign-in as part of the in-product experience flow, highlighting contextually the benefits of signing in when certain features are used (like "signed-in users enjoy…and can do…"). Another method is to display an "incomplete" UI: greyed-out elements or other visual distinctions in certain areas to encourage people to sign in and "fix it."

As a last note, you've probably noticed there's diverse terminology around the call-to-action label: Sign in, Login, Log in, Sign up, Register, Join, and others. While all these terms are valid, you should avoid using "Sign up" and "Sign in" together in the same product, as the similarity between these terms can be confusing to many users. If you go for "Sign up," combine it with "Log in," for example; and if you prefer "Sign in," then use "Register" or "Join" for the complementary action.

Sequenced Activities Flow

The continuous design approach can be used not only for supporting the continutation of a single action, but also to encompass a variety of user activities—all within a single experience.

In a sequenced activities flow, the experience design addresses broader user needs and goals that involve a series of activities and contexts, and aims to optimize these flows with the help of the multiple connected devices people now have within reach.

As part of the design process, the overarching experience (for example, going out to dinner with friends) is broken down to the user steps that it comprises (like looking for a restaurant, coordinating with friends, booking a table, arriving at the restaurant, etc.). Each step is then analyzed in terms of activity flow, contextual landscape, and the device(s) that can best advance people through that activity on to the next ones and finally to successful completion.

The following examples involve a series of activities that users perform across different contexts in order to accomplish a certain task or goal.

ADAPTING THE EXPERIENCE TO DEVICES' STRENGTHS: ALLRECIPES

Allrecipes (Figure 3-8) is a good example of exploiting the continuous approach by encompassing a broader, multi-step use case like cooking. It demonstrates how multiple devices can support the user every step of the way in a tailored, contextual fashion.

FIGURE 3-8

Allrecipes apps for iPad (Your Kitchen Inspiration) and iPhone (Dinner Spinner), demonstrating a continuous cooking flow made up of sequential activities taking place on different devices: (1) looking for recipes, (2) getting the groceries, and (3) cooking.

Under the hood: Examine the user's workflow

First, let's look at the "cooking" workflow and break it down into the discrete steps that it comprises. For each step, we'll identify the contextual elements involved and see how certain devices are more suitable aids for different portions of the experience. The steps are:

1. Decide what to cook.

   Often, the story begins with a user's basic need to cook dinner for friends. The first step is for the user to decide what to cook. This is where he takes the time to explore recipes—what is usually called the "research" or "seeking information" phase. This research is often done in front of bigger-screen devices like a PC or a tablet, which allow for better content consumption and information scanning while the user engages with the device in a relaxed way.

   For this, Allrecipes provides a desktop interface as well as an iPad app where the user can conveniently look for recipes. What's convenient about it? The bigger layout, easy navigation, search functionality, and easy-to-read recipe display provide all the details he needs: ingredients, dish type, time and ease of preparation, and ratings.

2. Shop for groceries.

   Once a user finds the recipes he'd like to make, he can add them to a "recipe box." This fires up the continuous experience torch: adding recipes to the recipe box automatically extracts the ingredients from all the recipes, syncs the list with his iPhone, and creates a nicely designed shopping list already grouped by dairy, produce, and so on (aligned with most grocery store aisles).

   This activity automatically shifts the experience to the smartphone, which is the device the user is probably going to have on him at the grocery store. There's no need for him to write down all the ingredients or worry about forgetting anything once he's at the store, as the experience seamlessly flows between the relevant devices, with all the content he needs.

3. Cook the meal.

   The user has done his shopping and returned home. When it's time to start the actual cooking, he'll switch from his smartphone to, very likely, a tablet (tablets have become popular as a cooking aid in the kitchen).[2]

   The Allrecipes iPad app provides a very comfortable UI for that—a nice, clear layout with the list of ingredients, the directions to follow (including highlighting the current step the user's on), and even a built-in timer to use during the cooking process. All that's left is for him to follow the instructions and not burn anything!

Design lesson: Rethink user flows

Allrecipes introduced a new way to approach the flow of cooking, taking into consideration the various steps involved and customizing the experience by leveraging the particular strengths and use modes of each device—from searching on the PC or tablet, to shopping for required ingredients with the smartphone, to the actual cooking using the tablet.

Throughout the last several years that I've been researching and working in this multi-device space, I have found that the Allrecipes example is the strongest generator of "aha!" moments from people to whom I'm explaining the continuous design approach. Its emphasis on a fluid, innovative user experience (supported by a strong active community) has paid off. In November 2011, Allrecipes surpassed the 10 million downloads mark. Its two apps—Dinner Spinner (for smartphone) and Your Kitchen Inspiration (for tablet)—continue to be the world's most downloaded recipe apps.[3]

Design lesson: Break down the continuous experience steps even more

When I came across the pioneering work Allrecipes had done with the cooking use case, it got me thinking about additional possibilities for how this product ecosystem can grow further in the near future. Here are a few examples for additional use cases at each of the different cooking steps, which can benefit from dedicated multi-device attention:

I want to be a master chef!

Cooking fans don't just browse the Internet for hot new recipes: they also get inspired by cooking shows on TV. Recently, some of the most popular ones have been reality shows like *MasterChef*, *Top Chef*, and *Hell's Kitchen*. What if Allrecipes could integrate with such shows, so that while watching *MasterChef* on TV, users could save the full recipe for a dish they really want to make? With one tap in the Allrecipes app on a smartphone, they could send the full recipe details automatically to the recipe list on Allrecipes. This could fulfill that immediate gratification desire and allow people to act on the spot, while they're engaged and excited about a delicious-looking pie.

Remember the milk

As we saw, Allrecipes generates a shopping list containing all the ingredients needed for the selected recipes. That's great. However, what it *doesn't* take into account are the ingredients users already have at home. Thus, one might not need to buy all the things on the list, or at least not the entire quantity appearing in the recipe. What if the fridge and kitchen cabinets could be equipped with sensors and network capability (a scenario that is not so far-fetched, as we'll discuss in Chapter 6),[4] so they could inform the Allrecipes shopping list what users already have? The list could adjust itself accordingly and truly represent what users need to buy at the grocery store. And hey, if it could also connect to the grocery store systems, we could inject some more continuity here by having the shopping list ordered and delivered to people's homes or available for them to pick up at the store.

Show me the money

Another enhancement Allrecipes could add to its smartphone app is support for in-store product comparison. We already see today that smartphone owners rely on their devices in-store to make the most of their shopping choices: 73.9% compare prices among retailers, 61% research the items, 39% claim mobile coupons, and 80% want more mobile product information in stores.[5] Wouldn't it be nice if people had a one-stop shop for that? They would just need to fire up the Allrecipes app on their phones to get access to all the price-comparison information they need.

Who moved my cheese?

If we break down the actual cooking process to more granular steps, we get the following cycle of actions:

a. Read the current step's instructions.

b. Get the required ingredients for it.

c. Complete the step's instructions.

d. Move to the next step.

Expanding on the notion of connected devices like refrigerators and cabinets, wouldn't it be useful if those compartments could communicate with a tablet, learn which cooking step a user is on, and shuffle around the items they contain so that the ones the user needs at any given moment are most accessible? Imagine yourself in the following user scenario: you're baking a cake and you get to step 4, where you need to whip the cream. You open the fridge door and—ta-da!—the cream is just in front of you, top shelf, within reach. This interaction could actually change people's behavior because they would no longer need to first take out *all* the ingredients involved in the baking in order to avoid the stressful moments of "where did that bag of cacao disappear to?"

Once we start playing with the idea of all these devices being able to talk to and inform one another, the resolution of user flow steps we can optimize gets finer and finer. This enables us to rethink behaviors and flows we might have taken for granted, dissect them to their most basic components, and really make the best, most tailored user experience for each.

ADAPTING THE EXPERIENCE TO CHANGING NEEDS: EVENTBRITE

Eventbrite (Figure 3-9) is another good example of contextual thinking through a continuous use case. This product, focused on organizing and attending events, shifts the experience across devices as users progress through time. Put more accurately, *the user's contextual needs change* through the sequence of an event.

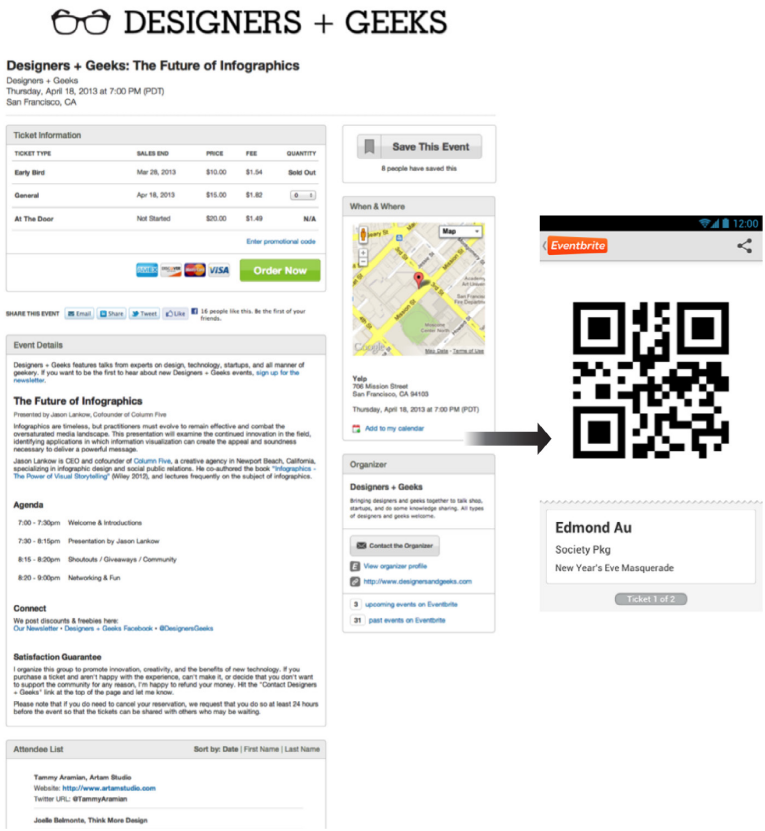


FIGURE 3-9

Eventbrite's example event page—at left on the desktop (before the user signs up to an event), and at right on the mobile app (after sign-up).

Eventbrite actually offers two sets of continuous experience flows—one for event organizers and one for event attendees. As most people more often belong to the second group, we'll examine that particular flow.

Under the hood: Examining the user's workflow

Let's follow the same process as we did for Allrecipes, and break down the broad events use case to the different steps that compose it:

1. Register for an event.

   The event flow is often triggered from an event invite over email, or casual browsing (through social networks, the Eventbrite website, blogs, etc.) that surfaces events of interest. At that point, the "research" or "information seeking" stage starts (similarly to Allrecipes), where

the user reads more on the event, checks the agenda and attendees, and possibly browses through other events while she's at it. This activity is more suitable for bigger-screen devices like the tablet or desktop, which can accommodate more information in a convenient layout. The desktop is also likely to be where the user acts upon the invite (especially if the event is professional), as she does most of her email activity on this device during the week.[6]

The need for a comfortable display for browsing event details intensifies as the event increases in length.[7] For example, a conference that spans a few days—offering multiple sessions per day with many taking place in parallel—requires much more exploration and research time. People need to review many more sessions, pick the ones they would like to attend (especially when sessions are scheduled simultaneously), and go over a longer list of attendees, identifying people they would like to connect with. In the case of a professional conference, the latter is of special importance since networking plays a very important role in going to the event. In fact, for many people, the opportunity to meet and get a conversation going with key people relevant to their work *is* the main goal of attending an event—a point I'll talk more about later.

2. Attend the event.

   Sign-up is complete, and the day of the event has finally arrived: the user is all set, ready to go on her way, and the device she most likely carries is her mobile phone. Understanding this context of use, Eventbrite offers a mobile app that picks up the flow of events:

   a. **Getting to the event.** The mobile app provides quick access to the event the user has signed up for, including the event information along with the location of the event. She can immediately pull up directions on her phone, and have the GPS navigate every step of the way.

   b. **Checking in.** When arriving at an event venue, users often need to register at the entrance. For that, Eventbrite offers a barcode that is easily scanned for check-in to the event. The user doesn't need to go through the hassle of printing the event invitation beforehand and remembering to bring it along—she just uses her phone, which is already there with her, and smoothly slides into the flow of it all.

Eventbrite is another example of a multi-step activity that ranges across different contexts. Across these contexts, peoples' needs change and the devices they use change, so designing an experience that admits these variations and adapts to them can provide much higher value.

Design exercise: Expanding the continuous flow

I've talked in this chapter about how the continuous design approach provides designers with the opportunity to rethink user flows, dissect them to component steps, and explore new ways to help people achieve their goals—all by leveraging the set of connected user devices. Now, I'd like to use our Eventbrite example to do a deep dive into rethinking *process*. Let's now explore the ways multi-device experiences can better meet user needs. Taking this deep dive doesn't mean all the ideas we generate are implementation-worthy (that would require more data, research, and assessments). Yet this is still an important design exercise that will help us to redefine the product landscape, identify areas for innovation, and prepare early for what's ahead.

We will do this exercise on two key types of events: conferences and concerts. Despite the different nature of these event types, there's still a unified theme between them: the social- and people-centered experience at their core. With conferences, for example, the events are aimed at bringing people with shared interests together, facilitating conversations among them, and hopefully taking some of these conversations beyond the geography of the conference venue. There is a lot of potential in addressing ways we can promote this social scene at events—and the multi-device ecosystem can play an important role in enhancing that.

Conferences. Let's look first at the conference event type and break it down into the main set of contexts involved. For each, we'll try to categorize user needs based on the following dimensions (see Table 3-1):

Agenda-centered
: What are the users' needs in terms of event logistics, facilities, and schedule?

People-centered
: What are the users' needs in terms of social interactions and networking?

Multi-device focus
: What are the dominant devices for this context?

