

# Designing Better UX

A large, stylized, red letter 'U' is positioned in the bottom right corner of the image. The background is a vibrant red with a network of white lines and nodes connecting various points. There are also several overlapping, semi-transparent red circles of different sizes in the lower half of the image.

# Imprint

© 2013 Smashing Magazine GmbH, Freiburg, Germany  
ISBN: 978-3-94454051-1  
Cover Design: Ricardo Gimenes  
PR & Press: Stephan Poppe  
eBook Strategy and Editing: Vitaly Friedman  
Technical Editing: Cosima Mielke  
Planning and Quality Control: Vitaly Friedman, Iris Lješnjanić  
Tools: Elja Friedman  
Idea & Concept: Smashing Magazine GmbH

## *Preface*

There are numerous traps that designers can walk right into while designing. Even the most thought-out and best-planned experience can get lost when attention to certain details falls short. To raise awareness for those little things that add up to an ideal user experience, we have put together this eBook.

Practical approaches for designing better UX are presented in this eBook — not only for the Web, but also for mobile apps, adaptive systems, and great multi-screen experiences. “Designing Better UX” is packed with in-depth tips and tricks shared by experts from the industry. They consider the importance of straightforward yet personal microcopy, explore the strengths and weaknesses of infinite scrolling, and take a look at how well-crafted feedback loops can influence a user’s behavior.

Once we decide to create a well-balanced experience, all of the covered aspects in this eBook have to be thoroughly considered, i.e. when is the user experience fostered and when is it curbed? If you’ve been looking for a valuable resource for your daily UX routine, you can now stop searching and start reading.

## TABLE OF CONTENTS

When You Shouldn't Use Fitts's Law To Measure User Experience .....	4
Five Ways To Prevent Bad Microcopy .....	19
Infinite Scrolling: Let's Get To The Bottom Of This .....	28
Designing Great Feedback Loops .....	41
Sketching For Better Mobile Experiences.....	54
Converting Our Stories Into Multi-Screen Experiences.....	65
Creating An Adaptive System To Enhance UX.....	77
About The Authors .....	92

# When You Shouldn't Use Fitts's Law To Measure User Experience

BY ANASTASIOS KARAFILLIS 🍷

The key statement of Fitts's Law is that the time required to move a pointing device to a target is a function of the distance to the target and its size. In layman's terms: the closer and larger a target, the faster it is to click on that target. This is easy to understand, not too difficult to implement and it doesn't seem to make much sense to contradict such a simple and obvious statement.

However, before you start applying Fitts's Law on every single pixel you can find, consider a few problems that might arise for you as an interaction designer.

## *Fitts's Rule Number 1: Create Larger Targets*

The likely most prominent statement derived from Fitts's Law is that the *larger* a target, the faster it is to acquire.

### THE BENEFITS

Creating larger targets will facilitate interaction as well as allow you to get the most pixels out of your interface.

For example, some websites do not extend the clickable area of a button or link to the entire target. As a result, more precision is required to move the cursor to the respective link, which leads to slower navigation times. Fitts's Law would suggest utilizing every available pixel<sup>1</sup> to enlarge the clickable area and thus making it a larger target to click.

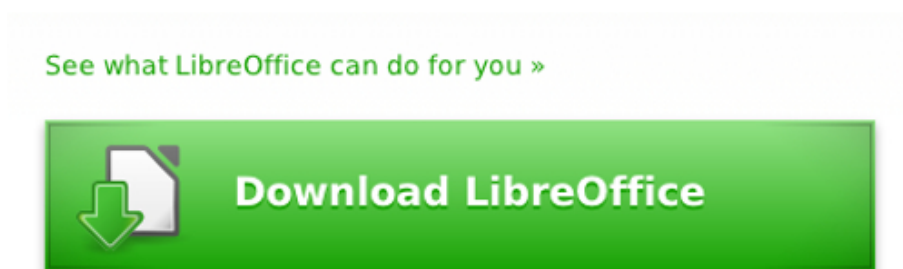
---

1. <http://msdn.microsoft.com/en-us/library/ms993291.aspx/>



*The target to the left lets a few pixels of screen real estate go to waste. The target to the right makes itself larger and quicker to click by exploiting every pixel at its disposal. (Left example: [Firefox](#)<sup>2</sup>, Right example: [Apple](#)<sup>3</sup>)*

Increasing the absolute or relative size of a button to make it easier to click is also a popular technique among designers for communicating priorities within the interface and prompting or calling users to perform a particular action.



*Relative and absolute size communicate priority within an interface. (Example: [LibreOffice](#)<sup>4</sup>)*

Although there are [many more considerations](#)<sup>5</sup> to be factored in when designing a call-to-action button, Fitts's Law provides one of its foremost theoretical foundations.

## THE PERILS

The downside of large targets is, of course, that they can break the balance in your interface as well as quickly take up screen real estate. However, even if you have plenty of space to spare, you do not have to constantly enlarge your target areas to make them more usable since the predicted usability of the size of a button progresses in a non-linear fashion<sup>6</sup>.

---

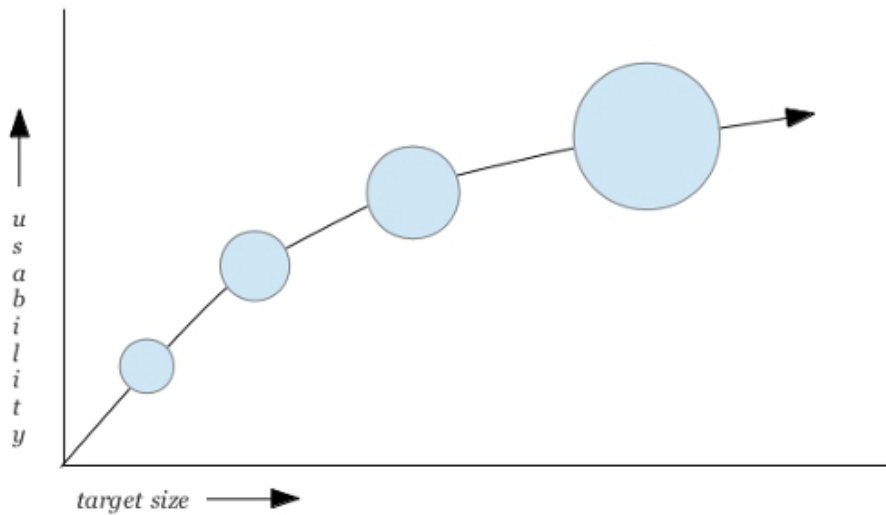
2. <http://support.mozilla.org/en-US/products/firefox/get-started>

3. <http://www.apple.com>

4. <http://www.libreoffice.org/>

5. <http://www.smashingmagazine.com/2009/10/13/call-to-action-buttons-examples-and-best-practices/>

6. <http://sixrevisions.com/usabilityaccessibility/improving-usability-with-fitts-law/>



*Good for your pixels: The usability of the size of a button is a non-linear relationship.*

Thus, if you make a small button 10% larger, it will become much more clickable, but if you make an already very large button 10% larger, you will not gain much more in terms of its usability.

## *Fitts's Rule Number 2: Minimize Cursor Movement*

A second statement one can deduce from Fitts's Law is that the *closer* a target, the faster it is to acquire.

### THE BENEFITS

If you place the links and buttons users are most likely to access on a regular basis next to each other, rather than distribute them across the interface, you will speed up interaction by reducing the amount of pixels the cursor will have to travel.

Consider, for example, the Ubuntu Unity<sup>7</sup> interface. It allows you to browse various sources and use two different filters on the results: a *text filter* and a *file type filter*. However, notice in the picture below how far apart the two filters are. While the text filter is placed at the very *top* of the screen, the file type filter is placed at the very *bottom*.

---

<sup>7</sup>. <http://unity.ubuntu.com>



*Fitts would disagree: Search filters are often accessed successively and should therefore be placed close to each other. (Example: [Ubuntu Unity](http://unity.ubuntu.com)<sup>8</sup>, Screenshot: [Webupd8.org](http://webupd8.org)<sup>9</sup>)*

For a fluent workflow, this arrangement is not optimal. When performing search queries, users often access the input field and file type filters in quick succession to adjust their search results. In order to do that here, the cursor would have to travel quite a distance. Instead, placing the file type icons next to the input field would have minimized cursor movement and sped up interaction (as well as freed up some vertical space).

## THE PERILS

Arranging elements strictly according to this formula can cause conflict with other important design principles, such as the principle of grouping and separating different classes of functionality or content. Its purpose is to give your interface a clear and consistent structure as well as increase its discoverability.

Notice in the picture below how the various tools are sorted into small, meaningful groups: in this case *table* tools to the left and *insert* tools to the right.

<sup>8</sup>. <http://unity.ubuntu.com>

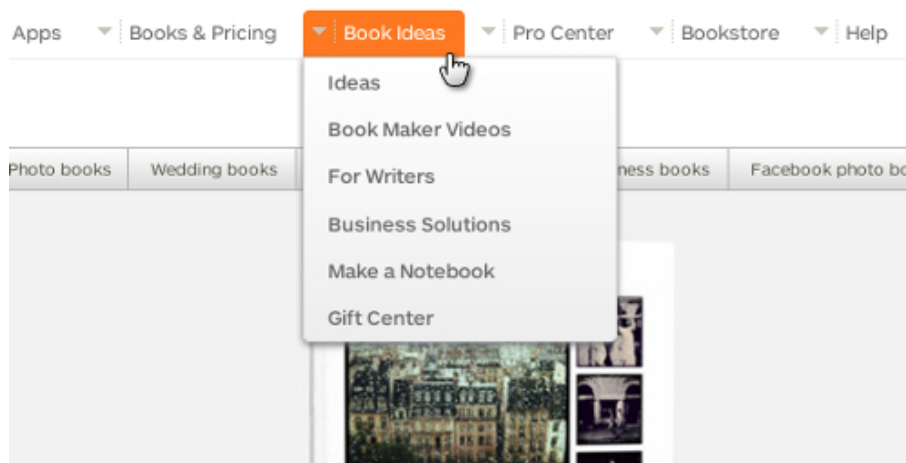
<sup>9</sup>. <http://www.webupd8.org/2012/04/ubuntu-1204-lts-released-see-whats-new.html>



Sorting interface elements into meaningful groups will give your interface a clear, consistent structure. (Example: [Numbers](http://www.apple.com/iwork/numbers/)<sup>10</sup>)

This allows users to create a familiar mental map of where to access a certain kind of information or tool. By contrast, if you analyzed the buttons purely by the frequency in which they are accessed, you would probably choose a different arrangement to minimize cursor movement. This, however, would break the functionally-consistent structure of your interface.

Another principle Fitts's Law can interfere with is the principle of providing a clean and tidy interface. In order to clean up their interfaces, many websites group their content into drop-down menus. Although many designers [raise objections](http://www.useit.com/alertbox/20001112.html)<sup>11</sup> against their usability (a discussion of which would go beyond the scope of this chapter), drop-downs are nonetheless acknowledged as a visually elegant and space-efficient way to unclutter the interface and organize its content.



Drop-downs can help you structure your content and unclutter the interface. (Example: [Blurb.com](http://www.blurb.com/book-ideas)<sup>12</sup>)

The reason Fitts's Law does not recommend the use of drop-downs is that they involve a longer cursor movement since users cannot access the target in a straight line. Instead, users will have to first click or hover over the drop-down menu, and then move the cursor down the list

<sup>10</sup>. <http://www.apple.com/iwork/numbers/>

<sup>11</sup>. <http://www.useit.com/alertbox/20001112.html>

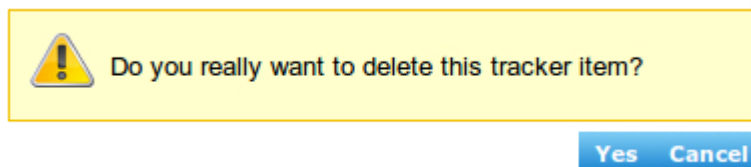
<sup>12</sup>. <http://www.blurb.com/book-ideas>



(and possibly across sub-menus as well) until they eventually reach the desired entry. But considering the benefits of drop-down menus, a longer cursor movement can be an acceptable trade-off.

A third, very important principle that may force you to defy Fitts's Law is the principle of building forgiving interfaces, which aim to prevent and minimize the costs of mistakes.

Fitts's Law suggests placing elements directly next to each other to minimize cursor movement, which would also save some space. However, saving those few pixels can result in users accidentally clicking the wrong item, especially when item boundaries are not easily discernible or when focused items are not highlighted distinctly enough.



*Accidents waiting to happen: the placement of interface elements can either cause or prevent mistakes. (Example: [Codebeamer.com](https://codebeamer.com)<sup>13</sup>)*

Be aware, though, that the consequences of mistakes are less severe when the elements represent navigational functionality as opposed to sharing or editing functionality.

If I open the wrong link, I can simply click the “Back” button to revert my mistake. Thus, when it comes to the links in your header or sidebar, there is no real harm in leaving no space between them.

When it comes to *navigating during consumption*, things can become more annoying. When playing video or audio files or displaying text files, accidentally clicking the “Stop,” “Eject,” “Next Item” or “Clear Playlist” buttons will require more effort on the users part to revert to the original state.

However, when it comes to *editing* and (especially) *sharing* functionality, mistakes can become downright destructive. If I click on a button such as “Send,” “Print,” “Delete,” “Download,” “Upload,” “Burn,” “Rip,” “Close,” “Shut Down,” “Connect,” “Disconnect,” “Accept” or “Decline,” my actions can be of much more severe consequences and may not be undone as easily.

Therefore, especially for elements with *editing* or *sharing* functionality, you should take precautions to minimize mistakes and the consequences of mistakes:

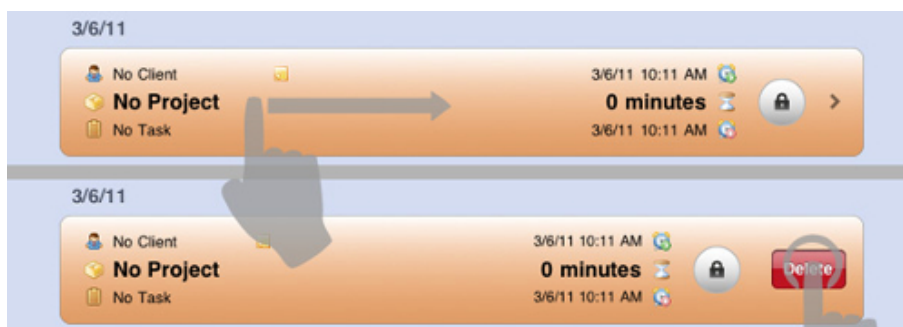
---

<sup>13</sup>. <https://codebeamer.com/cb/issue/35151>

- Offer easy ways to undo the performed action (one option: temporary undo buttons<sup>14</sup>).
- Try to leave a few pixels of space between the buttons.
- Make item boundaries clearly visible.
- Highlight focused items.
- Group buttons to minimize the damage if a mistake occurs<sup>15</sup>.

To give an example for the last point in the list: if I accidentally click on the “Get Mail” instead of the “Write Mail” button, the results are not dramatic, but do not place the “Reply” and “Delete” buttons next to each other.

A special type of input method aiming to prevent mistakes is the two-step input method. Two-step input methods are slightly less usable since the two steps entail a longer hand or cursor movement yet, as a consequence, they are more secure. The basic idea is that while you can accidentally perform both actions separately, it is unlikely that you will accidentally perform them successively. Take, for example, swipe-to-delete.



*First swipe, then delete. While each on its own can easily be triggered by accident, they become a fail-safe mechanism when combined. (Example: Timelogger App<sup>16</sup>)*

I can imagine myself accidentally swiping from left to right or accidentally pushing a button, but not accidentally performing both actions in succession.

Two-step input methods lend themselves for mobile devices, which are more prone to accidental inputs. However, they can also be more space efficient in that the second step doesn't need to be visible until after the first step has occurred.

<sup>14</sup>. <http://www.thepicky.com/internet/recall-email-that-is-send-undo-send-in-gmail/>

<sup>15</sup>. <http://www.codinghorror.com/blog/2010/03/the-opposite-of-fitts-law.html>

<sup>16</sup>. <http://www.mb-e.net/>

So, whether swipe/tap, swipe/swipe or tap/tap combinations, two-step input methods make things a bit more difficult than just pressing a large button, but the inconvenience adds a little bit of security that is sometimes necessary to safely get your work done.

### *Fitts's Rule Number 3: Avoid Muscular Tension*

The goal of Fitts's index of performance (PDF) is to quantify the information capacity of the human motor system. In other words: it aims to rank input methods according to the amount of physical effort they require to execute a certain command.

#### **THE BENEFITS**

The benefits of effortless input methods are most obvious when working with cumbersome devices. The most prominent example is vertical touchscreens, which are typically deployed in professional environments to create, visualize and manage large sets of data.

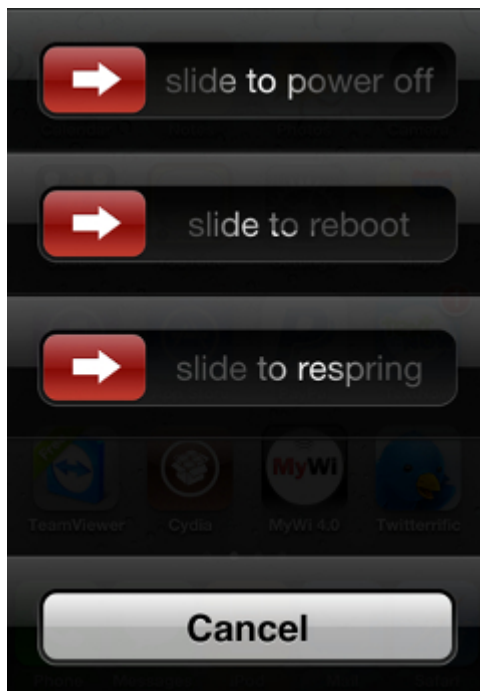


*Fitts's Law can facilitate and prolong interaction with vertical touchscreens. (Example: Perceptive Pixel)*

When working with vertical screens, keeping your arms in an upright position can quickly tire the deltoid muscles and cause input mistakes or force the user to abandon the interaction. Therefore, avoiding complex and strenuous input techniques can facilitate and prolong the interaction with those devices.

## THE PERILS

Input methods that are more difficult to perform can sometimes actually prevent mistakes. For example, mobile devices are often carried around in pockets, which can trigger commands by accident. In those situations, high-precision input methods are deployed, which use a higher input difficulty to make sure that a command is not executed accidentally. However, these input methods are also a way of making users aware of the severity of the command. Take, for example, the way an iPhone is turned off:



*Choosing UI elements by the severity of their consequences: Slide controls for dangerous commands, buttons for harmless ones. (Example: iPhone<sup>17</sup>, Screenshot: Outsideinnovation.com<sup>18</sup>)*

Powering off or rebooting the device are quite weighty commands; once triggered, they cannot be undone. Therefore, they are made into

<sup>17</sup>. <http://www.apple.com/iphone/>

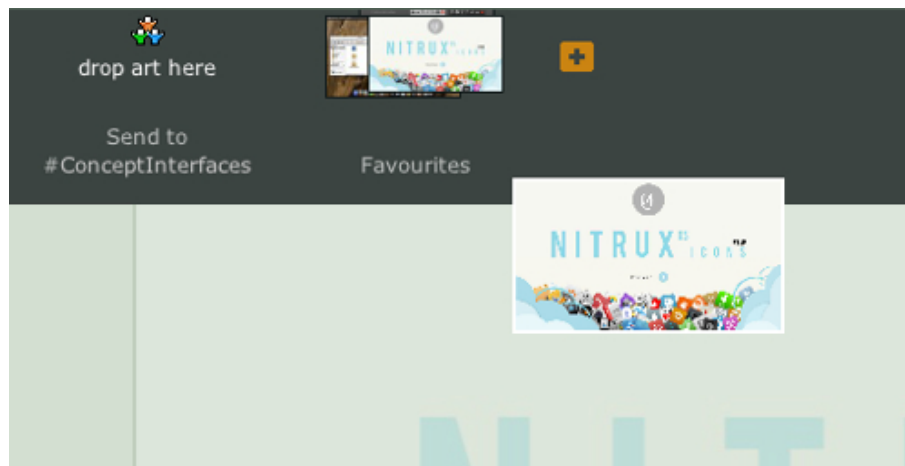
<sup>18</sup>. <http://outsideinnovation.blogspot.com/pseybold/2010/12/my-mortifying-ipad-experience.html>

sliders, which require a higher precision. By contrast, the cancel command is of no comparable consequence here; hence it is made a button.

Slide controls and other gestures that require a higher precision are the most secure but also the most tedious input methods. Therefore, in order to balance security and usability, they are usually reserved for dangerous commands that are executed infrequently, such as unlocking the screen, turning off the device, setting system-wide preferences, performing administrative tasks or silencing wake up alarms. When dangerous commands are supposed to be executed *quickly* or *frequently*, for example when editing, deleting or transferring items, a well-spaced icon arrangement or two-step input method are more appropriate. Although they do not offer the exact same degree of security, they are still fairly secure yet much easier to perform.

A second reason to implement more awkward input methods is to take advantage of the space-efficient nature of gestures. According to Fitts's index of performance, gestures, which involve some degree of dragging, require a higher muscular tension<sup>19</sup>, which is why Fitts's Law favors pointing-and-clicking. However, the advantage of gestures is that they trigger functionality without requiring UI controls.