TABLE 3-1. Analyzing the conference experience flow across shifting contexts and multiple devices along the agenda-centered and people-centered dimensions

	AGENDA-CENTERED	PEOPLE-CENTERED
Pre-event	**Big screens (mostly desktops)**	

	• Finding events
	• Exploring event info
	• Registering for an event
	• Choosing sessions (if relevant)
		• Sharing event with peers/friends and encouraging them to join
		  • Reviewing attendee list
		  • Getting a sense of the crowd and noting people to meet (possibly even contacting them in advance to set up a chat during the event)

Arriving at the venue on day 1	**Mobile devices (mostly smartphones)**	
	• Checking the event schedule • Getting to the event on time (transportation, directions, traffic, parking) • Checking in • Getting coffee to start the day	• Determining who's here • Anyone I already know? • Any of the people I noted?
During event[8]	**Mixed devices (stationary = laptop/tablet; on the go = mobile)**	

	• Checking the event schedule
	• Exploring events info
	• Attending sessions:
	  • When, where, and how to get there
	  • Session and speaker info
	  • Taking session notes
	• Taking breaks:
	  • Options (cafeterias, exhibitions)
	  • Time left until breaks/"break about to end" alerts
	• Attending social events:
	  • What, when, and where
	  • How to get there
		• Making new connections (many contexts):
		  • Meeting new people, especially those I want to talk with
		  • Catching up with people I already know
		  • Exchanging business cards
		  • Taking quick notes about people I spoke with for later use
		• Attending sessions/social events:
		  • Who's going?
		  • Who's here? (Can impact seat choice.)
		  • Basic professional info
		  • Conversation starters (e.g., common interests, background, recent articles published, recent updates)

Post-event	All devices as a starting point (data analytics can help determine which devices are needed)[9]	
	• Organizing event materials (slides, videos, books) • Posting photos/videos from the event (best moments, the event at a glance) • Researching related events	• Following up with people

Table 3-1 demonstrates how, by stepping back and taking a fresh look at end-to-end user journeys through the lens of multi-device experiences, you can identify important themes about your experience flows and rates and functions along the following scale:

- Well supported
- Supported, but can be improved
- Not supported at all

For example, two prominent design themes that arise in Eventbrite's conference experience are:

- It mostly focuses on the pre-event and event-day parts of the flow. Some features are well supported, while others could possibly be further optimized.
- It mainly supports the agenda-centered activities along the flow. The social needs receiving attention are only the pre-event needs (ability to see attendees list and share the event on social networks).

Identifying these gaps and opportunities is an important first step in setting the product's direction going forward. This doesn't necessarily mean that Eventbrite needs to deepen its design efforts in the social scene, or add other features just highlighted. However, it is important to make these decisions when you have all the information in front of you, after reassessing the product landscape given the new technological developments, and specifically the new multi-device opportunities. Next, we'll look at concerts.

Concerts. At a very basic level, concerts are similar to conferences in the sense that they involve both "logistical" needs and social needs that accompany the main set of contexts involved. However, concerts have an additional emotional layer to the event experience: individuals listen to songs performed right in front of them by the artist, while absorbing the energy of the crowd around them, dancing, singing along, taking photos, and more. It's a powerful experience. Capturing that emotional connection, and finding ways to allow people to extend that just a little bit longer, can be very impactful. Let's break down the concert experience as we did the conferences (Table 3-2).

TABLE 3-2. Analyzing the concert experience flow across shifting contexts and multiple devices along the agenda-centered and people-centered dimensions

	AGENDA-CENTERED	PEOPLE-CENTERED
Pre-event	**Big screens (mostly desktops)**	
	• Finding concerts • Exploring concert info • Buying tickets	• Determining which of my friends are going • Organizing a group to go together • Sharing concert with peers/friends, encouraging them to join, showing off
Event day—we're on our way!	**Mobile devices (mostly smartphones)**	

	• Getting directions
	• Parking:
	  • Help finding a spot is cool.
	  • Finding a cheap one is even cooler.
	  • Saving my location to take me back there later is legendary.	

Arriving at the venue and during the event	**Mobile devices (mostly smartphones)**	

	• While in line:
	  • How much longer will I be in line? (Will I miss the beginning of the concert?)
	  • I'm bored and somewhat anxious, but trying to keep my energy up. (How about a second-screen experience while waiting—fun facts, artist videos, stuff to keep that energy pumped up?)
	• Getting through the ticket check
	• Showing me a detailed concert lineup
	• Setting a five-minute alert before artists go on stage
	• Inside the venue:
	  • Help me find my seat
	  • Food (what, how much, and where?)
	  • Restroom location
	  • Merchandise (what and how much?)
	  • I already have my phone up to take pictures/videos during the song; wouldn't it be cool to see the song lyrics too so I can sing along confidently?
		• Determining whether my friends are here (even if our seats are far apart, we could at least meet later for a beer and a hot dog)
		• Tweeting and checking tweets about the concert
		• Adding people I meet to my social networks

Post-event	**All devices as a starting point (data analytics can help determine which devices are needed)**	
	• Posting photos/videos taken during the event (best moments, concert at a glance) • Receiving a special thank-you note from the artist, along with a video from the concert • Leveraging post-concert excitement to promote the artist's CDs, merchandise, and online presence • Sharing recommendations for related events	• Contacting new people I've met • Creating shared memories with my friends[10]

As with the conference analysis, in a concert there are clearly experience areas that can benefit from a more granular multi-device treatment, mainly in the social space, but also in some agenda-centered aspects. For example, providing a second-screen experience while people are waiting in line can be a very interesting avenue to explore. The smartphone has already become a time filler for people standing in line, so why not leverage this behavior pattern to get them even more excited about and engaged with the concert that's about to start?

Not only that, but second-screen experiences can (and already often do) incorporate a social layer to encourage group engagements. In the case of concerts, an app can be used to enhance connections between people standing in line together, who are already sharing a real-life experience—even if just for a few hours.

There is always room for pushing continuity a step further, reinforcing the idea that if we step back and expand our design lens to encompass the end-to-end user flow and acknowledge the range of connected devices people use, we can find new ways to optimize their experiences. This kind of optimization can stem from dissecting the existing flow into smaller pieces, or expanding its boundaries to include more activities that couldn't have been supported as well before all these devices came into users' lives.

Discussion: Educating for Continuity

Multi-device continuous experiences introduce new types and granularity of flows and use patterns. Most people are not yet familiar with them; their mental models are still anchored in consistent experiences that focus on access anywhere rather than contextual continuity, and they might not even have clear use cases in mind for these device relay races.

To tap into the potential of the new connected world, as designers we need to help people make the cognitive leap and expand—or even break—the existing mental models around multi-device experiences. Our responsibility is not only to build awesome, innovative ecosystems for our users, but also so they equip them with the tools and guidance to engage with these experiences, to feel empowered and in control.

This is where product education plays an important role. In researching for this book, I found myself doing *a lot* of active work to identify continuous product examples out in the market. Current ecosystem experiences do not offer descriptions in the app markets, multi-device tutorials, or in-product guidance; people are left to decipher the multi-device experience layers on their own.

To get our users onto the multi-device–experience bandwagon, we need to first make them aware of it, and—better yet—help them understand and get excited about the value it provides. How do we get them on board?

SEEDING THE ECOSYSTEM NOTION: FIRST-TIME EXPERIENCE, APP MARKETS, AND PROMOTIONS

A good opportunity to make people aware of the multi-device ecosystem and its very basic flow is the first-time experience. Whether you implement it through a marketing landing page, an introductory splash page, or a quick welcome tour, you can use the first to introduce people to the idea of your multi-device structure. A few important principles to remember:

- Users have a lot of experience using other products and services. So even when they start using yours, they have more experience with all the others. Their expectations about the experience you offer will be shaped by that.
- At the point of entry, users know very little about your product (if anything). They probably haven't used it yet and don't have a feel for the use cases and the value it offers, and thus there's a lot they can absorb and internalize at this initial stage.
- Users skip copy and instead scan for meaningful headings, highlighted words, bullet lists, and short paragraphs.[11]

Given these facts, it's important to keep your ecosystem messages short, simple, and concise. Incorporating meaningful visuals (e.g., a visual flow of devices with their roles as headings, or just a group of devices as part of a bigger message) can help you convey the message (see Figures 3-10 and 3-11).

FIGURE 3-10

TripIt website landing page, incorporating devices in a prominent manner. This visual introduces (part of) the ecosystem, demonstrates the value of the continuous experience, and shows a glimpse of the role each device plays.

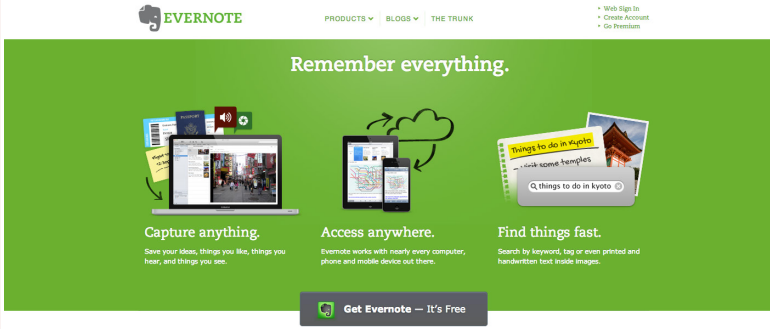


FIGURE 3-11

Evernote website landing page, incorporating a multi-device image as part of a broader product message. The image, along with the accompanying text, clearly indicates that this product spans multiple devices.

Taking a step back in the setup flow, you can already utilize the app markets (subject to their specific terms) as part of the user education process. In all the continuous products I came across, however, there was no indication in the app description about other devices taking part in the experience.

For example, Allrecipes simply released two separate apps to the market (Dinner Spinner and Your Kitchen Inspiration) without any messaging about their relationship to each other. It missed an opportunity to wrap users' minds around the ecosystem flow and continuity value before they download the products. Not only could this encourage people to get the apps, but it could also frame their product perception beyond just a specific device. Explaining the cross-device relationships from the very beginning (in each of the device pages, along with a link to the corresponding device) can help establish expectations and use patterns around multi-device experiences.

You can also seed the ecosystem concept through occasional promotional messages that bring product features to the surface. For example, you can display a promotional modal dialog at a certain frequency when the user enters the app (or accesses some other activity-related trigger). These can be integrated alongside the product content, such as in the notifications area or content stream. You can also use common gestures—such as shake—to trigger product education messages, as Google Maps does (see Figure 3-12). Make sure you select a gesture that is not too complex (undiscoverable) or too easy (triggered unintentionally too often).

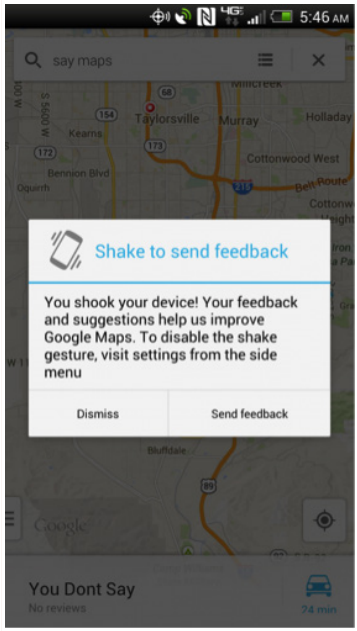


FIGURE 3-12
"Shake to send feedback" message triggered in Google Maps when the user shakes the phone.

INGRAINING THE MULTI-DEVICE VALUE: IN-PRODUCT FLOW

After the first-time experience, which seeds the idea of device continuity, the next (and more important) part is integrating continuity cues within the product flows.

In many ways, the challenge here is similar to that of gradual engagement in your sign-in strategy (discussed earlier in this chapter): it's all about finding those sweet spots during the flow where it makes sense to notify the user about "passing the baton" to the next device. When it's done contextually, after the user has just completed a certain relevant action that reflects a clear intent or state of mind, it is much easier for the user to understand the value of the cross-device continuity, as he's already engaged with the experience.

For example, if we go back to the Allrecipes cooking use case, the first time the user adds recipes to his recipe box on a desktop or tablet, we could show a modal dialog notifying him about the option of (and benefits to) automatically creating a shopping list on his phone. This dialog can ask for missing user details needed to complete the operation (gradual engagement). Also, incorporating an option to send a text message or email with a direct link to download the app to his phone would shorten the path to conversion even more. Finally, offering a checkbox to "Always create shopping list on my phone" could help streamline the flow for future cases, preventing recurring pop-ups from breaking the user flow.

In other cases, like with TripIt (shown in Figure 3-10), delivering a pop-up message following a trip booking might not be the optimal timing. When it comes to traveling, there might be a significant time gap between when the user books her trip via her tablet, and when she actually departs for the trip (with her phone). In this case, alerting the user one to two days before the trip about the mobile phone app and its advantages would better suit her state of mind. Also, on the trip day itself (assuming she has the app), triggering a phone notification a few hours before her flight with its details and status would be useful. As the trip progresses and different needs arise (hotel, car, attractions, etc.), it would be useful to balance phone alerts (for the highest-priority issues) with in-app UI notifications (which can wait until the next time the user logs in to the app).

A relevant example for contextual in-app notifications comes from Pocket. After copying a URL and opening the Pocket app, the user will automatically be prompted to save the content (see Figure 3-13).

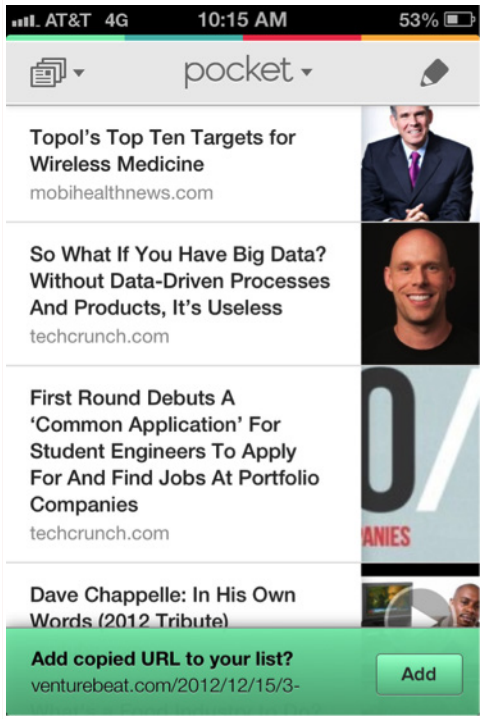


FIGURE 3-13
Pocket mobile app notification after a user copies a URL (via Myles Tan, littlebigdetails.com).

Contextual triggering can also take place based on additional signals, like location and time. For example, in the Allrecipes case, when the user finishes shopping at the grocery store using the mobile app (app usage + time + location) and returns home (location), the phone app can trigger a notification recommending that he switches to the tablet app for cooking. Given that tablets are not yet as widespread as mobile phones, it might be useful to offer an option of "Thanks, but I don't have a tablet," so that users are not annoyed with repeated tablet-related messages.