Take, for example, the way you manage your art collection at deviantART<sup>20</sup>. In order to add an item to your Favorites, you do not have to press a button. Instead, as soon as you start dragging a picture, a pane is displayed which you can drop it onto.



*Dragging-and-dropping provides functionality without needing visible elements to trigger it. (Example: deviantART<sup>21</sup>)*

---

<sup>19</sup>. <http://www.billbuxton.com/fitts91.html>

<sup>20</sup>. <http://deviantart.com>

<sup>21</sup>. <http://www.deviantart.com/>

Since dragging-and-dropping doesn't require buttons or other UI elements to trigger that functionality, it is very space efficient. The downside, of course, is that gestures do not offer any obvious visual clues as to their existence (except perhaps through tooltips). Nonetheless, if screen real estate is crucial, these input methods, although more difficult to perform, become almost a necessity.

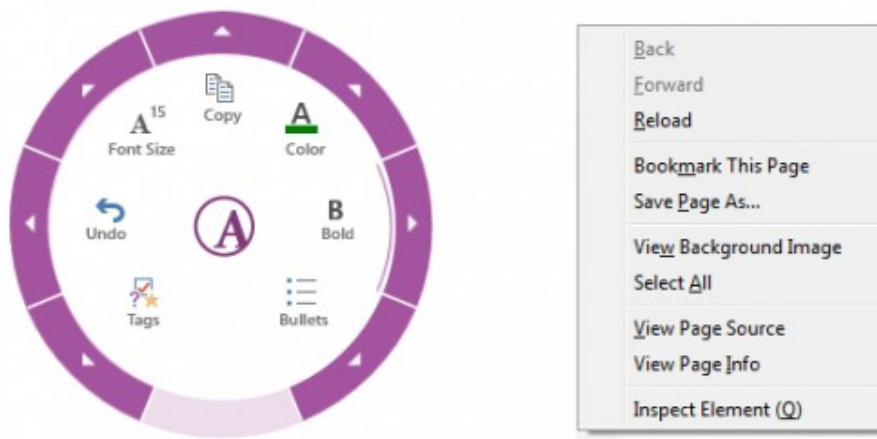
### *Fitts's Rule Number 4: Exploit The Prime Pixels*

The concept of prime pixels states that some pixels are faster to acquire than others. Corners and edges of the screen are especially fast to access. However, the fastest-to-acquire pixel in any situation is simply the pixel at the current cursor position, which has led to the introduction of the right-click context menu into human-computer interaction.

#### **THE BENEFITS**

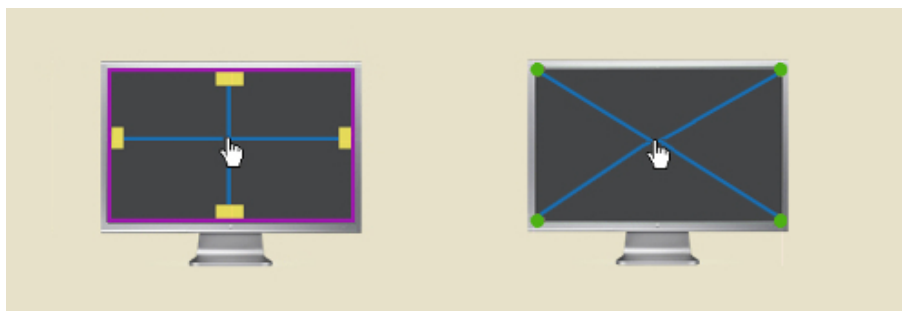
Context menus appear at the very pixel you right-click and provide you with context-sensitive options for the selected item. As a result, you do not have to move the cursor to a possibly distant fixed location in the interface. There are two kinds of context menus: linear menus and radial or pie menus.

Consulting Fitts's Law will reveal that it favors radial menus over linear menus. The reasons: First, the wedge-shaped menu entries offer larger target areas. And second, because the menu has a circular shape, the pixels the cursor has to travel to reach either menu entry are always identical. This consistency allows users to create a more efficient muscle memory. By contrast, in linear menus only the menu entries closest to the initial cursor position are quickly accessible, which is why the most frequently performed actions should have those spots reserved for them.



Fitts's Law favors radial over linear menus. (Left example: OneNote 2013<sup>22</sup>, Right example: Firefox<sup>23</sup>)

The benefits of placing items at the corners and edges of the screen are that the screen frame guides and positions the cursor once it reaches that location.



Corners and edges are prime screen real estate. (Diagrams: Particletree<sup>24</sup>)

The user could basically throw the cursor to a corner or edge to select the respective item without having to readjust the cursor position. The result: the targets will be much easier and faster to click.

## THE PERILS

Empirical studies confirm the theoretical assumptions<sup>25</sup> about radial and linear context menus, stating that seek time and error rates give the former a slight edge over the latter. Yet when the participants were asked purely about their subjective preferences, the pie menu was not favored anymore.

<sup>22</sup>. <http://www.microsoft.com/office/preview/en/onenote-2013-preview>

<sup>23</sup>. <http://www.mozilla.org/en-US/firefox/new/>

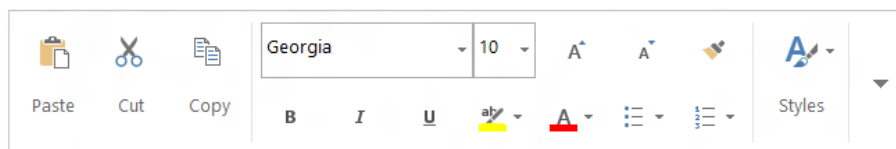
<sup>24</sup>. <http://www.particletree.com/features/visualizing-fittss-law/>

<sup>25</sup>. <http://www.donhopkins.com/drupal/node/100>

In fact, the pie menu, although favored by Fitts's Law, does have a few disadvantages that can outweigh its benefits in certain situations.

One issue is that the circular menu shape quickly leads to small target areas when more menu entries are added. One way to deal with this problem is to remove redundant options, in line with Hick's Law<sup>26</sup>. For example, if menu entries are not only applicable to the selected item, or if they are already accessible somewhere else in the interface, they do not have to be incorporated into the context menu as well ("Cut," "Copy" and "Paste" always apply solely to the selected area as opposed to "Undo," "Redo," "New File," "Save File," "Print File," or "Zoom In/Out" and thus lend themselves as fixed toolbar items).

A second way to manage a large number of options are sub-menus. While this is possible within radial menus, doing so will quickly clutter the screen and make them appear less organized than traditional linear menus. This is related to a very specific advantage of linear over radial menus: linear menus make it easier to express hierarchies via sub-menus as well as to group entries.



Grouping entries is more easily done in linear menus. (Example: Word 2013<sup>27</sup>, Screenshot: PCPro.co.uk<sup>28</sup>)

Finally, circular menus take up more space. This can lead to two problems: they can obscure selected items, and they are more likely to pop up at places other than the current cursor position when triggered close to screen edges.

Therefore, in summary, you should consider a linear context menu over a radial menu if:

- You have to integrate many options,
- You have to work with sub-menus,
- You want to group and rank menu entries,
- Screen real estate is critical.

<sup>26</sup>. <http://uxdesign.smashingmagazine.com/2012/02/23/redefining-hicks-law/>

<sup>27</sup>. <http://www.microsoft.com/office/preview/en/word-2013-preview>

<sup>28</sup>. <http://www.pcpro.co.uk/blogs/2012/07/17/word-2013-review-first-look/>



And finally, as to the corners and edges of the screen, two potential problems should be mentioned when working with mouse-operated devices. On large screens, the amount of pixels the cursor will have to travel can somewhat offset the aforementioned benefits. Also, Web designers will not be able to benefit from this rule because their content (except when in full-screen mode) is run in a browser window. As a result, they cannot take advantage of the edges of the screen and will almost necessarily have to opt for a more compact, centered layout.

However, when working with devices that are *not* mouse-operated such as touchscreens and styli, placing interface elements at the corners and edges of the screen will not only yield no positive results as to speeding up interaction. It can even have a detrimental effect. On large screens it would require users to constantly stretch their arms, which can tire them very quickly. Consequently, frequently accessed tools should be made into freely movable objects on large devices. Thus, users would be able to work more comfortably by placing these tools close to their preferred or *prime* hand position, which, incidentally, would be very much in line with Fitts's Law.

## *Final Thoughts*

The difficulty faced by interaction designers and user experience designers is that they have to consider, balance and combine measurable and non-measurable dimensions of user experience to create the best possible product. Fitts's Law tries to help user interface designers by giving them easily quantifiable, mathematically accurate values to base their design decisions on.

Of course, it is often possible to measure the quality of an interface using mathematical values: The fewer clicks required to access a certain set of data, the faster the navigation. The more vertical pixels a horizontally aligned interface preserves, the better it is suited for the respective device orientation. The closer the most frequently accessed buttons are placed, the more economic the cursor movement.

However, since interfaces are designed for humans, they also have to be consistent, considerate, inclusive, playful and discoverable: qualities that can hardly be measured as easily as clicks or pixels. The stunning accuracy and simplicity of mathematical formulas may sometimes lead designers to favor the measurable over not-so-easily-measurable dimensions. And while mathematical formulas can, indeed, help you enrich user experience, you should treat those formulas as *tools*, not as *principles*.

Instead, you should debate and choose anthropological principles first, and, if they permit it, use formulas such as Fitts's Law as much as possible to actually improve user experience. 🐼

## Further Reading

Resources you might be interested in:

- [Fitts' Law: Modeling Movement Time in HCI<sup>29</sup>](#)  
A brief overview of Fitts's Law.
- [The Information Capacity of the Human Motor System in Controlling the Amplitude of Movement<sup>30</sup> \(PDF\)](#)  
Paul Fitts's original article published in the Journal of Experimental Psychology in 1954.
- [Visualizing Fitts' Law<sup>31</sup>](#)  
A list of diagrams visualizing the basic rules of Fitts's Law.
- [Fitts's Law as a Research and Design Tool in Human-Computer Interaction<sup>32</sup> \(PDF\)](#)  
A case study comparing the efficiency of various input methods.
- [An Empirical Comparison of Pie vs. Linear Menus<sup>33</sup>](#)  
A controlled experiment testing the seek time and error rates for pie versus linear menus.
- [A Quiz Designed to Give You Fitts<sup>34</sup>](#)  
A small quiz to test your knowledge of Fitts's Law.