The important takeaway here is that we must think carefully about how to approach users *while they're engaging* with a task-related activity (cooking) in order to increase the value of helping them very tangibly with the next step in that task flow (grocery shopping). The benefit is not only educating users about the continuity flow as they go, but also encouraging them to engage with additional devices (possibilities they might not be aware of) and thereby advancing them toward their goals.

BRIDGING PHYSICAL AND DIGITAL: POP

POP (Prototyping on Paper) offers designers an ecosystem experience that aims to simplify and accelerate app prototyping in order to make the overall design process quicker and more effective. It enables users to more easily make the leap from wireframes on paper to a live prototype on the actual mobile device, providing a better feel for the interactions and overall flow across the app.

POP's continuous experience introduces a flow that differs from what we've seen so far. While this chapter's previous examples relied solely on wireless connections to generate the ecosystem flow, this product is different: it uses the smartphone camera as a key player in driving the sequence of activities forward.

The sequence starts with paper sketching and then moves into the digital space, where the dynamic prototyping is applied (see Figure 3-14).

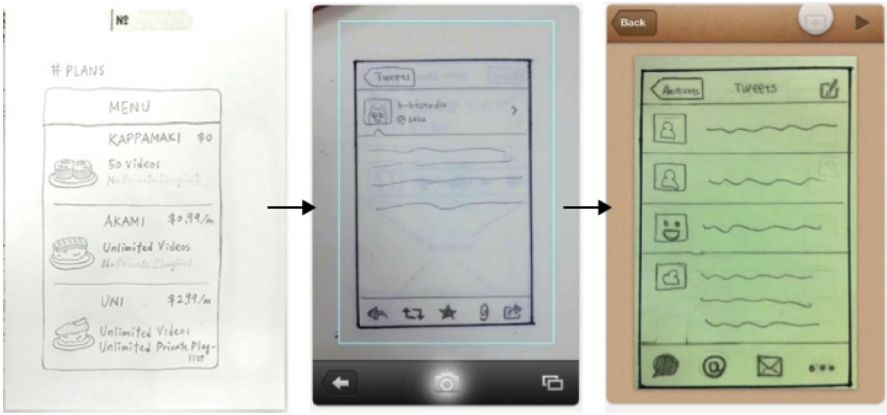


FIGURE 3-14
POP app continuous flow: from pen and paper to the mobile device, connected through the camera.

Under the hood: Examining the user's workflow

As with the previous examples, we will break down the experience flow into the user steps that compose it:

1. Design on paper.

   The first step of the process doesn't involve any ecosystem device or special technology: it's all about the designer following his regular workflow, starting off with creating the initial wireframes using pen and paper—a much faster process than using a software tool.

2. Create a prototype.

   The second step is to create a live prototype out of the sketches, composed of a full storyboard that shows the interaction and flow.

   To do that, the POP app lets you import the paper sketches into your phone by taking a picture of each of them. The app automatically adjusts the brightness and contrast to improve the sketch legibility and ensure the clarity of the simulation.

   Once the sketches are on the phone, you can easily create a storyboard by linking the sketches together through "hot spots" definition. When all the sketches are properly linked and ready to go, you can hit Play and simulate the UI flow from start to end.

3. Share for feedback.

   The last part in the process is sharing the prototype with friends or colleagues to collect feedback early on in the process and discuss it collaboratively. This capability is also incorporated into the app, which provides a built-in sharing mechanism that supports collaborative review across the three core devices: iPhone, iPad, and desktop (through the web browser).

   This entire process can then repeat itself iteratively—based on feedback received, you can create new paper sketches, get them on the device, link them together, and share new prototypes with your reviewers.

   Once the experience is ready for the next step, you can use more advanced software design tools like Photoshop, Fireworks, or Illustrator to build detailed, polished experience screens.

Design lesson: Continuous experience can start offline

What I like so much about the POP app—beyond the fluid continuous experience—is that it admits to the power of the oldest tools around, pen and paper. Instead of trying to come up with ways to beat them, it joins them. POP acknowledges that sophisticated technology is not always needed every step of the way; at times it's better to just stick to what we already do well offline (like sketching on paper), and leverage that. This allows us to really focus on the parts of the process that *don't* work as well (such as the difficulty involved in setting up a live prototype), and mitigate the effort and resources involved in getting those tasks done.

Note that in the case of transferring the experience from the physical world to the digital one, an explicit user action is required (in this case, taking a photo) to clearly connect the experience flow between the different touchpoints. Thus, the continuity between devices, embedded in the usage flow and done on the spot, doesn't require additional user education in this respect.

In the following chapters, we will see more examples of contextual experiences (continuous and complementary) where the multi-device design combines and helps to bridge the physical and digital spheres.

EXPANDING CONTINUITY WITH AN OPEN PLATFORM: POCKET

Pocket (formerly Read It Later) is another interesting example of a continuous experience that addresses a more focused, iterative flow of sequenced activities (see Figure 3-15). It demonstrates a close attention to the varying contexts in which people consume content across different devices, and shows that this kind of attention can really matter when you're creating a seamless, fluid experience tailored to user behavior patterns.

FIGURE 3-15

Pocket product ecosystem—desktop, tablet, and smartphone—across iOS and Android.

The basic premise of Pocket is to help people save interesting articles, videos, and more from the Web for later enjoyment. When Pocket users come across interesting content (articles, videos, or web pages) but don't have time to read or watch it at that moment, they can save the URL. Pocket makes clever use of continuity by tailoring the UI across the different devices, splitting the content discovery and content consumption roles between them. Let's take a closer look at the flow.

Under the hood: Examining the user's workflow

Pocket's experience flow can be broken down to two main steps:

1. Find interesting content.

During the course of a day, people use their computers for long periods of time—doing work, reading emails, checking favorite news sites, visiting their social networks, and searching content. As part of these activities, they often come across interesting content they would like to take a deeper look at, but cannot at the moment for one reason or another (e.g., they're at the office, running from one meeting to the other; they are at a break between classes at school; or they simply don't have the peace and quiet required to sit back and focus on the article or video from start to end).

A similar scenario can take place on the smartphone as well. While doing some quick phone browsing in the lunch line, a user might spot an interesting article that has been retweeted; she can't read it right then and there, so she adds the article to her list on Pocket.

Pocket's solution is to help users save content for later—with a single click. They do this using an online "pocket" icon integrated into the browser toolbar (Figure 3-16).

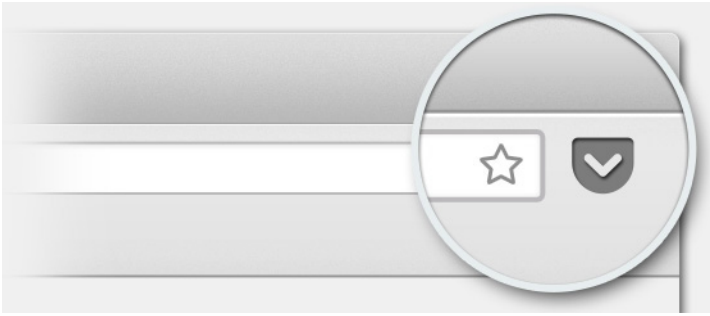


FIGURE 3-16
The "Save to Pocket" button integrated into the desktop browser.

Also, Pocket is integrated into many content apps such as Twitter, Flipboard, Zite, and gReader, and this allows for an easy save from many contexts (Figure 3-17).

FIGURE 3-17

"Save to Pocket" buttons integrated into popular content-based mobile apps.

In the background, all the saved content items are synced across all devices that have the Pocket app installed.

2. Read the saved content.

   Whenever users have some quiet time to catch up on all the stuff they didn't have time for during the day—at the end of the workday, during the train ride home, or maybe as bedtime reading—the tablet or smartphone takes over the experience, continuing the flow started during the day.

   Pocket's app design for both devices is tailored to the context of content consumption (see Figure 3-18). With a digital magazine design, composed of all the content items saved previously, Pocket offers users an easy way to catch up on the content collected during the day and consume it comfortably when it's convenient for them.

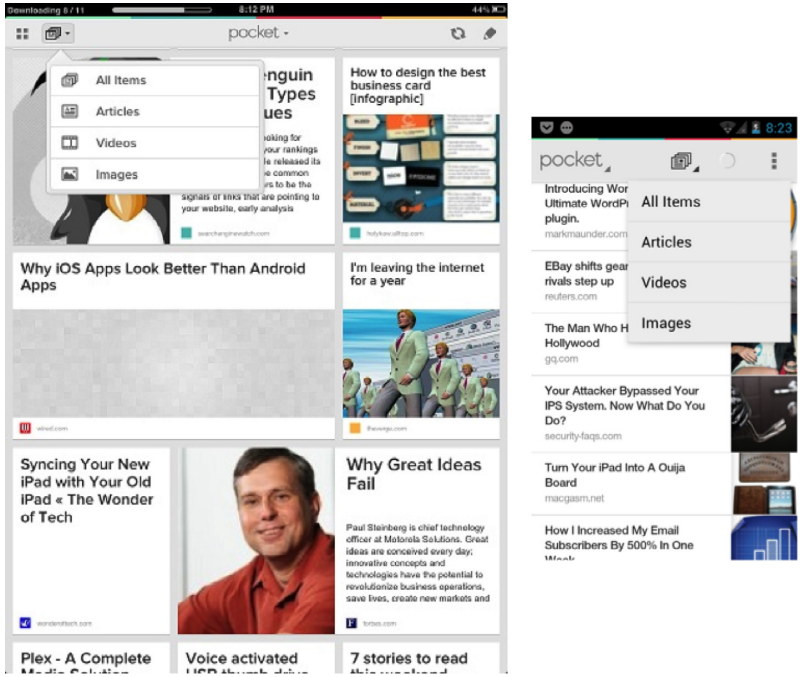


FIGURE 3-18

Pocket's mobile apps (for smartphone and tablet) are designed in a digital magazine format presenting the content saved during the day. This design acknowledges the strong role mobile devices (especially tablets) play as content consumption devices.

Design lesson: Divide and conquer

Pocket makes a very clear separation in its UI treatment between the context in which a user saves URL information (mostly the desktop browser, along with specific app integrations on other devices) and the one he uses to consume it (which is comprehensively handled in Pocket's own native app). The first merely offers a button to save the content the user runs into, and the latter is a fully designed, personalized, magazine-like experience made up of the personal saved content.

This workflow—aligned with the contextual behavioral patterns of Pocket's target audience—establishes a fluid, almost transparent experience. People don't really think about the different UIs; they just use them contextually, and it works.

Design lesson: The power of openness

Pocket has an open API that allows developers to create their own Pocket applications, integrate them into their existing apps, and share them with other Pocket users.

This capability has a couple of important advantages:

Increased distribution channels

The open API provides a more scalable solution for getting Pocket into many more entry points (or "saving" points) for users to access. Pocket is already integrated into over 500 applications, across many platforms people already use.

More channels feeding into the continuous experience

Users can trigger the experience from a growing number of channels, hassle-free, after installing the app once. While in many other ecosystem experiences, the continuous experience is limited to the specific product's website/native apps (Allrecipes and Eventbrite are two examples), in Pocket's case, any of the integrated apps is a starting point that can feed the continuous Pocket experience.

Open platforms carry many advantages in a multi-device world, as we'll see in Chapter 8.

Summary

- With a continuous approach, the multi-device experience flows from one device to the next, within different contexts, supporting users every step of the way through their task flow in two forms:
  - *Single activity flow*, where the same activity flows between devices, like watching a movie (Apple AirPlay), reading a book (Amazon Kindle), or composing a document (Google Drive).
  - *Sequenced activities flow*, which handles broader use cases composed of a sequence of different activities that all lead to the end goal—for example, cooking (Allrecipes), events management (Eventbrite), prototyping (POP), and content management (Pocket).
- Having users sign in to the product across all devices is significant in multi-device experiences—especially continuous ones. The increased complexity involved in these experiences, along with the constantly shifting contexts, requires building a robust knowledge

base of the users completing these activities so that their experiences can be personalized and improved over time. Sign-in is still the most prevalent way to identify the people using the product.

When building your authentication strategy, you should consider:

- Offering SSO
- Simplifying registration/sign-in flow and forms
- Incorporating gradual engagement
- Promoting the ecosystem benefits when users are signed in

- Continuous design, supported by multiple connected devices, enables us to rethink user flows we might have taken for granted, break them down into smaller steps, and then reconstruct them to create better, richer, more tailored user experiences.