---

<sup>29</sup>. <http://www.cs.umd.edu/class/fall2002/cmsc838s/tichi/fitts.html>

<sup>30</sup>. <http://smashed.by/paul-fitts>

<sup>31</sup>. <http://www.particletree.com/features/visualizing-fittss-law/>

<sup>32</sup>. <http://www.smpp.northwestern.edu/savedLiterature/FittsLawPapers/FittsLaw%20as%20Research.Design.Tool.in.HCI.MacKenzie.pdf>

<sup>33</sup>. <http://www.donhopkins.com/drupal/node/100>

<sup>34</sup>. <http://www.asktog.com/columns/022DesignedToGiveFitts.html>

# Five Ways To Prevent Bad Microcopy

BY BILL BEARD 🍷

You've just created the best user experience ever. You had the idea. You sketched it out. You started to build it. Except you're already in trouble, because you've forgotten something: the copy. Specifically, the micro-copy<sup>35</sup>.

Microcopy is the text we don't talk about very often. It's the label on a form field, a tiny piece of instructional text, or the words on a button. It's the little text that can make or break your user experience.



If you think you've built the best user experience but didn't make sure the microcopy was spot on, then you haven't built the best user experience.

With the adoption of agile development and lean UX, we're often concerned about racing through iterations and getting our products in front of customers. But we can't forget that design is still about words<sup>36</sup>.

Everyone frets about marketing copy — and they should — but communication doesn't stop once you've sold the user. In some ways, you could argue that words become more important once the marketing experience is done. With most products, users have to be sold to only

---

<sup>35</sup>. <http://bokardo.com/archives/writing-microcopy/>

<sup>36</sup>. <http://37signals.com/svn/posts/3404-reminder-design-is-still-about-words>

once — or once in a while — and then they'll use the core product all the time.

If your microcopy isn't getting the job done, you'll lose users — and all the marketing in the world might not get you a second chance.

With that in mind, here are five ways to make sure your website's microcopy doesn't end up sinking your UX.

## *1. Get Out Of Your Own Head And Get To Know The User.*

I'm willing to bet that your experience is plastered with internal terminology, especially your labels and navigation. Every company has its own language, which often sneaks onto the website when we're not careful.

Don't let it happen. Never assume that what works for you will work for the user.

Here's a simple way to check whether your microcopy is too internal — or confusing, for that matter.

Let's assume that you're running some form of usability testing<sup>37</sup>. (If you're not, there's only about a thousand articles out there that will convince you you're making a mistake, so you don't need me for that.)

When you're testing, you probably get caught up in watching how the user interacts with your website and their facial expressions. But instead of simply watching, make a point to really listen to — and take notes on — the actual words the user says during testing. Listen closely to the phrases they utter when describing their actions. After all, you've told them to think out loud.

Listen to the inflection in their voice as they read microcopy: Did they say that label or term with a question in their voice? Don't hesitate to have your moderator follow up on copy. Have them go back and ask the user whether they've understood that label.

Take it a step further: Listen to what users say from the moment they walk in the building. Listen to their banter with the moderator, the jokes they make and the words they use to express their frustration or enjoyment.

You'd be surprised by what you can learn about a user and their language set from a comment they make about a cup of coffee. Everything someone says tells you something about them and can inform your copywriting process.

---

<sup>37</sup>. <http://uxdesign.smashingmagazine.com/2013/01/08/improving-your-website-usability-tests/>

## 2. The User Is A Person. Talk To Them Like One.

Because brevity is essential on the Web, most of us tend to truncate everything — particularly labels. Labels are great for design. They organize and keep tidy essential parts of a UI, such as navigation and forms.

Unfortunately, labels have an inherent problem: They're easily subject to a user's personal context because they don't provide explanation. They're on an island in the user's mind.

Not too long ago, we encountered this problem with a label at TheLadders. TheLadders is a job-matching service. Like any matching service, we required information to match a user with the right job.

### Job Goals



Target Job

Target Job Title \*

Creative Director

Recent Salary \*

Select

ZIP Code \*

10019

Desired Salary

Select

We thought this form was very clear. “Job Goals” is the label we’ve used for our matching criteria for almost 10 years. It’s brief, which helped to keep the navigation neat. But in a recent redesign, we noticed that users kept stumbling when first arriving on the page.

Turns out that people who don’t work in the job-search industry think of job goals as accomplishments they hope to achieve at their job, not as the details of their next job.

(We also fell into the trap covered in the first point: internal terminology = bad.)

So, we made it more conversational: “What job do you want?” Instantly, we could see that users no longer hesitated. Why? Because taking this new line of copy out of context was impossible.

Instead of forcing a label on a form or field for the sake of the UI, use natural language. The experience should be a conversation with the user, not a filing cabinet for them to drudge through.

Most of all, the labels in the navigation shouldn’t be more important than the user’s interaction with the pages that the labels represent.

## 3. Use Copy As A Guide, Not A Crutch.

*“We can fix that with copy.”*

I've heard this too many times when the UX falls short, and I hate it. If there's a problem with the design, then fix the design. The best experiences have minimal copy because they're intuitive. When designing the UX and you find yourself writing a sentence or two to help the user take an action, step back.

Tests have been conducted on readability and on the optimum length of content for understandability since the 1880s. With the rise of the Internet, this story became about line length<sup>38</sup>. Most sources net out between 45 to 75 characters as the ideal line length.

To me, line length is moot, especially with responsive and mobile design. Besides, character counts seem tedious and not very lean.

Instead, I subscribe to the original readability tables of Rudolf Flesch (pictured below), in which sentences with eight words or fewer are regarded as “very easy” to read.

Style	Flesch Reading Ease Score	Average Sentence Length in Words	Average No. of Syll. Per 100 Words	Type of Magazine	Estimated School Grade Completed	Estimated Percent of U.S. Adults
Very Easy	90 to 100	8 or less	123 or less	Comics	4th grade	93
Easy	80 to 90	11	131	Pulp fiction	5th grade	91
Fairly Easy	70 to 80	14	139	Slick fiction	6th grade	88
Standard	60 to 70	17	147	Digests	7th or 8th grades	83
Fairly Difficult	50 to 60	21	155	Quality	Some high school	54
Difficult	30 to 50	25	167	Academic	High school or some college	33
Very Difficult	0 to 30	29 or more	192 or more	Scientific	College	4.5

*The readability tables of Rudolf Flesch.*

It may be an old standard, but it still may be the best measuring stick we have, and it's the easiest for lean teams to follow. On the Web, we're shooting for “very easy to read” every time, and we want to be able to communicate with as many people (93%) as possible.

If you can't explain what a user needs to do in eight words or fewer, then reconsider the design.

Once the user has gotten past the marketing portion of the experience, use copy as a guide to usher them through the product. The best copy on basic UI features, such as a form, will get read but not really noticed. The user absorbs the words and takes the desired action without a hiccup.

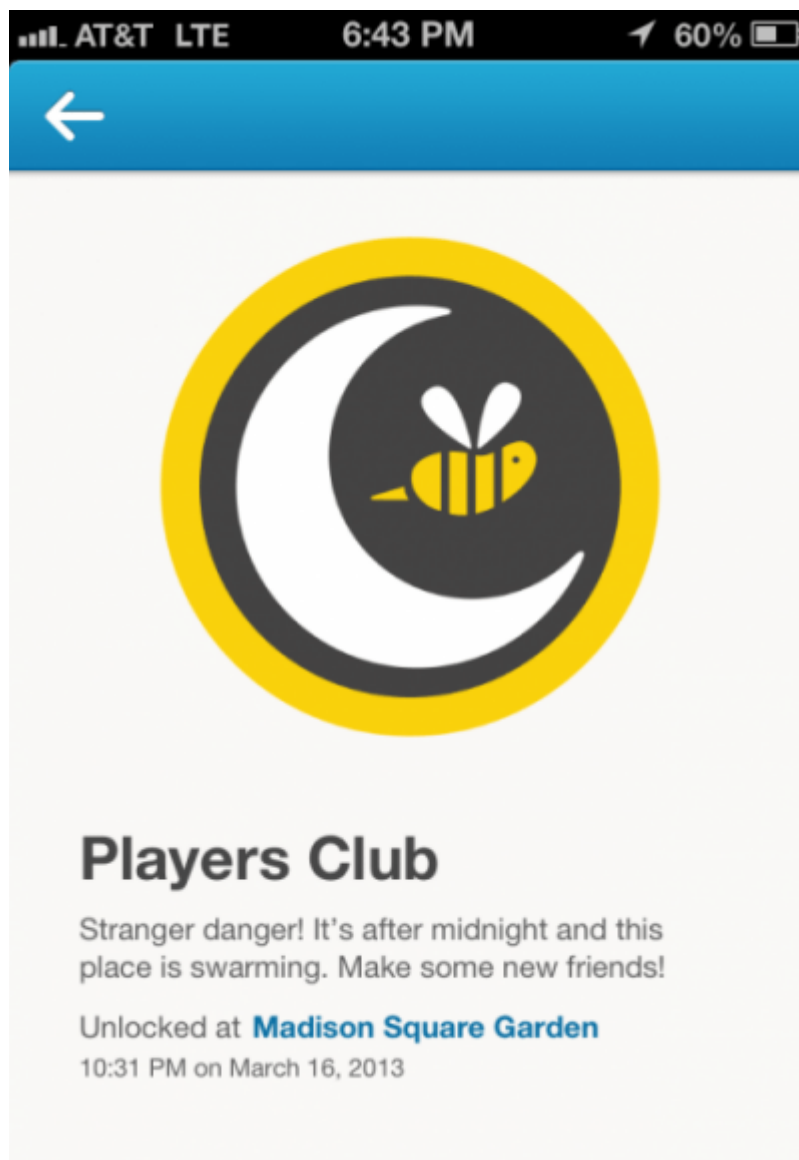
---

<sup>38</sup>. <http://baymard.com/blog/line-length-readability>

#### 4. Treat Every Moment Like A Branding Moment, Even When It's Not.

There are multiple definitions of a “branding moment.” When we talk about copy in a UX, I define it as a moment when you purposefully inject your brand’s tone and voice into what would normally be a straightforward user interaction.

For example, Foursquare has a lot of great branding moments with its badging system. I unlocked the one below not too long ago. It’s fun and a bit edgy, on point with Foursquare’s brand.



*A good job of a branding moment with Foursquare's brand.*

But getting carried away is easy. Think hard before using fun or quirky — or whatever your brand’s voice is — copy in a situation that the user wants and expects to be straightforward.

Your brand's tone and voice are essential to consider when writing all of your copy, but it should not get in the way of a user who is trying to take action.

**Avoid over-branding copy on:**

- navigation,
- forms and field labels,
- instructional text,
- selection text (drop-downs, radio buttons),
- buttons.

**Consider incorporating your brand's voice in:**

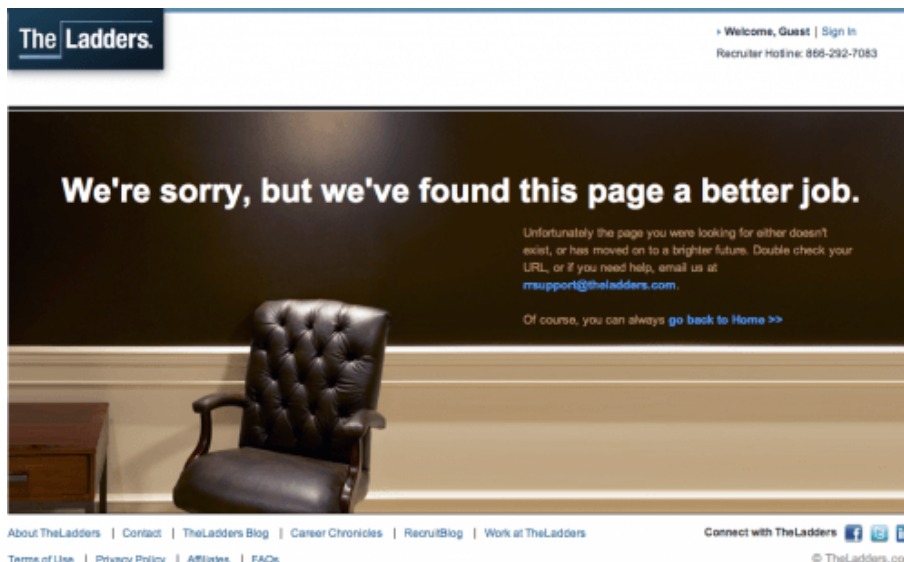
- confirmation messaging,
- rewards (badges, points),
- 404 pages,
- server errors,
- error messaging.

The difference between these lists is simple. In the first list, the user is attempting to take action; the second list is the results of actions.

In the first list, you don't want to risk confusing users as they try to accomplish something and cause them to abandon. Clarity is essential.

In the second list, you have an opportunity to embrace the user's success (Foursquare's "You're on fire!") or mitigate a failure (TheLadders 404 page, below) by injecting your brand. You don't need anything from the user at these points.





*TheLadders 404 page.*

This isn't to say that you can't brand that first list. But if you're going to do it, test it first. With branding moments, execution is paramount. If you're unsure, don't risk it.

By choosing *not* to brand parts of the experience to keep it simple and easy for the user, you'll provide an enjoyable experience, which will make your brand stronger. So, every moment is a branding moment. Even when it's not.

## *5. If Content Is King, Then Treat Context Like A Queen.*

The hot saying right now is "Content is king." Native advertising, or the integration of relevant content into a natural experience for the purpose of acquisition, is becoming a core offering of many agencies and has spawned a few popular startups.

But without context, content is useless. (And if you're big on Game of Thrones, then you'll know that queens have all the real power!)

Whether you're labeling a form or writing a blog post, you have to either understand the user's existing context or provide context for them.

A user's context will define how they interpret the copy on the page. That context could come from anywhere: an email they've just read, or something that happened to them when they were eight.

When a user doesn't have proper context, they get confused. When a user gets confused, they abandon.

If you're agile and iterative, accounting for a holistic experience adds an additional layer of complexity, in the form of consistency. A

simple change to copy on one page could affect 10 other pages. One minute you're calling something "Job Goals," and the next you're changing it to "What job do you want?" Well, where else have you used "Job Goals"?

To better understand the user's context — and to check for consistency — sit down at least once per iteration and experience your "contextual flow" as the user sees it.

For example, if you have a subscription service, the flow might be something like this:



That's at least 10 distinct steps in which a user's context could be created, confirmed or altered.

Sit down, take a breath and wipe your mind of what you know is there. Then start with Google or your home page or wherever the first touch usually happens.

Does your onboarding experience deliver the same promise as your Google ad? Have you described a feature using the same language throughout? Are your labels so subjective that the context gets lost? These are questions to answer as you go through the flow.

## *Whatever Happens, Don't Ignore Your Microcopy.*

Microcopy often falls victim to personal bias, internal terminology, poor branding, broken contextual flows, time crunches and other factors. Any of these can undermine even the most well-designed UX and the copy within.

Here's the thing about mistakes with microcopy: They're so easy to make yet so hard to identify after you've made them.

You have a much better chance of stopping the mistakes in advance than of identifying them after the fact. When you're testing, how often do you think, "Hey, maybe we should change the label on the third field of this form?" You're wrapped up in other UX mistakes that you know

you've made. Unfortunately, a repeated pattern of noticeable failure is usually needed in order for microcopy to get updated or even tested.

So, the next time you're creating or improving an experience, I hope you employ some of the tactics provided here so that you avoid these "easy" mistakes and do right by your microcopy — and by your user. 🍷

# Infinite Scrolling: Let's Get To The Bottom Of This

BY YOGEV AHUVIA 🐾

Infinite scrolling promises a better experience for users. However, the good is often accompanied by the bad and the ugly. Once we understand the strengths and weaknesses of infinite scrolling, we can begin to use it to enhance our interfaces.

Human nature demands hierarchy and structures that are easy to navigate. But infinite scrolling sometimes leaves users feeling disoriented as they travel down a page that never ends.



## *The Good*

Long lists are not new, but the way in which we scroll these lists has fundamentally changed since the arrival of mobile interfaces. Due to the narrowness of mobile screens, list items are arranged vertically, requiring frequent scrolling.

Infinite scrolling is highly trending<sup>39</sup> as an interaction behavior on pages and lists. The basic functionality is that, as the user scrolls through content, more content is loaded automatically. With the popularity of social media, massive amounts of data are being consumed; in-

---

<sup>39</sup>. <http://www.google.com/trends/explore#q=Infinite%20Scroll>

finite scrolling offers an efficient way to browse that ocean of information, without having to wait for pages to preload. Rather, the user enjoys a truly responsive experience, whatever device they're using.



*Pagination versus infinite scrolling.*

Websites with lots of user-generated content today are using infinite scrolling to handle content that is being generated every second. By unspoken agreement, users are aware that they won't get to see *everything* on these websites, because the content is updated too frequently. With infinite scrolling, social websites are doing their best to expose as much information as possible to the user.

Twitter<sup>40</sup> integrates infinite scrolling effectively. Its feed fits the criteria: a large amount of data (tweets) and a real-time platform. From the perspective of the user, all tweets are equally relevant, meaning that they have the same potential to be interesting or uninteresting; so, users will often scroll through all of the tweets in their feed. Being a real-time platform, Twitter is constantly being updated, even if the user leaves their feed unattended. Infinite scrolling seems to have been created especially for websites like Twitter, which successfully employs the technology.

Infinite scrolling appears to have found its niche on the Web. However, there are also drawbacks that must be considered before assessing its value.

## *The Bad And The Ugly*

With so much data to browse, users must stay focused on the information they are searching for. (Remember about being goal-oriented?) Do users always want a never-ending stream of data? Analytics show that when users search for information on Google, only 6%<sup>41</sup> advance to the

---

<sup>40</sup>. <http://www.twitter.com/estellewv>

second page. So, 94% of users are satisfied with receiving only 10 results, which suggests that users find Google's ranking of results to be relevant.

### TO CLICK OR NOT TO CLICK

Google has implemented infinite scrolling for image search results but has yet to implement<sup>42</sup> it for its general results. Doing so would eliminate the need for users to click to reach the second page. Google will probably maintain pagination because this pattern is quite symbolic for its brand. If it does switch to infinite scrolling, when would users typically stop scrolling? After 20 results? 50? When does an easy browsing experience become more complicated?

Looking for the best search result could take a second or an hour, depending on your research. But when you decide to stop searching in Google's current format, you know the exact number of search results. You can make an informed decision about where to stop or how many results to peruse because you know where the end is. According to studies<sup>43</sup> conducted in the field of human-computer interaction, reaching an end point provides a sense of control; you know that you have received all relevant results, and you know whether the one you are looking for is there or not. Knowing the number of results available provides a sense of control and helps the user make a more informed decision, rather than be left to scour an infinitely scrolling list.



*Pagination is a barrier of clicks.*

When items are distributed across Web pages, they are framed and indexed and have a start and end point. The information is presented clearly and orderly. If we select an item from a paginated list and are

41. <http://www.quora.com/What-is-the-distribution-of-traffic-between-Google-organic-search-results-e-g-1-vs-2-in-rankings-first-page-vs-second-page>

42. <http://thenextweb.com/google/2011/08/17/google-testing-infinite-scrolling-in-search-results/>

43. [http://videlectures.net/chio8\\_kieras\\_phc/](http://videlectures.net/chio8_kieras_phc/)

taken from that page, we know that clicking “Back” will return us to that page (probably to the same scroll position). Our Web search will continue right where it left off.

If you scroll the same list of results with infinite scrolling, you are left without that sense of control because you are scrolling through a list that is conceptually infinite. Let’s say you count yourself among the 94% who stop reading after the first page (i.e. 10 results) of a Google search. When the list scrolls infinitely, there is essentially no end to the first page. Rather than look for the end of the page, which doesn’t exist anyway, you decide to stop scrolling at the 10th item. This poses a problem with infinite scrolling, because the 11th item is directly in sight. With a paginated list, on which you wouldn’t see the 11th result, deciding not to continue browsing is easier. However, when the next results are already there<sup>44</sup>, you’d probably just keep on scrolling and scrolling.

As Dmitry Fadeyev points out:

*“People will want to go back to the list of search results to check out the items they’ve just seen, comparing them to what else they’ve discovered somewhere else down the list. Having a paginated interface lets the user keep a mental location of the item. They may not necessarily know the exact page number, but they will remember roughly what it was, and the paginated links will let them get there easier.*

*Not only does the infinite scroll break this dynamic, it also makes it difficult to move up and down the list, especially when you return to the page at another time and find yourself back at the top, being forced to scroll down the list once again and wait for the results to load. In this way the infinite scroll interface is actually slower than the paginated one.”*

— Dmitry Fadeyev, When Infinite Scroll Doesn’t Work<sup>45</sup>

## When Infinite Scrolling Fails

The best companies are constantly testing<sup>46</sup> and studying<sup>47</sup> new interactions with their users. Increasing numbers of these studies are showing that infinite scrolling does not resonate with users if it does not support their goal on the website.

---

<sup>44</sup>. [https://www.youtube.com/watch?v=j9UfY\\_94sKU](https://www.youtube.com/watch?v=j9UfY_94sKU)

<sup>45</sup>. <http://www.usabilitypost.com/2013/01/07/when-infinite-scroll-doesnt-work/>

<sup>46</sup>. <http://www.businessinsider.com/marissa-mayer-google-9-2010-11>

<sup>47</sup>. [http://readwrite.com/2010/08/26/revolt\\_angry\\_digg\\_users\\_want\\_their\\_baby\\_back](http://readwrite.com/2010/08/26/revolt_angry_digg_users_want_their_baby_back)

## TEMPTATION

When you're looking for that perfect search result and are tempted to continue scrolling into a wasteland of irrelevant results, time is wasted. Chances are that the best result will appear in the first 10 items. Therefore, infinite scrolling merely tempts you<sup>48</sup> to continue reading, wasting time and decreasing productivity in the process.