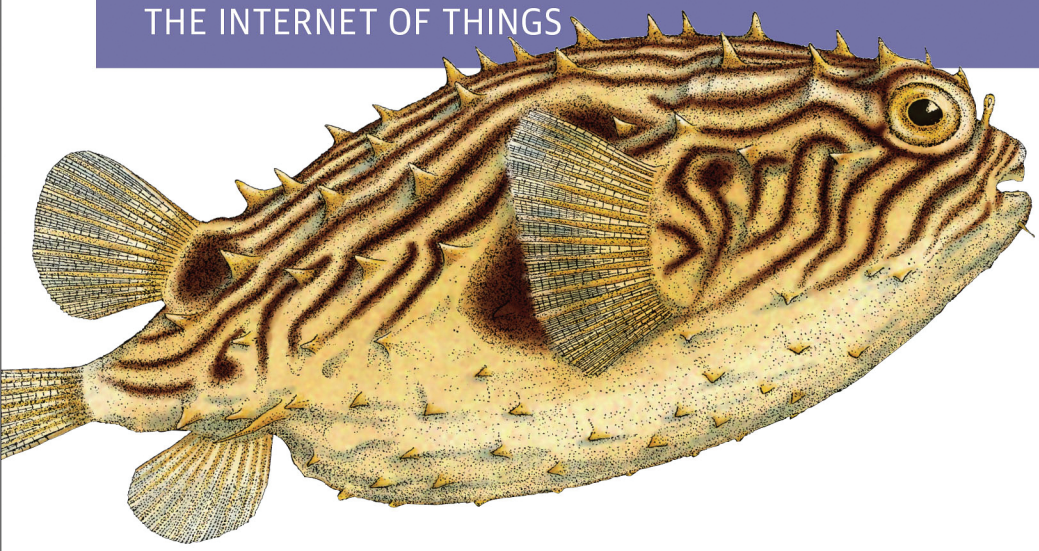
NOTES

1. Facebook login (*http://bit.ly/1eon4SU*); Twitter sign-in (*http://bit.ly/1eon8lS*); Google+ sign-in (*https://developers.google.com/+/*).
2. Hendrik Muller, Jennifer L. Gove, and John S. Webb, "Understanding Tablet Use: A Multi-Method Exploration," *Proceedings of the 14th Conference on Human-Computer Interaction with Mobile Devices and Services (Mobile HCI 2012)*, ACM, *http://bit.ly/1eop0Lp*.
3. "Allrecipes' Mobile Apps on Fire—Surpassing 10 Million Downloads," *Fresh Bites blog*, November 17, 2011, *http://freshbitesblog.com/2011/11/allrecipes-mobile-apps-on.html*.
4. Samsung has already launched a fridge with a built-in Android tablet and aWiFi-enabled washing machine (*http://bit.ly/1cQWnTS*).
5. Bill Siwicki, "Smartphone Owners Want More Mobile Information in Stores," Internet Retailer, December 31, 2012, *http://bit.ly/1ji1IJc;* "Mobile Devices Empower Today's Shoppers In-Store and Online," Nielsen, December 4, 2012, *http://bit.ly/JBNdQ5.*
6. "Email in Motion: Mobile Is Leading the Email Revolution" (infographic), *http://bit.ly/18ReaWu*.
7. Most of the events currently offered on Eventbrite are still primarily on the small, local, one-day scale. Still, if we're touching the events space, we might as well talk about it broadly, especially since Eventbrite has already started supporting more complex events—from classes to concerts and festivals, where it provides ticketing solutions for 20,000–80,000 attendees.
8. This is a significant change in context compared to the pre-event phase. Laptops or tablets are useful for stationary contexts (lokike taking notes during sessions), and smartphones are best for on-the-go situations (during breaks between sessions). The event UI on these devices can change accordingly.
9. A conference is often perceived as somewhere between business and pleasure; thus, participants can engage in post-event activities either during working hours (usually in front of the PC), or after (where the mobile devices are more prominent). Tracking your device usage with data analytics—discussed in Chapter 8—can help you determine how the different devices serve post-event needs.
10. Flayvr (*http://www.flayvr.com*) is a great example of an app that generates such an emotional experience.
11. "Myth #1: People read on the web," UX Myths, *http://bit.ly/1i6NkSX*.

O'REILLY®

Designing for Emerging Technologies

UX FOR GENOMICS, ROBOTICS, AND THE INTERNET OF THINGS



Jonathan Follett, Editor

Designing for Emerging Technologies

UX for Genomics, Robotics, and the Internet of Things

Edited by Jonathan Follett

O'REILLY® Beijing · Cambridge · Farnham · Köln · Sebastopol · Tokyo

[Contents]

[1]

Designing for Emerging Technologies

JONATHAN FOLLETT

A Call to Arms

Since the dawn of technology, humanity has lived with both its benefits and burdens. The fire that cooks our food also burns our hands; the mills and factories that produce our clothes often pollute our water and air; the computers that process our data sometimes crash and send our mission-critical records into oblivion. From the Agricultural to the Industrial to the Information Revolution, humanity has enjoyed great advantages from technology, but we have suffered the consequences of flawed thinking regarding its use, resulting in wasteful consumption of our world's resources, problems for our environment, and social disruption. Humanity enters the new millennium struggling with the challenges we've created for ourselves in areas from energy to infrastructure, transportation to healthcare, and manufacturing to agriculture. To address these challenges, we will no doubt turn once again to technology: in the coming century, we'll be able to hack our DNA, embed computers in our bodies, and print replacement organs. The question is, what will we do when we find ourselves with the capability to do just about anything we can dream of?

To explore that question—at least from a design perspective—let's consider the implications of four significant emerging technologies whose growth, maturation, and widespread commercial adoption has the potential to disrupt the current economic order:

- A networked, intelligent world connected by the Internet of Things (IoT)
- More efficient and effective manufacturing, healthcare, and disaster relief aided by advanced robotics
- Custom, just-in-time manufacturing, driven by additive fabrication/3D printing
- Medicine, food, and fuel created by altering the code of life itself, through genomics and synthetic biology

Through the lens of these disruptive technologies, we'll look at what designing products, services, and experiences for people might require and examine some of the high-level user experience (UX) tenets practitioners might consider when approaching the design for such new fields.

Today, we're on the cusp of a significant technological period, not unlike the Second Industrial Revolution that occurred in America from the end of the Civil War until World War I, when major discoveries and inventions completely transformed the economic, social, and political fabric of the United States. During this particularly prolific era, inventors, innovators, and scientists such as Alexander Graham Bell, Thomas Edison, Henry Ford, and Nikola Tesla introduced the world to technologies that would define modern life in the twentieth century. These included the light bulb, the telephone, and the mass-produced automobile, among many others. The light bulb and basic electrical service provided the cornerstones of the electric age, while the telephone started a communication revolution, and the automobile began an era of personal transportation that would alter the landscape of America itself. Historically, this period provides a powerful example of the systemic disruption that occurs when multiple technological innovations emerge, mature, and reach popular adoption along roughly the same timeline. Just as the inventions of the Second Industrial Revolution transformed the United States on almost every level, robotics, additive fabrication, the IoT, and synthetic biology similarly have the potential to define and shape our next era.

As these technologies evolve, they will influence humanity's progress as a species, making the tumult of our current Information Revolution look like a minor blip by comparison. Although the miracles of our age are many, computers, the Internet, and mobile devices primarily serve to *accelerate* human communication, collaboration, and commerce. Without dismissing their importance, we observe that many existing models of interaction have been enhanced, rather than transformed, by moving from the physical to the digital realm—becoming cheaper, faster and, perhaps, better in the process. Email is an accelerated version of the postal service, e-commerce, a more convenient and efficient version of the brick-and-mortar store, and so on. Conversely, in technologies such as synthetic biology and additive fabrication, we can see the potential to remake our current order in substantial fashion, with the formation of entirely new industries and the birth of new markets.

A May 2013 McKinsey Global Institute report titled "Disruptive technologies: Advances that will transform life, business, and the global economy," identifies the global markets ready for disruption and describes the potential economic value of this transformation. Genomics has the potential to alter the $6.5 trillion healthcare and $1.1 trillion agriculture industries through products such as personalized medicines and genetically modified foods. Additive fabrication/3D printing and advanced robotics will upend the global manufacturing industry—affecting the $6 trillion in labor expenditures and $11 trillion in global manufacturing GDP, respectively. And, the IoT will disrupt the manufacturing, healthcare, and mining industries to the tune of $36 trillion.[1]

Design for Disruption

Let's look briefly at the disruptive potential of each of these emerging technologies—the IoT, advanced robotics, 3D printing, and synthetic biology—and the need for design thinking in their formation.

1 James Manyika, Michael Chui, Jacques Bughin, Richard Dobbs, Peter Bisson, and Alex Marrs. "Disruptive technologies: Advances that will transform life, business, and the global economy." May 2013; McKinsey Global Institute Report (*http://bit.ly/1CJ2q8v*).

THE IOT, CONNECTED ENVIRONMENTS, AND WEARABLE TECHNOLOGY

The IoT is a popular shorthand that describes the many objects that are outfitted with sensors and communicating machine-to-machine. These objects make up our brave, new connected world. The types and numbers of these devices are growing by the day, to a possible 50 billion objects by 2020, according to the Cisco report, "The Internet of Things: How the Next Evolution of the Internet Is Changing Everything."[2] Inexpensive sensors providing waves of data can help us gain new insight into the places in which we live, work, and play, as well as the capabilities to influence our surroundings—passively and actively—and have our surroundings influence us. We can imagine the possibilities of a hyper-connected world in which hospitals, factories, roads, airways, offices, retail stores, and public buildings are tied together by a web of data.

In a similar fashion, when we wear these sensors on our bodies, they can become our tools for self-monitoring. Combine this capability with information delivery via Bluetooth or other communication methods and display it via flexible screens, and we have the cornerstones of a wearable technology revolution that is the natural partner and possible inheritor of our current smartphone obsession. If we consider that the systems, software, and even the objects themselves will require design input on multiple levels, we can begin to see the tremendous opportunity resident in the IoT and wearables.

ROBOTICS

In 2013, Google's purchase of eight robotics companies (which are to be consolidated into a new division led by Andy Rubin, the former head of its successful Android operating system) was publicly heralded as an inflection point for the robotics industry. Not coincidentally, the stocks of competitors such as iRobot rose dramatically.[3]

2 Dave Evans. "The Internet of Things: How the Next Evolution of the Internet Is Changing Everything." April 2011; Cisco White Paper (*http://bit.ly/1CJ2sNE*).

3 John Markoff. "Google Puts Money on Robots, Using the Man Behind Android." December 4, 2013; The New York Times (*http://nyti.ms/1CJ2tkR*).

More so than any other emerging technology, robotics has captured the imagination of American popular culture, especially that of the Hollywood sci-fi blockbuster. We're entertained, enthralled, and maybe (but only slightly) alarmed by the legacy of *Blade Runner, The Terminator, The Matrix* and any number of lesser dystopian robotic celluloid futures. It remains to be seen if robot labor generates the kind of societal, economic, and political change depicted in the more pessimistic musings of our culture's science fiction. Ensuring that it does not is a design challenge of the highest order.

In the near term, robots are ideal for taking care of jobs that are repetitive, physically demanding, and potentially hazardous to humans. As such, immediate opportunities for advanced robotics lie in areas where human labor is still intensive, such as manufacturing and logistics.

3D PRINTING

Additive manufacturing—more popularly known as 3D printing—is a process of creating a three-dimensional object by printing one miniscule layer at a time, based on a computer model. This flexible technology can use a wide variety of substrates including plastic, metal, glass, and even biological material. Custom production using additive manufacturing techniques promises to disrupt many industries, from construction to food to medicine. Possibilities for this technology range from immediately practical applications such as printing new parts just-in-time to fix a broken appliance; to controversial, uncomfortable realities, including generating guns on demand; to hopeful and futuristic methods, perhaps the ability to create not just viable human tissue, but complete, working organs, which could be used in transplants or for the testing of new drugs and vaccines.

Today, additive manufacturing is already changing architecture and construction. In April 2014, WinSun, a Chinese engineering company, reported that it can construct 10 single-story homes in a day by using a specialized 3D printing technology that creates the main structure and walls using an inexpensive combination of concrete and construction waste materials.[4]

4 "China: Firm 3D prints 10 full-sized houses in a day." April, 25 2014; BBC News (*http://bbc.in/1CJ2rJE*).

In the field of health, the work of roboticist Easton LaChapelle represents the change made possible by additive fabrication in medical-device prototyping and production processes. The 17-year-old wunderkind has created an ultra-light, fully functioning prosthetic arm whose parts can be 3D-printed for about $500. Traditionally manufactured prosthetic arms that are currently available can cost upward of $80,000. LaChapelle's prosthetic arm is controlled using an EEG headset, which measures brainwaves and communicates with the arm wirelessly via Bluetooth.

At the Business Innovation Factory BIF9 conference in Providence, Rhode Island, held in September 2013, LaChapelle demonstrated his invention and discussed his amazing progression through the design and prototyping phases. The first generation of the product LaChappelle created was a robotic hand, made of Lego bricks, surgical tubing, and five servo motors. He created the second-generation robotic arm by using 3D-printed parts and a Nintendo Power Glove. Now in its third generation, the arm is made almost entirely of 3D-printed parts, and most dramatic of all, it has human strength. While LaChapelle has not made the leap from prototype to a manufacture-ready device, it's easy to imagine the potential for disruption in the market it represents.

From a process standpoint, LaChappelle's methods in designing and engineering the prosthetic demonstrate the speed at which ideas can move from a designer's imagination to becoming something real and testable. Even though prototyping has always been a part of the designer's toolkit, additive fabrication makes it possible to apply the same rapid and flexible process of ideation, creation, testing, validation, and iteration to physical products that used to be reserved for the realm of digital development.

GENOMICS AND SYNTHETIC BIOLOGY

In April 2003, the publicly funded Human Genome Project completed the sequencing of the entirety of our human DNA, providing the blueprint for building a person at a price of $3 billion. At the time, this scientific achievement was heralded as one of the greatest in history, with far-reaching implications for health and medicine. Then-President Bill

Clinton, in announcing the working draft sequence of the Human Genome in 2000, said "Without a doubt, this is the most important, most wondrous map ever produced by human kind."[5]

Fast forward just slightly more a decade, and the cost of sequencing a human genome has dropped to roughly $1,000—an exponential reduction in price far exceeding what Moore's Law would predict. A host of companies are racing to introduce technology to make even more rapid and inexpensive sequencing possible. With the widespread affordability of this sophisticated test quickly becoming reality, genomics can provide the map for a new wave of personalized therapies: highly targeted drugs for fighting cancer, cardiovascular disease, diabetes, and other hereditary illnesses.

As with genomic sequencing, the price of DNA synthesis continues to drop. It now approaches 25 cents per base pair via services such as GenScript, DNA 2.0, and others. Writing the code of life is the cornerstone of the science of synthetic biology; the intentional design and engineering of biological systems will make incredible things possible. In his book *Regenesis*, George Church, geneticist, Harvard professor, and perhaps the most well-known scientist in this field, outlines some of the inventive solutions offered by this future potential, including biofuels, targeted gene therapies, and even virus-resistant human beings. In Church's expansive vision, we see a future in which humans have the capability to design nature itself, changing the fabric of biology and human evolution.[6]

If advanced robotics, additive fabrication, synthetic biology, and other emerging technologies extend humanity's grasp, making it possible for us to achieve our goals in manufacturing, health, energy, and other industries rapidly and efficiently, they also will place their own set of demands upon us. The fields of graphic design, industrial design, and software UX design have all evolved in response to the unique demands of new technologies. It's a classic human characteristic to relate to and interact with our latest tools. Graphic design makes information

5 "Breaking News—President Clinton, British Prime Minister Tony Blair Deliver Remarks on Human Genome Milestone." June 26, 2000; CNN.com Transcripts (*http://bit.ly/1CJ2xky*).

6 George Church and Edward Regis. *Regenesis: How Synthetic Biology Will Reinvent Nature and Ourselves*. 2012; Basic Books.

depicted in printed media clear, understandable, and beautiful; industrial design makes products elegant, usable, and humane; and UX design makes the interaction with our digital products and services efficient and even pleasurable.