## OPTIMISM

Even more annoying is that scroll bars do not reflect the actual amount of data available. You'll scroll down happily assuming you are close to the bottom, which by itself tempts you to scroll that little bit more, only to find that the results have just doubled by the time you get there.

## EXHAUSTION

Infinite scrolling overwhelms users with stimuli. Like playing a game that you can never win, no matter how far you scroll, you feel like you'll never get to the end. The combination of temptation and optimism play a big role in exhausting the user.

## POGOSTICKING

Infinite scrolling often causes your position on the page to get lost. "Pogosticking" happens when you click away from an infinitely scrolling list and, when you return by clicking "Back," are brought to the top of the previous page, instead of to the point where you left off. This happens because the scroll position is lost when you navigate away from an infinitely scrolling page, forcing you to scroll back down each time.

## LOSS OF CONTROL

Infinite scrolling leaves you with the feeling that you might be missing out on information. You continue scrolling because the results are right there, but you feel overwhelmed because you're losing control over the amount of data being shown. There is something nice about defined pages on which the amount of content is quantified, where you can comfortably choose whether to click to view more or to stop. With infinite scrolling, you don't have control over the amount of data on the page, which becomes overwhelming.

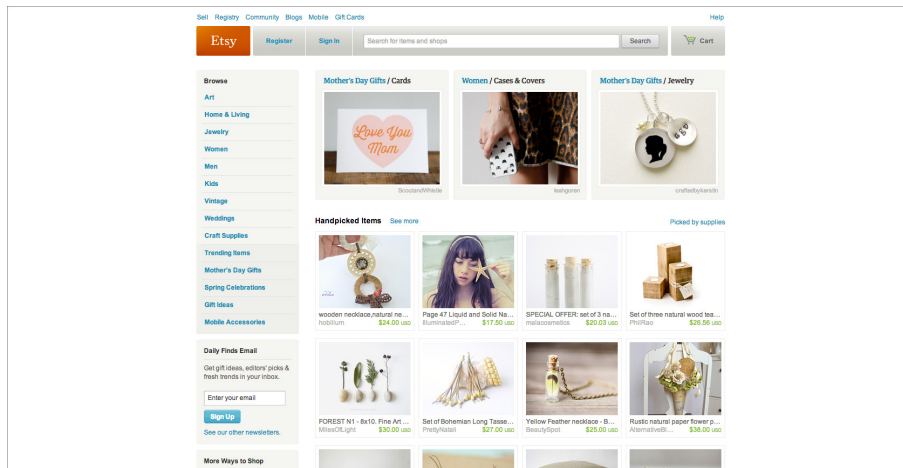
---

<sup>48</sup>. <http://techcrunch.com/2012/08/18/infinite-scroll-the-webs-slot-machine/>



## DISTRACTING

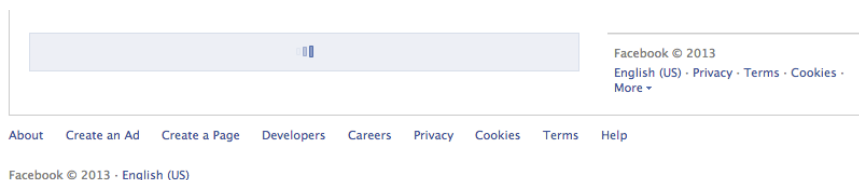
Etsy<sup>49</sup>, an e-commerce marketplace, implemented infinite scrolling, only to find that it led to fewer clicks<sup>50</sup> from its users. Infinite scrolling was unsuccessful because users felt lost in the data and had difficulty sorting between relevant and irrelevant information. While infinite scrolling provided faster and more results, users were less willing<sup>51</sup> to click on them, defeating its very purpose.



Etsy's home page (Large version<sup>52</sup>)

## UNREACHABLE

Have you tried reaching the footer of Facebook lately? The footer block exists below the news feed, but because the feed scrolls infinitely, more data gets loaded as soon as you reach the bottom, pushing the footer out of view every time. Footers exist for a reason: they contain content that the user sometimes needs. In Facebook's case, the user can't reach it. The links are repeated elsewhere but are harder to find. Infinite scrolling impedes the user by making important information inaccessible.



Facebook's auto-loading news feed makes the footer unreachable.

49. <http://www.etsy.com/>

50. <http://www.slideshare.net/danmckinley/design-for-continuous-experimentation>

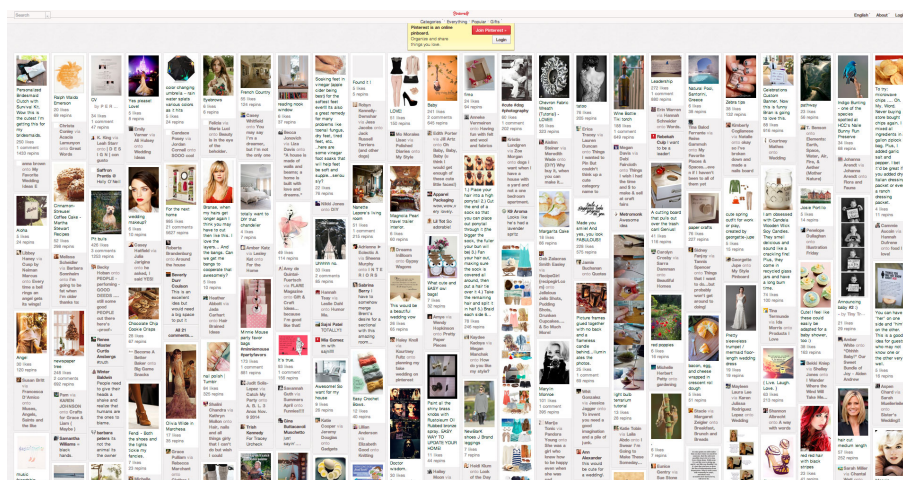
51. <http://danwin.com/2013/01/infinite-scroll-fail-etsy/>

52. <images/etsy-home-page.jpg>

Footers serve as a last resort. If users can't find something or they have questions or want more information or explanation, they often go there. If they don't find it there, they might leave the website altogether. Companies that implement infinite scrolling should either make the footer accessible by making it sticky<sup>53</sup> or relocate the links to a sidebar.

## NOT EXCLUSIVE

Pinterest<sup>54</sup> does not have a footer at all, which makes sense given the problem we just saw with Facebook. Through infinite scrolling, Pinterest emphasizes its profusion of data, an endless sea of inspiration taken from all over the Web.



*Pinterest's ocean of pins (Large version<sup>55</sup>)*

Images are faster and easier to scroll than text, so Pinterest and Google Images succeed with infinite scrolling to an extent. However, billions of images<sup>56</sup> are on the Web, and users would prefer to see only the best of them. There is something to be said for exclusivity, which seems to be lacking in Pinterest's layout.

Limiting the number of images on Pinterest's home page, with an "Editor's picks" or "Most popular" list, might make the website more appealing. How exclusive can a pin be when a ton of other similar pins are next to it?

Pinterest's tactic of using infinite scrolling for its plethora of data suffers because it overwhelms users<sup>57</sup>. The collection looks bottomless,

53. <http://struck.com/>

54. <http://www.pinterest.com>

55. [images/PinterestOcean.jpg](http://images/PinterestOcean.jpg)

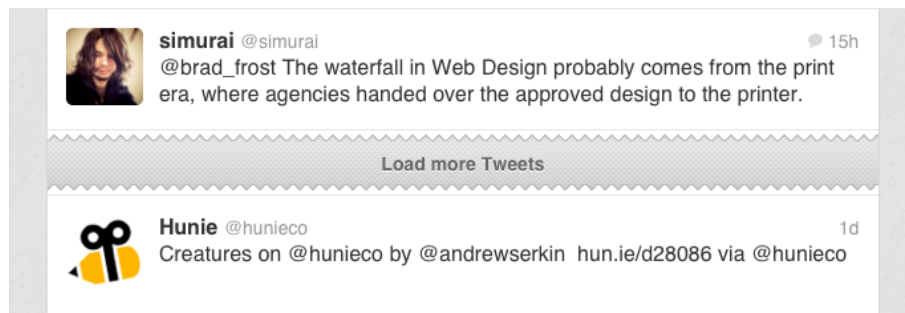
56. <http://techcrunch.com/2013/01/17/facebook-photos-record/>

57. <http://nicolelumberras.com/2013/04/17/pinterest-overwhelms-me/>

but its immensity is somewhat daunting, and browsing it might seem a waste of time. Ultimately, Pinterest is trying to expose users to infinite inspiration<sup>58</sup>, but that actually undermines the human need for control<sup>59</sup>. The amount of data becomes intimidating, and users are left with mixed feelings.

## When Usability Wins

As mentioned earlier, Twitter integrates infinite scrolling effectively. The user sees an infinitely growing list of tweets and can comfortably click on a tweet to expand it in place, preventing the page from refreshing and, as a result, maintaining their scroll position.



*Twitter's torn feed.*

On its mobile version, Twitter even adds a “torn paper” marker, indicating to the user where to resume reading. This subtle and simple solution enables the user to scroll up and down the list, while having a recognizable point to return to. Psychologically, that marker reassures the reader by dividing read and unread content. Such markers give the user a sense of control and a better perception of the content’s depth and how far they’ve gotten into it.

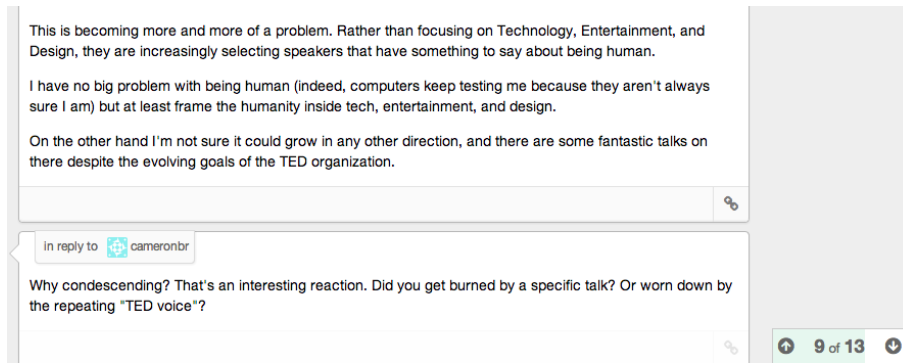
Twitter is not the only one. Discourse<sup>60</sup>, an emerging discussion platform, also has infinite scrolling that empowers the user. The company considered the importance of infinite scrolling to its user experience and implemented an intriguing and unique progress indicator. The indicator appears when needed and remains in view (without interfering) while the user reads the content. The indicator numbers the item currently being viewed out of the total number of items. This is a great way to make the user feel in control, even with a lot of data.

---

<sup>58</sup>. <http://stephencorwin.com/blog/?p=97>

<sup>59</sup>. <https://www.youtube.com/watch?v=6ELAkV2fC-I>

<sup>60</sup>. <http://www.discourse.org/>

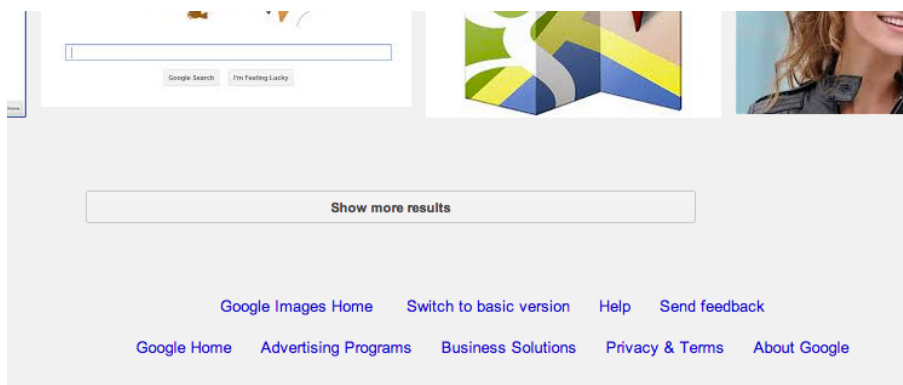


*The smart progress indicator on Discourse.*

## GO HYBRID

A hybrid of infinite scrolling and pagination is also a good option in many cases. With this solution, you would show a “load more” button at the end of a preloaded list, which, when clicked, loads another batch of items onto the list. The same behavior that infinite scroll does automatically, this button does on demand. The interface gains some of the advantages of infinite scrolling, without some of its drawbacks.

Because infinite scrolling requires the website to fetch so much content, the hybrid solution is used at times to control the data load. In Facebook’s news feed and Google’s image search, the infinite scrolling is automatic at first but becomes on-demand once a certain number of items have loaded. This maintains the interface while limiting the load on the server.



*Hybrid infinite pagination on Google Images (Large version<sup>61</sup>)*

<sup>61</sup>. [images/hybrid-infinite-pagination.png](#)

## *Infinite Pages*

Infinite pages<sup>62</sup> take the concept of infinite scrolling to a new level. Websites that employ this concept are “one-pagers.” To remove the barrier of clicking to the next page, the designer turns the entire website into one long scrollable page. Examples are Unfold<sup>63</sup> and Lost World’s Fairs<sup>64</sup>.

On these one-page websites, the sections are spread vertically, one after another. This makes the whole website less comprehensible – hence, less accessible. These websites might not have infinite scrolling, but the user might still have that feeling of a never-ending page.

On infinite pages, the height of each section will vary according to its contents. Although the approach can make for some creative interactions, it can leave users disoriented and unsure where to scroll for the next piece of information. The scroll bar is hidden on many such pages, leaving users feeling lost as they unconsciously look for the scroll position<sup>65</sup> to track their progress. Hidden scroll bars deprive users of that chance for rescue. Users shouldn’t be left helpless; the interface should clearly show them how to navigate the page.

---

<sup>62</sup>. <http://www.usabilitypost.com/2012/10/29/the-return-of-the-scroll/>

<sup>63</sup>. <http://unfold.no/>

<sup>64</sup>. <http://lostworldsfairs.com/atlantis/>

<sup>65</sup>. [http://www.cxpartners.co.uk/cxblog/the\\_myth\\_of\\_the\\_page\\_fold\\_evidence\\_from\\_user\\_testing/](http://www.cxpartners.co.uk/cxblog/the_myth_of_the_page_fold_evidence_from_user_testing/)



*Not knowing where they stand can leave the users disoriented.*

UX engineers need to take extra care when designing infinite pages. They must take into account accessibility; tell users where they stand by showing the length of the page and how much they've viewed. Solutions could include a fixed menu<sup>66</sup>, a map of the page<sup>67</sup> or a scroll progress bar<sup>68</sup>.

Another trick is the parallax effect<sup>69</sup>, whereby different layers on the page move at different speeds according to the user's scrolling, creating the illusion of depth (as seen on Andrew McCarthy's website<sup>70</sup>). While it can help to create beautiful<sup>71</sup> and innovative<sup>72</sup> experiences, it is some-

---

<sup>66</sup>. <http://www.netcraft.co.il/devshots/>

<sup>67</sup>. <http://larsjung.de/fracs/>

<sup>68</sup>. <http://www.putzengel.at/>

<sup>69</sup>. [http://en.wikipedia.org/wiki/Parallax\\_scrolling](http://en.wikipedia.org/wiki/Parallax_scrolling)

<sup>70</sup>. <http://andrevv.com/>

times heavily overused, and users can get confused by how much they need to scroll for more content. When the parallax effect is combined with animation, the user can get confused about whether the page is being scrolled by their movement or is moving autonomously. It's wise to use the technique to enhance content, not as the content itself.

## *Let's Get To The Bottom Of This*

Infinite scrolling can be an innovative feature that greatly improves an interface by exposing content and making performance more efficient. But it needs to be used correctly.

Avoid the following sinkholes to achieve a strong infinite scrolling experience:

- **Users want immediate access to exclusive data.**  
Users don't want to be left to explore all of a website's data on their own. When implementing infinite scrolling, identify what data is exclusive to your website and elevate it to the top of the page, and filter less relevant information.
- **Users want to feel in control.**  
Infinite scrolling sabotages that feeling of control. Add a smart progress indicator, a fixed menu or a map.
- **Users often look for landmarks when scrolling.**  
When scrolling through long lists, users expect to be able to easily distinguish between new and viewed data. Add landmarks along the interface to keep users oriented.
- **Consider conventional pagination or a hybrid solution.**  
Good old pagination is always an alternative to infinite scrolling. And if that doesn't fit the context, then a hybrid solution, using a "load more" button, could greatly enhance the interface.
- **Provide interesting content without an ambiguous interface.**  
Having to traverse a never-ending list is logical only if the user leaves feeling that it was worthwhile.
- **Users often expect a footer.**  
If footer-type information is functional to the interface, then it should appear at the bottom of the page. A fixed footer is usually the way to go with infinite scrolling.

---

71. <http://www.angelamorelli.com/water/>

72. <http://www.shibui.me/web/scroll/index.html>

- **An infinite list is still a list.**

Infinite scrolling still needs to meet common interface standards.

Whether users take their eyes off the screen for a moment or click a link and then click “Back,” they expect to return to the exact point where they left off. Whatever your interface, make sure it meets users’ expectations.

- **Effects are nice to have but not a must.**

Many infinitely scrolling interfaces have various effects to show more data (whether by sliding in new content or another method). Be mindful not to focus more on effects than on presenting data in the most effective way possible.

## USE IT CORRECTLY

Users are goal-oriented and find satisfaction in reaching the end of their exploration. To be effective, infinite scrolling has to account for this. Nothing is really infinite, not even these infinitely scrolling lists we’ve looked at. Users should always know where they stand, even when the content has not finished loading. They should know what the total amount of data is and be able to easily navigate the list. Infinite scrolling has to be implemented in the best possible way so that users can always find their way. 🐼



# Designing Great Feedback Loops

BY LOREN BAXTER 🍷

The products we build are full of feedback loops, whether we know it or not. People who study human behavior agree that feedback loops play a critical role in what we do. From biofeedback to the quantified self, designers and psychologists alike are discovering the real power that these cyclical interactions play in shaping our day-to-day choices. Designing for behavior change can increase user engagement, create business value and improve lives.

Whatever you're designing, it probably involves feedback. Designing that feedback to be as effective as possible can mean the difference between a successful and failed product. This chapter discusses how to influence behavior by designing well-crafted feedback loops.

## *What They Are, And Where They Live*

Feedback loops consist of a few simple steps:

1. A person takes an action,
2. The action has one or more effects,
3. The important effects of this action are presented back to the person,
4. This loop is repeated regularly.



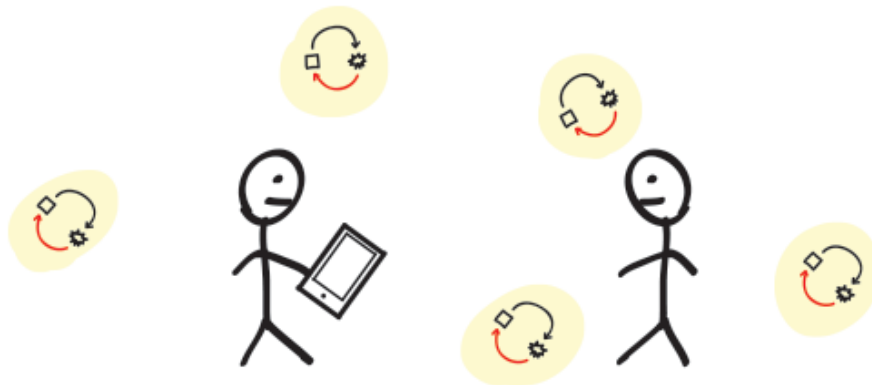
There are many natural feedback loops. They reveal to people whether they are making good or bad choices, and people learn from them. If someone eats something bad, they get sick. If they eat something good, it is delicious. If a person is nice to someone, and that someone responds in kind, the person feels good. If someone damages their body, it hurts. The brain has systems of reward and punishment that support basic survival.

There are now many man-made examples of these as well. People take tests on their knowledge and receive a grade. They post a status update to Facebook and get rewarded when people like or comment on it. “Post more stuff like this,” the feedback seems to say.

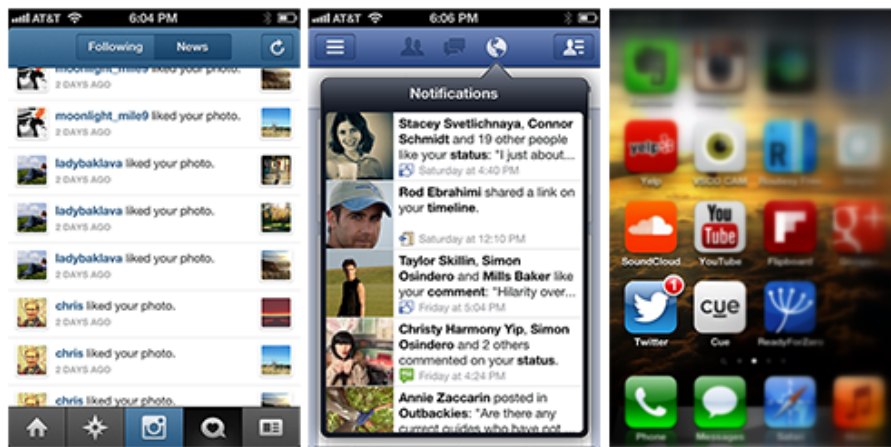
So, let’s dive into improving our own feedback loops!