Historically, bridging this gap between man and disruptive technology has not been an easy task. In addition to the positive outcomes of the Industrial Revolution, the political, societal, and economic change across the globe also engendered negative responses by people whose lives and livelihoods were displaced. Most famously, the Luddites in England sabotaged machines in the factories, preferring to destroy the technology they felt would ruin man's existence, rather than proceed forward in concert with it. Mass production similarly had its tensions with traditional craftsman—from woodworkers to painters to architects—who responded by creating the influential Arts and Crafts movement in the late 1800s. And, in the literature of the time, industrialization faced one of its most enduring critiques in Mary Shelley's *Frankenstein*, which articulated most eloquently the idea that science left unchecked and pursued for its own sake was not to be trusted. We, too, can expect that the road to adopting the emerging technologies of our time will be subject to societal tensions and negative reactions, critiques, and counter-movements, which is all the more reason for designers to involve themselves early.

Eight Design Tenets for Emerging Technology

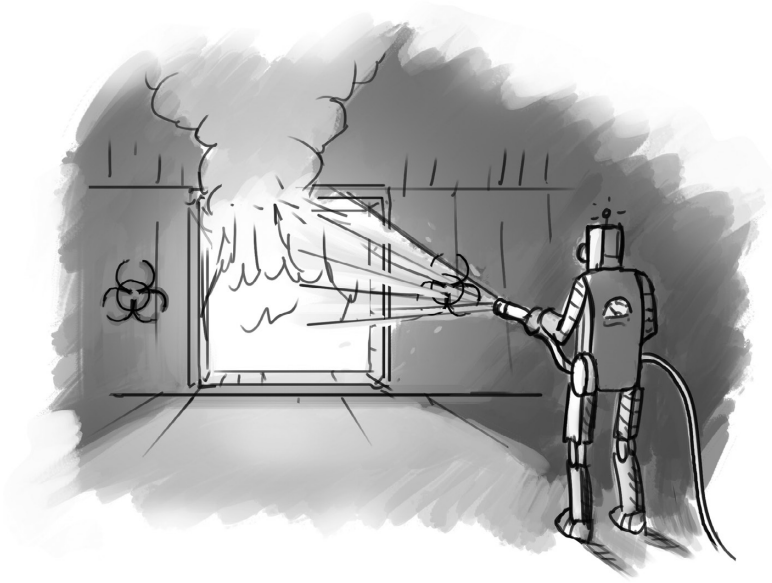
As we face a future in which the definition of what it is to be human might be inexorably changed, we will need design to help frame our interactions with technologies, from skin-top embeddable computers to bioprinted organs to swarming robots, which often seem to be racing ahead of our ability to process and manage on an emotional, ethical, and societal level. Designers have an opportunity to help define the parameters of and sculpt the interactions between man and technology, whether we're struggling with fear and loathing in reaction to genetically altered foods, the moral issues of changing a child's traits to suit a parent's preferences, the ethics guiding battlefield robots, or the societal implications of a 150-year extended lifetime. At its best, not only can design provide the frame for how technology works and how it's used, it can also situate it within a broader context: incorporating system thinking, planning for a complete technological lifecycle, and evaluating the possibility of unintended consequences.

Our field of practice will be transformed, as well, and we must prepare for it by moving from design as facilitation, shaping the interface and workflow, to design as the arbiter, driving the creation of the technology itself and applying our understanding of interaction, form, information, and artistry to new areas. To balance those asking, "How can this be done?" we should ask, "Why should we do this, to what end, and for whose benefit?" We must move from being passive receptors of new technology to active participants in its formation. As design thinkers and practitioners we're called to be explorers. And, although it's true that not every designer will want or be able to follow this path, those who do will have an opportunity to contribute in significant fashion.

Today, design work is changing at an unprecedented pace, and we are all too aware of the need to constantly evolve our skills to match the demands of the marketplace. With this uncertainty comes opportunity: the design positions of the future are not yet defined. Just as there was no industrial, graphic, or interaction designer at one time, so too the designer of emerging technologies has, at least for now, a broad canvas to explore. How we bring our current skills to bear on new problems, how we determine new subjects to learn, and how we integrate with burgeoning new industries will all play a part in the way our emerging design practices form.

What competencies will be most important for the designer in these new areas? What approaches will be most effective for managing the disruptive power of emerging technologies? What thinking processes will help the designer negotiate the technical, social, and ethical complexities that emerging technology will inevitably present? As a first attempt at answering these questions, the tenets that follow articulate some high-level guidelines for creative thinking and process development, drawing broad-based inspiration from related professional fields, including architecture, art, ethnography, engineering, and most of all, user experience. Although these tenets are certainly also applicable to knowledge work in a general sense—for the scientist, technologist, or entrepreneur—we will consider them in the context of design for emerging technology.

1. IDENTIFY THE PROBLEMS CORRECTLY



The gap between the problems we face as a species and the seemingly unlimited potential of technologies ripe for implementation begs for considered but agile design thinking and practice. Designers should be problem identifiers, not just problem solvers searching for a solution to a pre-established set of parameters. We must seek to guide our technology, rather than just allow it to guide us.

On the cover of the November/December 2012 issue of MIT Technology Review, the shortcomings of the past decade's technological achievements are expressed in the damning headline dramatically superimposed in white type over the bemused portrait of astronaut Buzz Aldrin: "You Promised Me Mars Colonies. Instead I Got Facebook." The subhead elaborates tellingly: "We've stopped solving big problems. Meet the technologists who refuse to give up." The accompanying article "Why We Can't Solve Big Problems"[7] details some of the current limitations in American culture, finance, and politics that, since the Apollo moonshot, have relegated big thinking and technical aspirations to the sidelines. The author, however, concludes the following:

7 Jason Pontin. "Why We Can't Solve Big Problems." MIT Technology Review, October 24, 2012 (*http://bit.ly/1CJ2zJ9*).

> It's not true that we can't solve big problems through technology; we can. We must. But all these elements must be present: political leaders and the public must care to solve a problem, our institutions must support its solution, it must really be a technological problem, and we must understand it.

We are on the cusp of a new technological age, saddled with the problems of the previous one, demanding that as we step forward we do not make the same mistakes. To do this, we must identify the right challenges to take on: the significant and valuable ones. Chief among our concerns must be the environment, not only in reducing the carbon we release as a result of consumption and seeking new sources of energy, but also in understanding the effects of a growing global population, against the backdrop of limited resources. We must also improve human health and consider the ramifications as humans live longer lives. And, we must find new ways to manufacture goods and produce food and clean water for a planet currently with 7.2 billion inhabitants—a population that is projected to explode in the next 35 years by an additional 2.4 billion, reaching 9.6 billion by 2050, according to the UN report, "World Population Prospects: The 2012 Revision."[8] Recognizing these major challenges for humanity in the twenty-first century and seeking proactive solutions, even in significant areas such as the environment, energy, health, manufacturing, agriculture, and water usage, will not be an obvious or easy task.

We can see an example of this in the tragic events of the Fukushima meltdown. On March 11, 2011, a 9.0 magnitude earthquake and subsequent tsunami damaged the Fukushima Daiichi nuclear reactors in Japan. Over the course of 24 hours, crews tried desperately to fix the reactors. However, as, one by one, the backup safety measures failed, the fuel rods in the nuclear reactor overheated, releasing dangerous amounts of radiation into the surrounding area. As radiation levels became far too high for humans, emergency teams at the plant were unable to enter key areas to complete the tasks required for recovery. Three hundred thousand people had to be evacuated from their homes, some of whom have yet to return.

8 United Nations, Department of Economic and Social Affairs, Population Division (2013). World Population Prospects: The 2012 Revision (*http://esa.un.org/wpp/*).

The current state of the art in robotics is not capable of surviving the hostile, high-radiation environment of a nuclear power plant meltdown and dealing with the complex tasks required to assist a recovery effort. In the aftermath of Fukushima, the Japanese government did not immediately have access to hardened, radiation-resistant robots. A few robots from American companies—tested on the modern battlefields of Afghanistan and Iraq—including iRobot's 710 Warrior and PackBot were able to survey the plant.[9] The potential for recovery-related tasks that can and should be handled by advanced robotics is far greater than this. However, for many reasons, spanning political, cultural, and systemic, before the Fukushima event, an investment in robotic research was never seriously considered. The meltdown was an unthinkable catastrophe, one that Japanese officials thought could never happen, and as such, it was not even acknowledged as a possible scenario for which planning was needed.

The Fukushima catastrophe inspired the United States Defense Advanced Research Projects Agency (DARPA) to create the Robotics Challenge, the purpose of which is to accelerate technological development for robotics in the area of disaster recovery. Acknowledging the fragility of our human systems and finding resilient solutions to catastrophes—whether it's the next super storm, earthquake, or nuclear meltdown—is a problem on which designers, engineers, and technologists should focus.

In the DARPA competition mission statement, we can see the framing of the challenge in human terms.

> History has repeatedly demonstrated that humans are vulnerable to natural and man-made disasters, and there are often limitations to what we can do to help remedy these situations when they occur. Robots have the potential to be useful assistants in situations in which humans cannot safely operate, but despite the imaginings of science fiction, the actual robots of today are not yet robust enough to function in many disaster zones nor capable enough to perform the most basic tasks required to help mitigate a crisis situation. The goal of the DRC is to generate groundbreaking research and development

9 Marina Koren. "3 Robots That Braved Fukushima." Popular Mechanics (*http://bit.ly/1CJ2DZu*).

> in hardware and software that will enable future robots, in tandem with human counterparts, to perform the most hazardous activities in disaster zones, thus reducing casualties and saving lives.[10]

The competition, so far, has been successful in its mission to encourage innovation in advanced robotics. In the competition trials held in December 2013, robots from MIT, Carnegie Mellon, and the Google-owned Japanese firm, Schaft, Inc., competed at a variety of tasks related to disaster recovery, which included driving cars, traversing difficult terrain, climbing ladders, opening doors, moving debris, cutting holes in walls, closing valves, and unreeling hoses.

2. LEARN CONSTANTLY

Designers will need to understand the implications of science and technology for people. To do this effectively, we must be able to immerse ourselves in new technical domains and learn them quickly. Just as our understanding of and empathy for people allows us to successfully

10 "DARPA Robotics Challenge." DRC. *http://www.theroboticschallenge.org/* (accessed June 10, 2014)

design with a user's viewpoint in mind, understanding our materials, whether they be pixels or proteins, sensors or servos, enables us to bring a design into the world. To achieve this, designers need to be early adopters of technology, learning constantly.

The ability to quickly learn new materials and techniques has always been one of the most important of a designer's core competencies. However, the speed at which this is expected and at which technological change occurs is the critical difference today. *How* we learn will soon become as important a consideration as *what* we learn. To prepare designers for the new roles that emerging technology will bring, schools will need to develop curricula that emphasize continuous learning as a core competency and provide tools and methods to enable it.

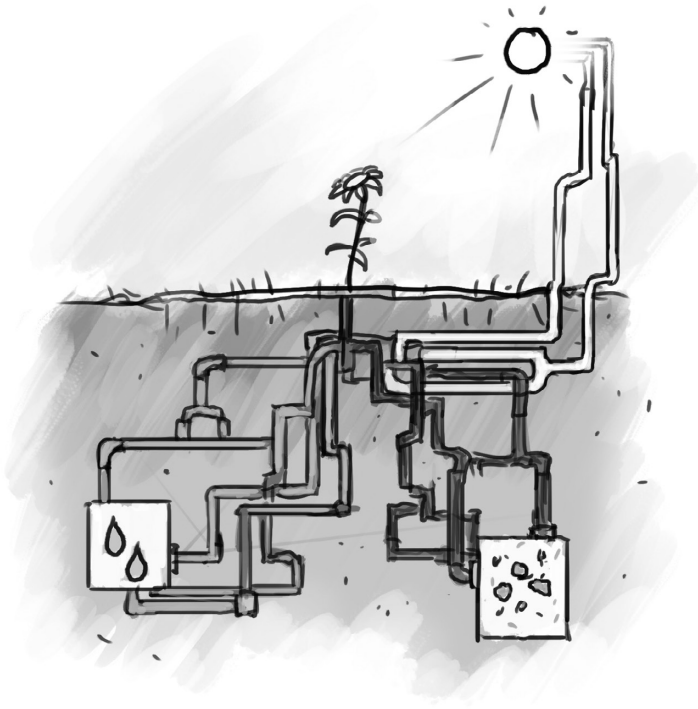
AIGA, the professional association for design, and Adobe Systems, Inc., the design software giant, released research, "Defining the Designer of 2015," based on the input of 2,500 designers and a variety of experts and focused on the future of the field.[11]

> In order to fulfill the expectations placed upon designers in the future, they will need to employ a set of skills that include some beyond today's typical scope. No single designer is likely to have all the skills required, yet this research revealed the range of competencies that a studio or design department, among its full complement of staff, will need in order to meet the demands of the future.

Although the AIGA/Adobe survey results focus largely on communication-related design, it acknowledges that among the competencies needed by the designer of 2015, the need for "understanding of and ability to utilize tools and technology" and the "ability to be flexible, nimble and dynamic in practice." Ultimately, designers will need to be lifetime learners.

11 "Defining the Designer of 2015." AIGA. *http://www.aiga.org/designer-of-2015/* (accessed June 10, 2014)

3. THINK SYSTEMICALLY

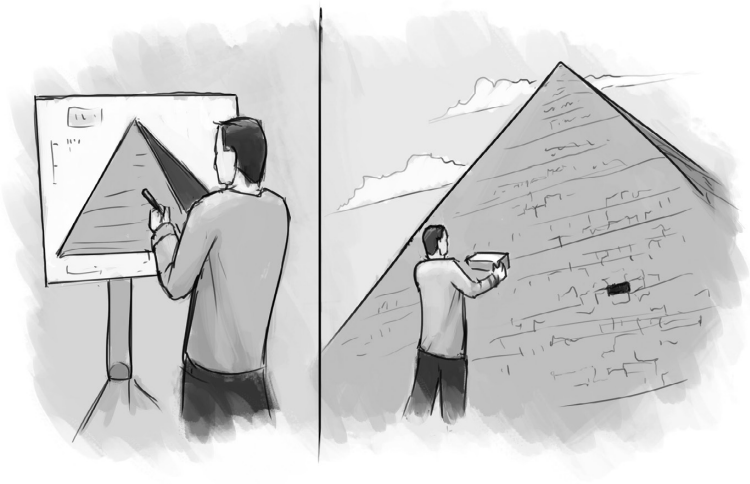


Increasingly, designers will also need to be system thinkers. As we consider the fields of advanced robotics, synthetic biology, or wearable technology, the design of the ecosystem will be just as important as the design of the product or service itself.

A good example of such a product is Mimo, a next-generation baby-monitoring service that goes far beyond the usual audio and video capabilities in soothing the anxieties of new parents. A startup company led by a group of MIT engineering grads called Rest Devices has created an ingenious baby "onesie." It's a connected product that delivers a stream of data including temperature, body position, and respiration information, ensuring that mom and dad are fully versed in the minutiae of their offspring. What at first glance might seem like the enablement of over-parenting paranoia, could, in fact, also provide valuable scientific data, particularly given that crib death or SIDS (Sudden Infant Death Syndrome) is a phenomenon that is still not fully understood.

From a design perspective, a company such as Rest Devices has a range of needs typical of those startups in the budding wearable technologies industry. The onesie itself must be designed for both functional and aesthetic elements—a mixture of industrial design for the "turtle" on-body device that houses the sensor and the fashion design of the garment itself. The mobile software application that provides the data interface requires interaction design and visual design—not to mention the UX design of the total system, which must be optimized for setup and navigation by nervous parents. Whether one person or many provide these different design skills for Rest Devices, it's clear that at every point at which people touch the technology, there is ample opportunity for the interaction to be carefully examined and optimized in relation to the entire ecosystem. In this way the Mimo is a good example of the first wave of wearable technology. Like the Nike FuelBand, the Fitbit, and even the Recon heads-up ski display, these wearables represent technology embedded into the infrastructure of our lives in a way never before seen. But the magic of the consumer experience of these products is only possible through the design of a complete, and hopefully seamless, ecosystem.

4. WORK AT A VARIETY OF SCALES



Designers should be able work at a variety of scales, from the aforementioned overall system view, to the nitty-gritty details. Moving between these levels will be important, too, as each one informs the other—the macro view informs the micro, and vice versa.

At the highest level, designers can work proactively with politicians and policy makers to effectively regulate new technology. As one example of this, in September 2013, the FDA released final guidance on mobile medical apps, which was crafted with input from industry experts. From bioethics to industrial regulations governing the use of robotics, designers will want and need to have input into the realm of policy. Just as free markets cannot exist without effective and enforceable contract law, so, too, technological advancement cannot exist without sensible, effective, and enforceable regulation with a long-term view. Designers will need a seat, not just at the computer or the lab bench, but at the policy-making table, as well.

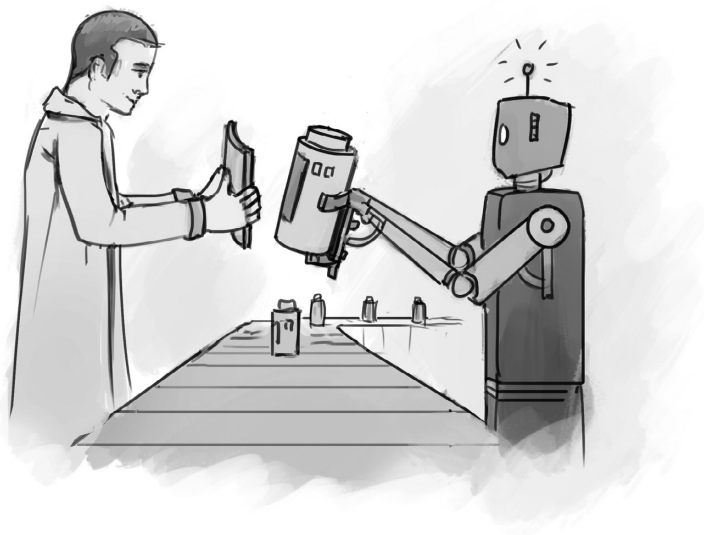
At Involution Studios, an experience design firm, we've worked with healthcare industry clients on emerging technology projects such as genomics research software and connected medical devices. As one depiction of how design thinking can be utilized at the organizational level, one engagement we recently completed for the Personal Genome Project (PGP) at Harvard University exemplifies how design research tools and techniques can help cutting-edge organizations focus on human-centered problems.

Founded in 2005 by George Church, the mission of the Personal Genome Project (PGP) is to sequence and publicize the genomes of 100,000 volunteers for use by the scientific community in the improvement and management of human health and disease. When Involution first engaged with the PGP, however, everyone on the team saw the organization functioning slightly differently, with different objectives.

It was clear that the volunteer members—those who were donating their genome and medical histories—were the type of people who were willing to take on great risk to help advance scientific discovery. The PGP had done a good job of educating their members on the possible perils that could come with publishing their genomic data for the world to see. However, the members received little to no feedback on whether their information was being used by the scientific community, how it was being used, or the potential impact of their contributions. Because members could reclaim their contributions at any time, the PGP needed to find a way to better nurture these relationships so that the project could continue striving toward its goal of 100,000 participants.

Involution and the PGP collaboratively mapped the organization's ecosystem to better visualize how it was functioning. "It was time for us to step back and talk about what that overarching goal was, and we did that through this workshop. It was very helpful," said Jason Bobe, executive director of the PGP, who believes the workshop gave his team members a forum to tease out their big ideas as well as their roles within the organization. As a result of sketching, brainstorming, and mapping exercises, the PGP team members were able to see the risks and benefits of their organizational model through the lens of the many different users they were serving. "We had a team where we had a bunch of ideas about what we were doing, but we didn't have a cohesive, shared understanding of how we were going to do everything we wanted to do," said Bobe. With Involution's help, the team was able to uncover multiple challenges that were holding the PGP back. The result from the workshop was significant: a complete reimagining of the PGP's organizational model, separating member recruitment efforts from data collection and sequencing, and truly focusing on member relationships.

5. CONNECT PEOPLE AND TECHNOLOGY



Design should provide the connective tissue between people and technology. The seamless integration of a technology into our lives is almost always an act of great design, coupled with smart engineering; it's the "why" that makes the "what" meaningful. It is through this humane expression of technology that the designer ensures a product or service is not just a functional experience, but one that is also worthwhile. We must consider the outputs of these technologies—what people need and want. The designer should ask: "Why are we doing these things? How is humanity represented against what's possible with technology?" It is the designer's duty to be a skeptic for the human side of the equation.

For instance, as robots take a greater role in the fields such as manufacturing by automating repetitive and dangerous tasks, as well as augmenting human abilities, we can see that even though there are many benefits, there remains a question as to how such robotic optimization can coexist with meaningful work for people in the long term. At first glance, the combination of collaborative robotics and agile manufacturing seems to be one potential answer to this problem. Rethink Robotics' Baxter, Yaskawa Motoman's Dexter Bot, and Universal Robotics' UR are examples of collaborative robots designed with human-like characteristics, flexibility regarding the tasks they can execute, and ease of programming, opening up new possibilities for working in tandem with human workers on the factory floor. In this model, human labor is augmented by, not replaced with, the robotic technologies.

Advanced collaborative robotics could readily provide the flexible systems required to meet the demands of agile manufacturing. A key advantage to robotic manufacturing is its adaptability: robotic production lines can be easily modified to accommodate shorter-run, customized products. We could soon see robots replace expensive dedicated industrial machinery made for specific production processes, which can be extremely difficult to repurpose when changes to a process are required. As a part of this agile manufacturing paradigm, robots with the ability to work in collaboration with human beings—in factories, warehouses, and other industrial settings—will be a critical component. Human workers will be responsible for programming, monitoring, supervising, and otherwise interacting with a robotic workforce that is repurposed regularly to handle the creation of custom, short-run production.

6. PROVOKE AND FACILITATE CHANGE

It is not only the designer's responsibility to smooth transitions and find the best way to work things out between people and the technology in their lives; it is also the designer's duty to recognize when things are not working, and, rather than smooth over problems, to provoke wholesale change. Technological change is difficult and disruptive. Even today, there are countless examples of technologies outpacing the frameworks for controlling them, resulting in a sense of unease in people about the seemingly unprecedented and unchecked advances, from digital surveillance encroaching on our privacy to genetically modified foods filling our grocery stores. Designers can start the discussion and help lead the process of transformation.

As one illustration of this, despite the seemingly unlimited potential of genomics, or perhaps because of it, the tension between those who wish to move the science forward and those cautioning restraint is palpable. Take the example of 23andMe, a company that provides inexpensive personal DNA sequencing. In November 2013, the United States Food and Drug Administration (FDA) shut down the service for

supplying medical interpretation of the DNA data in the reports the company issued to its customers.[12] Here's the notice that appeared on the 23andMe website:

> At this time, we have suspended our health-related genetic tests to comply with the U.S. Food and Drug Administration's directive to discontinue new consumer access during our regulatory review process. We are continuing to provide you with both ancestry-related genetic tests and raw genetic data, without 23andMe's interpretation. ... We remain firmly committed to fulfilling our long-term mission to help people everywhere have access to their own genetic data and have the ability to use that information to improve their lives.

The FDA's action brings into question the future regulatory environment that scientists, entrepreneurs, designers, and engineers could encounter in the realm of personalized medicine. The regulator's dilemma for emerging tech is that the rules governing familiar industries might not apply to technology likely to be disruptive to those industries. In the case of 23andMe, regulators must balance the benefits of customers learning about diseases they carry or are at risk for with the dangers of false positives, misuse, or misinterpretation of the data. 23andMe is disruptive because it provides personal genomic testing at a low price point. It also ventures into territory never before seen by an industry familiar with expensive testing regimes typically administered in a reactive, rather than proactive, manner.

In December 2013, the Presidential Commission for the Study of Bioethical Issues released its report, "Anticipate and Communicate: Ethical Management of Incidental and Secondary Findings in the Clinical, Research, and Direct-to-Consumer Contexts." This document provides guidance on how to manage the issues of incidental and secondary findings.

> How clinicians, researchers and direct-to-consumer companies manage incidental and secondary findings will likely touch all of us who seek medical care, participate in research, or send a cheek swab to a company for a peek at our own genetic make-up," said Amy Gutmann,

12 Chris Welch. "FDA orders 23andMe to halt sales of DNA test kit." The Verge. *http://bit.ly/1CJ2GEt* (accessed June 10, 2014)

> Ph.D., Commission Chair. "The reality is that we might find out more than we bargained for. Yet practitioners are getting conflicting advice about how to manage such findings across contexts and modalities such as genetics, imaging, and biological specimen testing. We all need to know how to better manage health information we did not expect.[13]

As designers, we need to be engaged in proactive, society-wide conversations such as this so that we can help define safe boundaries and people-centric policies ahead of time, rather than trying to figure things out—and spend time defending our industries' very existence—after the horse has left the gate.

7. WORK EFFECTIVELY ON CROSS-DISCIPLINARY TEAMS

The challenges inherent in much of emerging technology are far too great for an individual to encompass the requisite cross-domain knowledge. For this kind of work, then, the team becomes paramount. It is a multidisciplinary mix of scientists, engineers, and designers who are best positioned to understand and take advantage of these technologies. And, it is crucial that these creative disciplines evolve together.

13 "Bioethics Commission on Incidental Findings: Anticipate and Communicate." December 12, 2013; Presidential Commission for the Study of Bioethical Issues (*http://bioethics.gov/node/3186*)-

From such collaborations new roles will be created: perhaps we will soon see a great need for the synthetic biological systems engineer or the human-robot interaction designer. This cross-pollination of science, design, and engineering is already happening at organizations such as the Wyss Institute at Harvard, whose mission is to develop materials and devices inspired by nature and biology. Wyss structures itself around multidisciplinary teams. Forward-thinking design firms such as IDEO have also added synthetic biology to their established practices of industrial and digital design.

As an example of this cross-pollination, in a presentation, "Life is what you make it," given at a Friday Evening Discourse at The Royal Institution of Great Britain in London, esteemed scientist and Imperial College professor Paul Freemont described how biological design could take its cues from computer software engineering, using an abstraction hierarchy for biological design.[14] In the design of complex systems, an abstraction hierarchy makes it possible for engineers to focus on solving the problems at hand because they don't necessarily need to understand the complexity of the lower levels of the hierarchy. In software development, for example, engineers can code in Java or C++ and not need to understand the machine-level code that ultimately executes the program. In the coming revolution in biological design, such an abstraction hierarchy will offer bioengineers the capability to operate similarly.

Although programming might be an apt analogy for that manipulation of nature, there are fundamental differences between the writing of computer code and genetic code. Even if we know the outcome of the genetic code we write, the environment into which it is released is far more complex than the controlled operating system of a computer or mobile device. There is so much unknown about biological systems that prototyping and testing will be critical steps for responsible innovation. Even though designers won't necessarily need to become genetic engineers to contribute to the field of synthetic biology, they'll need to understand the materials just as deeply.

14 Paul Freemont. "Life is what you make it." November 29, 2013; The Royal Institution, Friday Evening Discourse. (*http://bit.ly/1CJ2Mfj*).

At Boston University, the Cross-Disciplinary Integration of Design Automation Research (CIDAR) lab is creating bioCAD tools such as Clotho, an open source software framework for engineering synthetic biological systems. The larger goal for Clotho—named for the Greek goddess of Fate who was responsible for spinning the thread of human life—is to create standardized data, algorithms, and methodologies for synthetic biology. Other software tools such as Genome Compiler and Gene Designer aim to improve the process of genome creation from design to quality assurance to fabrication. At the intersection of software design and genome design, these tools for automating aspects of the synth bio process are cross-disciplinary efforts.

8. TAKE RISKS, RESPONSIBLY

To find our way forward as designers, we must be willing to take risks—relying upon a combination of our education, experience, and intuition—which can be crucial to innovation. We must always keep in mind both the benefits and consequences for people using these new technologies, and be prepared for mixed results.

The Glowing Plant Kickstarter project is a good example of such inspired risk taking in action. There is perhaps no technology more fraught with perceived peril than genomics and synthetic biology. Seeing the opportunity to both inspire and educate the public, a team of biochemists started a project to generate a bioluminescent plant, which they touted as "the first step in creating sustainable natural lighting." Financed on the crowd-funding website Kickstarter, the Glowing Plant project generated so much grassroots excitement that it raised $484,013 from 8,433 backers, far exceeding its initial goal of $65,000.

However, soon after the Glowing Plant project finished its campaign, Kickstarter, without any explanation, changed its terms for project creators, banning genetically modified organisms (GMOs) as rewards for online backers.[15] Glowing Plant, with its project financing already in place, might be the last example of crowd-funded synthetic biology for a while. Although this incident, in and of itself, might seem minor, it's worth remembering that Kickstarter is the primary resource for crowd-funding in the United States. Removing this financial option for synthetic biology startups, in a seemingly arbitrary decision, will have a chilling effect on future innovators.

The results of the Glowing Plant crowd-funding project illustrate the promise and perils of designing for such a disruptive technology as synthetic biology. How do we evaluate the risk and reward, in this case, knowing the outcome? Even though the team initially received immense grassroots enthusiasm and financial backing, they also caused the Kickstarter ban, as an established corporate entity reacted with fear. During this transition time between fear and acceptance, designers of genetically modified organisms, like the team behind the Glowing Plant project, will continue to push the envelope of what companies, regulators, and the government find acceptable. It's safe to say

15 Duncan Geere. "Kickstarter bans project creators from giving away genetically-modified organisms." August 2, 2013 (*http://bit.ly/1CJ2QvH*).

that until synthetic biology is better understood, policy decisions such as this ban will continue to happen. It might be that a willingness to push forward and to take risks will be important to making the transition, to reach public acceptance and ultimately help move the technology forward.

Changing Design and Designing Change

People are less interested in the science and engineering, the mechanisms that make emerging technologies such as advanced robotics, synthetic biology, and the IoT possible, but they are deeply concerned with the outcomes. As these technologies emerge, grow, and mature over the coming years, designers will have the opportunity to bridge human needs and the miraculous technological possibilities.

It will be a great and even intimidating challenge to involve design early in the process of defining new products and services, but it will be critical as we establish the practices of the twenty-first century—from the design of technology policy, to systems, to tactical interaction frameworks and techniques. Policy design will involve advising regulators and politicians on the possibilities and perils of emerging tech; system design will demand clear understanding of the broader interactions and implications that surround the immediate details of a product; and framework design will benefit our day-to-day tactical work, providing a foundation for designers and design practice to come. What all of these technologies will create, as they evolve together, remains to be seen. But, the most interesting discoveries will be at the intersections.

Understanding new technologies, their potential usage, and how they will impact people in the short and long term will require education and collaboration, resulting in new design specializations, many of which we have not yet even considered. In the coming years, as the boundaries between design and engineering for software, hardware, and biotechnology continue to blur, those who began their professional lives as industrial designers, computer engineers, UX practitioners, and scientists will find that the trajectory of their careers takes them into uncharted territory. Like the farmers who moved to the cities to participate in the birth of the Industrial Revolution, we can't imagine all of the outcomes of our work. However, if history is any indicator, the convergence of these technologies will be greater than the sum of its parts. If we are prepared to take on such challenges, we only have to ask: "What stands in the way?"

[2]

Intelligent Materials: Designing Material Behavior

BROOK KENNEDY

Bits and Atoms

Will bits eventually control atoms? It is certainly tempting to think so—digital tools mediate ever-increasing parts of our physical environment. Walk down any urban street these days and you will see droves of people glued to their devices—checking their messages, posting photos, even turning their heat down at home, all digitally—while being completely oblivious to the world of atoms around them. And, this is only the beginning. Sure, the physical design of our gadgets might earn our admiration and devotion, but isn't it what happens on the screen that really commands our attention? Just as the iPhone represents iconic industrial design now, it could just as easily be remembered one day as a milestone in the inevitable shift to bits from atoms. After all, who needs a wallet, a clock, a map or even a flashlight when "there's an app for that"?

Science fiction films such as *Minority Report* present future visions of digital experiences integrated into our lives to such an extent that the physical object disappears altogether—from hardware-free interfaces that we control by waving our hands through the air to Google Glass. Pervasive computing of this kind will certainly continue to expand into all of the activities around us in the home, at the office and in the public domain. But what would happen if digital technology were to reenter the physical world at the most basic material level? What if changing the wallpaper on the walls of your home were just as easy as changing the wallpaper on your computer desktop? Could the materials of products and environments themselves actually "behave" more like the

dynamic screens with which we interact? At some point in the not-so-distant future, the answer will be yes. Converging knowledge at the intersection of biology, additive manufacturing, and computing are driving new research frontiers such as *adaptive materials* and *programmable matter* that might bring about this future. For the purpose of this chapter, we will call these new fields *Intelligent Materials*—when combined, the outcome of these emerging research areas will have a huge impact on physical design.

In traditional physical design disciplines such as architecture and industrial design, understanding materials has always been an important foundation in learning the craft. Materials have unique properties that are employed to construct buildings or mass-manufacture the products we have traditionally relied on in our daily lives. Stone, wood, metals, plastics, and composites are harvested, quarried, forged, and synthesized in a chemical facility, or a combination thereof. When delivered, physical designers will then shape, mill, mold, and manipulate these materials into an assembly of other parts to create a finished product. Materials are selected for their inherent properties whether those properties are appearance, strength, elasticity, translucency, or any other combination of desired qualities that are suitable for the intended use case of a designed object. Materials frequently perform a specialized function by means of their chemical properties but often with an undesirable trade-off of toxicity or recyclability. The resounding pattern here is that materials are basically static and designers have to accept their properties and limitations and compromise accordingly.

In coming decades, we will see a fundamental evolution in the meaning of the word material. Materials will be able to be optimized to a particular purpose by fine-tuning the microscopic physical surface structure rather than by altering their chemistry. More to the point, we will also see the introduction of more materials that can change on demand through devices or computer control to fit our needs. Just as screen technologies such as LCD and E-Ink can change quickly to display moving images, physical material properties like color, translucency, the ability to repel or attract water, and even the ability to change shape will be controllable by the user, mediated by embedded sensors and computers. These new advances are beginning to be brought about by accomplishments in the sciences and engineering that would not be possible without their deep interdisciplinary collaboration. The outcome of this work will have considerable implications on what the

world looks like in the coming decades. At that time, physical designers will have a greater ability to design the materials themselves, not just the physical artifacts that the materials are used to make.

We will also see fields of design continue to evolve beyond their traditional silos. Just as physical designers have crossed the boundary into digital experiences from atoms to bits to create broader, richer user experiences (UX), digital and interaction designers will similarly be able to design the UX of changeable physical materials and products. Bits will control atoms.

INSPIRED BY NATURE AT A SMALL SCALE: MATERIAL PROPERTIES FROM MICROTEXTURES

In recent decades, life scientists around the world have made astonishing discoveries about how nature endures in the most challenging environments by evolving high-performance physical "technologies" that operate on a microscopic and even molecular level—and all without damaging the environment. As recently as the past decade some researchers have uncovered how these microtextures on the skins and bodies of living organisms in the plant and animal kingdoms can be adapted or applied to human problems. Let me give you a few examples.

Sharklet Technologies, a startup company based in Aurora, Colorado, has discovered that shark skin is often composed of microundulating scale structures called *denticles* that perform two remarkable functions at once: they prevent bacterial colonization and improve hydrodynamic performance. At the outset of Sharklet's breakthrough research for the United States Navy, founder Dr. Anthony Brennan of the University of Florida was exploring alternative solutions for cleaning algae from the hulls of warships.[1] This was a huge concern because buildup of algae impacts cruising-speed performance and is costly and time consuming to remove in dry dock. Additionally, toxic biocidal chemical treatments are frequently used to remove the algae, which the Navy was under increasing pressure to abandon. Dr. Brennan discovered that shark skin has a microtexture that bacteria and other microorganisms could not stick to, and as a result, bacterial colonies could not form. This helps reduce drag and increase swimming performance, which give sharks an advantage over their prey while also helping them to

1 Sharklet (*http://www.sharklet.com*)

dodge predators. When this phenomenon was translated into a prototype of a new type of material, it not only met the Navy's goals, but it also offered the promise of an additional, more significant application: antifouling of this sort could be useful in hospitals where infectious diseases are frequently spread. Sharklet Safe Touch was born, proving advantageous over chemical spray solutions that had been traditionally used in healthcare environments by *preventing* bacterial colonization rather than killing it after the fact. Killing bacteria with germicides has the distinct disadvantage of creating resistant super bugs such as MRSA and others.

Lotusan paint, developed by STO in Germany, is another example of a product for which microscopic texture yields a specific benefit based on a biological model. As the name suggests, Lotusan was inspired by the intricate surface texture of Lotus leaves, which have long been known for their ability to shed water and dirt. Reproducing the small, imperceptible hydrophobic texture drove the design and development of a paint, which naturally repels dirt from surfaces, an attribute that is very useful in public spaces that need to be cleaned and maintained, usually with toxic chemical detergents and monkey grease. Lotusan also reduces the amount of water and energy exerted to maintain surfaces, which is a welcome quality for cash-strapped municipalities.

Unfortunately, the production of Lotusan paint is also complex and expensive due to the scale of the physical features required to enable the hydrophobic effect.[2] At Harvard's Wyss Institute for Biologically Inspired Engineering, a team of researchers led by Dr. Joanna Aizenberg has developed a potential improvement in a material called SLIPS (Slippery Liquid-Infused Porous Surfaces), which is able to do many of the things Lotusan can and more.[3] In addition to water, SLIPS can repel oils and other liquids. It is modeled after the surfaces of a Nepenthes Pitcher Plant flower, which captures insect prey (and eats them!) by causing them to slip and fall into a chamber where they are unable to escape. Just as with Lotusan, when used on fabrics, prototypes of coatings that use a microscaled porous, textured solid—in this case infused with lubricating film—have demonstrated the ability to repel wine, blood, and every other imaginable liquid that could stain or

2 Lotusan (*http://www.lotusan.com*)

3 Wyss Institute of Biologically Inspired Engineering (*http://wyss.harvard.edu*)

congregate on a surface. Imagine children's clothing that would never stain! Another advantage of SLIPS, according to Aizenberg, is that the effect can be created by using existing materials and continues to function even after being scratched or abused.

In contrast, other examples of microtextures and features have been found that promote "dry adhesion" or stickiness without a chemical substrate. Biologist Robert Full of the University of California, Berkeley studied how geckos can climb flat walls without falling as humans would. At the core of this superhuman ability are nanometer scale keratin hairs (setae) on their toes that adhere to surfaces by means of intermolecular forces. To substantiate this hypothesis, Dr. Full asked Stanford engineering professor Mark Cutkowsky to develop a robot and subsequently a human climbing suit based on the principles of gecko toes. The demonstrations of this remarkable technology (which you can see on YouTube) have inspired other research teams to investigate other superhuman animal qualities for their potential commercial application. However, later follow-up articles about the gecko technology have suggested that the robots only work on clean surfaces and would require greater finesse to work on more uneven or textured surfaces. Smaller hairs like those actually on the gecko's feet would help, but again the tiny scale and reliability of these hairs are difficult to reproduce and maintain. In the meantime, the technology is being translated into a potential reusable dry adhesive.

Maintenance is certainly a concern with microscopic-scale features that are delicate and potentially broken. Whereas complex natural organisms have the biological means to regrow fine hairs on a gecko toe, regenerative materials are an entirely new level of complexity to challenge human ingenuity. Notwithstanding, successful experiments have been made to create regenerative materials such as self-healing concrete that is able to fill its own small cracks with resin to prevent fractures from becoming bigger problems requiring costly maintenance.

These are only a few of a growing library of examples of biologically inspired microtextures and features that could spur on innovative materials and design in exciting and impactful ways. But before this can happen, in many cases economical manufacturing capability needs to catch up with the discoveries being made. Biological function frequently operates at a smaller (microscopic), more intricate scale, which is difficult to reproduce reliably in large quantity using current

methods of manufacturing. Traditional production methods—plastic injection molding, casting, milling, and machining—are limited to a certain level of scale, detail, and resolution, but the technology is changing quickly. With recent rapid advancements in the scale and resolution of additive manufacturing, the potential to emulate these biological properties will likely soon be feasible.

Emerging Frontiers in Additive Manufacturing

Popular media coverage about additive manufacturing—or as it's more commonly known, 3D printing—has produced tremendous excitement and speculation about what it will mean in the future when everyone has access to it. The current reality is that consumer 3D printing is just in its infancy with limited capabilities. Complex shapes can be created, but only out of a few solid and elastomeric-based materials at a time and at a low resolution. This might help you custom design and print a plastic smartphone case but it is far from being able to print a smartphone itself, with a display, battery, printed circuit board, and other materials layered and assembled together. But, what if 3D printers could use a wide assortment of different materials, from plastics and electronics to living cells and semiconductors, mixing and matching the materials with microscopic precision? The ability to print all of these materials is currently being explored in labs across the world, and the abilities of the technology are changing with increasing rapidity.

Materials scientists, such as Dr. Jennifer Lewis at Harvard's Wyss Institute, are developing the chemistry and machines to make this kind of multimaterial 3D printing possible. She prints intricately shaped objects from "the ground up," precisely adding materials that are useful for their mechanical properties, electrical conductivity, or optical traits. This means 3D printing technology could make objects that sense and respond to their environment. As Lewis says, "Integrating form and function is the next big thing that needs to happen in 3D printing."[4]

Others are taking a different approach by trying to use the process of additive manufacturing itself to create a variety of performance properties from one material through the arrangement, topography, and structure of a single material. Neri Oxman, a veritable Renaissance

4 Jennifer Lewis, Wyss Institute (*http://wyss.harvard.edu/viewpage/412*)

woman at MIT with degrees spanning Architecture and Computational Design, has been exploring the synergy of biological approaches to creating structure and 3D printing. Rather than relying only on different material chemistries to produce desired properties, some of her work is placing the onus on surface topography to produce the desired material properties. Carpal Skin, an experimental carpal tunnel therapy glove, is 3D printed for a patient according to the "pain map" that individual is experiencing. The pain map then corresponds to the surface geometry of the glove, offering flex and support tailored to the user's condition.[5]

Also at MIT, Dr. Markus Buehler has been investigating how different material properties can be encoded at a molecular scale by using basic chemical building blocks. In his words:

> Proteins are the main building blocks of life—universally composed of merely about 20 distinct amino acids—realize a diversity of material properties that provide structural support, locomotion, energy and material transport, to ultimately yield multifunctional and mutable materials. Despite this functional complexity, the makeup of biological materials is often simple and has developed under extreme evolutionary pressures to facilitate a species' survival in adverse environments. As a result, materials in biology are efficiently created with low energy consumption, under simple processing conditions, and are exquisite as they often form from a few distinct, however abundantly available, repeating material constituents.[6]

The significance of Dr. Buehler's work here lies in understanding how material properties could eventually be designed through their molecular arrangement and then fabricated to meet a desired human application. Being able to create materials in this manner from chemically benign building blocks could revolutionize material science.

Micro Manufacturing

So far, we have shown how science has advanced considerably in understanding how materials can deliver remarkable performance properties, but this is only half of what will lead to creating new materials.

5 Neri Oxman, Mediated Matter Group (*http://www.media.mit.edu/research/groups/mediated-matter*)

6 *http://web.mit.edu/mbuehler/www/*

Additive manufacturing at a nano scale is still very experimental, so the kinds of material construction that Dr. Buehler's work suggests is still far from being feasible to produce in any mass volume. But, the ability to manufacture at a smaller scale continues to develop. Newer additive manufacturing techniques such as Micro Laser Sintering (MLS) are pushing the boundaries of small-scale production to the level of micrometers and smaller. These have been used to produce insect-sized flying robots and microfluidic medical devices, but they also have been used to experiment with microtexturing and self-assembled structures. Going back to the example of shark skin, researchers under Dr. George Lauder at Harvard's Wyss Institute have just recently managed to scan and recreate the shark skin's denticles for the purpose of hydrodynamic performance. Different from Sharklet, the goal was to re-create an array of the denticles' geometry at actual scale to see if it would perform. Using a state-of-the-art 3D printer that is capable of printing multiple materials simultaneously at a tiny scale, Dr. Lauder's team succeeded in achieving a 6 percent efficiency increase with their prototype when compared against a control model without the texture.

Dynamic Structures and Programmable Matter

On an architectural scale, designers have been keenly interested in the ability to change the shape or properties of a building in some manner to respond to environmental conditions. Sun load in particular is a large source of interior heat generation and is usually counteracted with mechanical air conditioning systems, at enormous energy cost. This inspired architects such as Achim Menges (University of Stuttgart)[7] and Doris Sung (University of Southern California)[8] to explore passive dynamic facades that open and close in response to humidity and heat load, respectively, to allow (or prohibit) water and air from passing through. Yet there have not been widely successful efforts yet to control external façade systems like these with automated intelligent systems—although many are trying.[9] One example developed by kinetic

7 Achim Menges, Center for Design Computation, University of Stuttgart (*http://icd.uni-stuttgart.de/?cat=6*)

8 Doris Sung, USC (*http://arch.usc.edu/faculty/dsung*)

9 French architect Jean Nouvel's celebrated Arab Institute in Paris (1988) employed a façade system composed of an array of electronically controlled oculi, like camera lenses, to control the heat gain in the building from daily sunlight (*http://www.imarabe.org*).

sculptor Chuck Hoberman's Adaptive Building Initiative called *adaptive fritting* hints at least at the possibilities of this kind of dynamic intelligent control.[10] Many more projects like these are undoubtedly in the works and are bound to become more prevalent with the accessibility of electronic prototyping tools such as Arduinos and Raspberry Pis.

It is especially exciting to think about the possibilities of creating dynamic computer control of physical materials at the micrometer scale. Again, biology has provided some of the vision of what these kinds of complex dynamic materials could do. Biologists have been fascinated by the changeable behaviors of certain organism's bodies and skin. Bioluminescent organisms from the deepest parts of the ocean are able to control their luminescence on demand for communication and protection. Similarly, several organisms, including the octopus, have the ability to change the appearance of their skin entirely for the purpose of camouflage. Roger Hanlon and David Gallo, scientists at the Woods Hole Oceanographic Institution, along with other researchers, have begun to learn how these underwater creatures are able to achieve this effect.[11] It is a complicated system, but one that the United States Defense Department and fashion designers alike would be interested in emulating: the ability to change the color, texture, and shape of your clothing to match your environment or to change your appearance to fit any occasion.

Similar to the processes of an octopus, technology giant Qualcomm developed a digital screen technology called Mirasol based on the controllable light reflectance behaviors of butterfly wings. Unlike energy intensive LCDs, Qualcomm explored how reflecting light using micromirrors could produce color. In this case, electric charge is used to control the angles of these "mirrors." After several years of development, Qualcomm made the decision to shelve the technology for a familiar reason: the challenges involved in production made the cost too high and unreliable. But, just as we saw in the examples earlier, rapid advancements in additive manufacturing scale and resolution combined with programmability will reduce the barriers to producing such a technology.

10 Chuck Hoberman, Adaptive Building Initiative (*http://www.adaptivebuildings.com*)

11 Woods Hole Oceanographic Institution (*http://www.whoi.edu*)

At present, there are numerous experimental frontiers in *programmable matter*, which we will begin to see realized and manufactured. Research teams at MIT, Carnegie Mellon University, Cornell, and other universities have been pushing the boundaries in this field. Part science fiction (think of the T-1000 model android in *Terminator 2*) but also a tangible reality, these efforts endeavor to build structures from the micro scale and up that can fold, shape-shift, and otherwise reconfigure their form from the bottom up to fit any number of applications. There are many new efforts in this area: at Carnegie Mellon, a programmable matter proposal called "Claytronics" (*http://www.cs.cmu.edu/%7Eclaytronics/movies/carDesign_12_vo_H264.mov*) is being explored for collaborative design visualization applications. In this wild concept, physical objects themselves would be able to transform into different shapes according to human or environmental input.[12] Teams working in product development could use this tool to collaborate in the refinement of a physical design. One possible near-term application for this kind of technology is so called "morphing wings," which are being researched by NASA. This technology explores the flight control potential of airplane wings that respond to flight conditions by transforming fluidly from one airfoil form to another. Rather than pivoting like a swept wing, morphing wings could optimize their form in limitless ways.

In a similar defense-related contract, but with a more tangible result, an interdisciplinary team of researchers from MIT and Harvard, working in conjunction with the United States Defense Advanced Research Projects Agency (DARPA) have been developing folding origami structures that self-fold and assemble into different physical configurations under computer control. The significance of projects like these suggests a future wherein human-made structures could dynamically fold and change shape—imagine a dynamic building façade with embedded solar panels that can simultaneously track the sun while changing shade levels to control heat gain. This is the future of intelligent matter.

12 Claytronics, Carnegie Mellon University (*http://www.cs.cmu.edu/~claytronics/*)

Connecting the Dots: What Does Intelligent Matter Mean for Designers?

As we have just seen, many of these developments between biology, manufacturing, and computing lead toward a new era and new definition of what it could mean to be a designer of "physical things." Just as societies have progressed through ages based on mastery of materials—the Stone Age, Bronze Age, and more recently, the Plastic Age—perhaps we are seeing even more evidence that we stand at the dawn of a new age. Instead of being defined and constrained by a material, in the decades to come we stand to define the materials that will be all around us. Could this be the dawn of the Intelligent Materials Age? If this is so, where does the designer fit into this new age, especially the designer of physical artifacts? Let's consider some possible scenarios about how the designer might approach their field.

DESIGNING MATERIAL BEHAVIORS

Imagine buildings with dynamic structures and programmable, functional material properties such as the aformentioned Lotusan or SLIPS being applied throughout the structure's environment at every scale. Let's examine a couple of potential applications.

Rainwater management in civic spaces and building exteriors

What if every expansive surface on a building could be optimized to control how water behaves? Depending on how much it has rained or snowed, buildings might want to adapt dynamically to how they react to water. If rain has been sporadic, perhaps a building would want to capture the water and store it for internal use. Maybe the roof and gutters could mechanically expand or unfurl, similar to a morphing wing, to collect more water. On the other hand, if it has been raining normally or perhaps excessively, roof materials would be designed to shed the water quickly by increasing the hydrophobicity to minimize leaks. Wall surfaces could repel water and dirt and never need cleaning; subway stations and underpasses could be lower maintenance and (mostly) graffiti-free. Indoors, some of the most reviled rooms could also be lower maintenance or even maintenance free, such as the bathroom.

Building interior spaces and water management

Imagine maintenance-free bathroom surfaces for showers, tubs, sinks, and even toilets! By maintenance-free, I mean surfaces that would not need to be scrubbed: vertical surfaces or tiles would be manufactured with hydrophobic microtextures to deter water minerals, soap, and other residue from collecting. Other banes such as tile grout mildew would have a harder time growing if these surfaces would dry more easily on their own. Toilets and sinks could maintain their clean appearance with less water use to clean them. Think about what it would mean for our health and watershed if we could drastically cut down on the harsh chemistry and water required to clean a bathroom.

Shower floors would present a slightly different challenge. A shower floor would need simultaneous or changeable properties—the ability to shed water on one hand, but also provide grip for safety. By rearranging the microtexture of the floor, grip level might be something that could "turn on" the same way you turn on the hot water or the lights. Perhaps the floor's "grip mode" would activate when a sensor detected a person in the shower space.

Because these textures are so small and invisible, how would a designer communicate the presence of these attributes? Although advanced additive manufacturing will be able to custom texture materials to control these performance qualities, the designer's task in this case would be to communicate that grip is "on" or "off." Rather than using an obvious indicator like a red light to communicate that a floor is slippery, how could a programmable material communicate physically that it is slippery even if you cannot "see" slipperiness?

Similar programmable intelligent materials with on-off modes could find practical applications in the kitchen. Consider, for example, the design of dishware, flatware, and cookware that is often cleaned in dishwashers. Many types of plastic, metal, and ceramics (such as ceramic tile in the shower) retain residual water on their surfaces. After a dishwasher cycle is complete these products sometimes need to be hand dried, which could be a time-consuming extra step. Commercial chemical rinse agents such as Jet-Dry solve this issue but have not been tested conclusively for toxicity. In the age of intelligent matter, this variable of hydrophobicity or water dispersal might be able to be tuned in or turned on, and design decisions will have to be made to balance or find the right places to do this. If placed on the outside of a bowl, water

could disperse, but the designer might not want to coat the inside of the bowl in order to avoid liquid foods such as soup or cereal sloshing out and making a mess. Perhaps a dishwasher might be able to activate a hydrophobic texture in a wash cycle using a magnetic field, which would allow residue to disperse in the washer without interfering with the eating experience during mealtime.

DESIGNING MODES FOR PHYSICAL PRODUCTS

Beyond water management, physical products of all kinds will have the opportunity to adopt dynamic behavior. More than the bimodal scenarios discussed in the last section, products could have multiple, if not limitless, states and modes. Like the shape-shifting ability of the octopus, what if our shoes could "adapt" to different weather conditions, seasonal activities, and social occasions, freeing us to own fewer pairs? If it were raining heavily outside, perhaps your shoes could adapt to be impermeable to water like a rubber boot or perhaps be hydrophobic to drive water away. If it were hot out, the shoe could become structurally more porous to allow your feet to breathe better. Perhaps the color and texture could also change to reflect the context and formality of the social environment. If it were to snow, the sole of your shoes could change texture to provide traction on slick surfaces.

DESIGNING PHYSICAL BEHAVIORS IN PHYSICAL PRODUCTS

Thomas Heatherwick, principle of Heatherwick Studio, has a distinct body of work along with some other pioneering firms that have begun to explore the unique considerations involved in designing physical design behavior. One of his iconic commissions, Rolling Bridge (2004), a pedestrian drawbridge in London, is unremarkable in its open and closed states, but when it moves, it all changes. Watching the lobster-shell design roll up or unroll is both surprising and remarkable to behold. As design and architecture become more expressively dynamic such as this bridge, designers will need to consider how a structure like this opens, not just that it opens, and how it needs to look or be constructed. You can make a bridge appear friendly and trustworthy in form and materials, but how can the motion of its unfurling build confidence in a user? How can you make the nuances of motion appear friendly or inviting? Could the bridge make you laugh? Could it slowly accelerate, or speed up and then decelerate to a gentle stop?

In the same way that interaction designers create digital experiences that behave in context or brand-appropriate ways (think of the slow, "breathing" pulse of an Apple power indicator light when a computer is sleeping), product designers will also be faced with the opportunity to bring these dynamic behaviors to the physical world. This is where the act of being a physical designer will surely evolve. In designing the personality with which a transition is made between modes, physical designers will have to think more like animators, choreographers, or any other design field involving motion.

One area in which motion plays a critical role today is in the behavioral design of safety lighting, such as that on trains, planes, and emergency vehicles. In the 1950s, the average police car had a single, slowly revolving light indicating engagement in a pursuit—hardly urgent in its behavior and not terribly good at attracting attention. Today, police cars and ambulances use fast moving, abrupt, pulsating LEDs and sound bursts to capture your attention. Beyond the bright lights and loud sounds, it is the motion and transition of these cues that define a behavior suitable for emergencies.

In a recent industrial design studio, I challenged my students to create a piece of safety equipment for bicyclists to help promote better visibility on the road. Of the many different approaches the students took, one concept emerged that I believe forecasts considerations designers will face in the near future. Called the "puffer jackets," the student envisioned a vest that would emulate the behavior of a puffer fish, which uses physical parts of its body to startle and repel predators. The design proposed electronically controlled mechanics as used in experimental fashion design by Studio XO and Hussein Chalayan. The design would inflate an area of the jacket on command with reflectors, thereby making the cyclist look bigger and more noticeable (this was a short conceptual user experience project). In the ensuing development of this idea, the class reached a significant conclusion: what really mattered in the design was less the visual quality of the vest and more the motion behavior transitioning from a normal state to the attention-grabbing mode. In a future of dynamic intelligent matter, these kinds of considerations will only continue to grow in importance.

Conclusion

Much of what I've been discussing has focused on technical possibility and, to a lesser extent, ecological implications of the future of materials and what it means for design. Of course, no design should proceed just because it is simply novel and feasible—design should always be concerned with what *should* be done. Is it desirable? Does it fulfill a need? Is the world going to be better off with it? That said, altogether the implications ahead of these technologies are huge. In an age of intelligent matter, physical design will no longer be three-dimensional and static. The fourth dimension, time, and behavior will come to the physical tactile world just as it has existed in the digital realm to date, and designers will need to think about the possibilities and opportunities to create meaningful user experiences driven by these new parameters.