## *Discover Your Product’s Basic Feedback Loops*

The first step is to identify your product’s feedback loops. What are the fundamental repeated behaviors that your users engage in?



Many social apps share the following basic loop: a user posts something, people see it, people click a button to “like” it, and the likes are presented back to the user. Email has a strong, simple loop: log in, see new emails, deal with the emails.



*The social apps Instagram, Facebook and Twitter all have feedback loops.*

But they can also be harder to see. [Yelp](http://www.yelp.com)<sup>73</sup> has built great feedback loops around local reviews. A restaurant provides food and services, and better services lead to better reviews. Meanwhile, customers dine in and leave reviews, which themselves are reviewed in turn by other people (see those “Useful,” “Funny” and “Cool” buttons). Some people who leave enough reviews can get invited to special parties and more (“Yelp Elites”). These loops are purposefully designed: they drive people to engage with the website and to contribute content that is valuable to Yelp’s business.



*My Yelp profile, with prominent feedback.*

Look for patterns of repeated behavior, and list the ones you find. These behaviors could span time, digital and physical space, channels of communication, and even different people. Look closely at user actions that trigger a notification sent to them or others. Investigate places where users communicate with each other or with employees, either directly or indirectly. These areas are fertile ground for feedback loops, and you should be able to draw up a robust list without much effort.

<sup>73</sup>. <http://www.yelp.com>

## Understand How These Loops Are Broken, And Fix Them

Now that you have a list of areas that can be improved, identify the most important or common ones in your product. They can be improved in any one of these aspects:

1. Speed,
2. Measurability,
3. Context,
4. Connection to motivations.

### 1. SPEED

*“It’s too slow!”*

**Problem:** The longer it takes for feedback to arrive, the less it will influence future decisions.

In operant conditioning, this is called “immediacy<sup>74</sup>,” and it’s also related to “hyperbolic discounting<sup>75</sup>.”

For example, when the user of a financial website makes an investment, seeing the outcomes of that investment could take years. Without any simulation of what their investment’s growth *might* look like, they have little emotional material to support that decision.



**Solution:** Tighten the feedback loop by presenting effects as soon as possible. This could entail revealing real effects, or simply simulating them. As long as they’re grounded in reality, they will have influence.

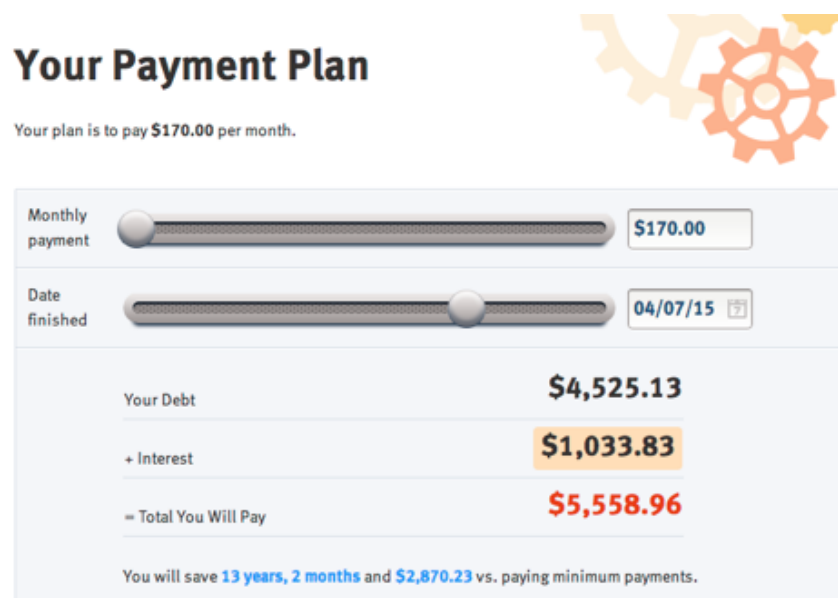
---

<sup>74</sup>. [http://en.wikipedia.org/wiki/Operant\\_conditioning#Factors\\_that\\_alter\\_the\\_effectiveness\\_of\\_consequences](http://en.wikipedia.org/wiki/Operant_conditioning#Factors_that_alter_the_effectiveness_of_consequences)

<sup>75</sup>. [http://en.wikipedia.org/wiki/Hyperbolic\\_discounting](http://en.wikipedia.org/wiki/Hyperbolic_discounting)

Even presenting simulated feedback right *before* someone makes a decision is effective. Knowing that an investment will likely earn them \$1,000 will help the user feel good about making that decision.

On ReadyForZero<sup>76</sup>, our tool to help people organize and pay off debt, we enable the user to simulate the impact of different payment plans in real time, instantly revealing an outcome that could happen years from now.



*ReadyForZero simulates the impact of a payment plan in real time.*

Smartphone push notifications have also gone a long way to giving immediate feedback. Posting something to Instagram, Facebook, Twitter and so on and then hearing one's phone buzz with social feedback within minutes is a powerful loop. Always be sure to use push notifications when appropriate.

Credit cards provide a physical example of how delayed feedback can make for a dark pattern that drives harmful behavior. These cards have made banks huge amounts of money by alleviating a painful loop: pulling out your wallet and handing over your hard-earned cash. The more cash you spend, the more it disappears from your wallet. Soon, you are forced to go to the bank to get more, making you physically see and feel the effects of spending. Credit cards effectively delay the feedback on how much you spend until you bother to check the balance or are sent a bill at the end of the month. Even then, you have the option of delaying payment of the full balance (accruing interest in the meantime). This makes sliding into debt very easy.

---

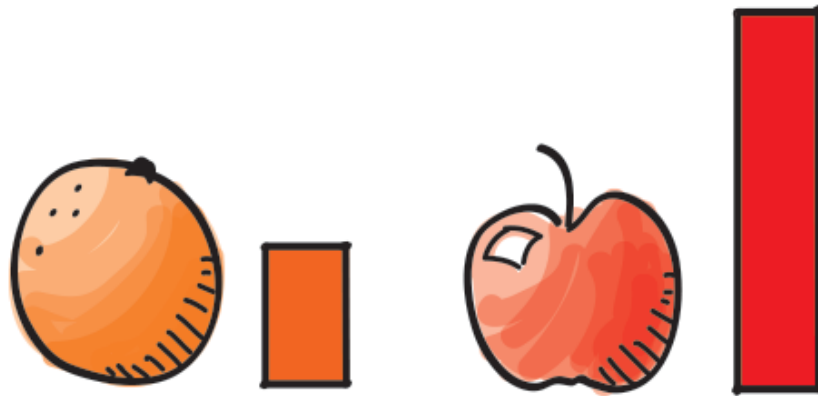
<sup>76</sup>. <http://www.readyforzero.com>

## 2. MEASURABILITY

*“How does it compare?”*

**Problem:** Comparing vague, unquantifiable feedback to other factors in order to make an informed decision is too difficult.

Take nutrition, a notoriously vague problem space for feedback. How do people know whether they’re eating well? If they’ve heard that the thing they’re eating is unhealthy, *how* unhealthy is it? If the feedback is presented in some quantifiable way that relates to reality, it will be easier to understand and the person’s decisions will be more informed. Nutrition labels have made the effects of food more measurable. Now, someone can actually quantify the benefit of eating an apple, and the problem with eating a pile of bacon.



**Solution:** As nutrition labels have done for food, give quantifiable feedback to users. Use visual comparisons as well, when possible. The human brain is great with tangible comparisons between the sizes, locations and other variables of objects. Read up on the [framing effect](http://en.wikipedia.org/wiki/Framing_effect_(psychology))<sup>77</sup> as well as [anchoring](http://en.wikipedia.org/wiki/Anchoring)<sup>78</sup> for more on the psychological reasons for why this works.

[Klout](http://www.klout.com)<sup>79</sup> is attempting to make social influence measurable with its Klout score. It demonstrates the power of making things measurable, enabling users to objectively compare the online social influence of different people. It also highlights the risks of creating metrics where there were none before. The Klout score is interesting, but it’s based on a limited set of social signals, such as followers, likes, replies, retweets

<sup>77</sup>. [http://en.wikipedia.org/wiki/Framing\\_effect\\_\(psychology\)](http://en.wikipedia.org/wiki/Framing_effect_(psychology))

<sup>78</sup>. <http://en.wikipedia.org/wiki/Anchoring>

<sup>79</sup>. <http://www.klout.com>

and comments. Metrics must be believable and must accurately reflect reality, a hard thing to accomplish with social influence.



*My Klout score changes over time based on social interactions.*

A lesson we've learned about measurability at ReadyForZero is to use numbers that are at a scale that people deal with in real life, typically up to 100. Telling a user that their debt will cost them \$150,000 in interest means almost nothing to them other than "a lot of money" and isn't that useful. Instead, we focus on how much someone's debt costs them in interest *every day*, because this number tends to be in the \$1 to \$30 range and is comparable to many things they purchase daily, such as coffee and meals. Feeling good about their progress has helped our users significantly.

### 3. CONTEXT

*"What does it mean?"*

**Problem:** Feedback presented without the context of a larger goal isn't as effective.

Saving some extra money can make someone feel good. But saving towards a goal is more meaningful. "This deposit of \$500 has put me one third of the way towards my vacation." Providing context for the feedback not only creates a signpost by which to measure progress, but actually frames the way people think about a problem. For example, "You've spent \$1,200 this week" is a very different way to frame one's finances than "You've spent \$1,200 of your budgeted \$700 this week," even though they represent the same data.



**Solution:** Frame effects in the context of user goals. How much closer to or further from a goal has this action taken the user? On ReadyForZero, we show the amount of time and interest saved by making a big payment towards a debt, and the amount lost by buying things on a credit card.

This is a form of “anchoring<sup>80</sup>.” By showing a desired goal, you’ve set the anchor by which the user can then evaluate their current situation. If they have not yet reached the goal, they’ll feel the tension of wanting to reach it.

The Nike+ Fuelband<sup>81</sup>, along with many other quantified-self activity-tracking devices, explicitly creates feedback loops and goals around user behavior where there were none before. The user’s physical activity is tracked and reported throughout the day, compared to goals, and shared online. Note the strong use of progress bars, both on the watch and in the app, which communicate how close the user is to reaching their daily goal.

---

<sup>80</sup>. <http://en.wikipedia.org/wiki/Anchoring#Anchoring-and-adjusting>

<sup>81</sup>. <http://nikeplus.nike.com/plus/products/fuelband>





*The Nike+ Fuelband shows daily activity relative to goals.*

#### 4. MOTIVATIONS

*"I don't care."*

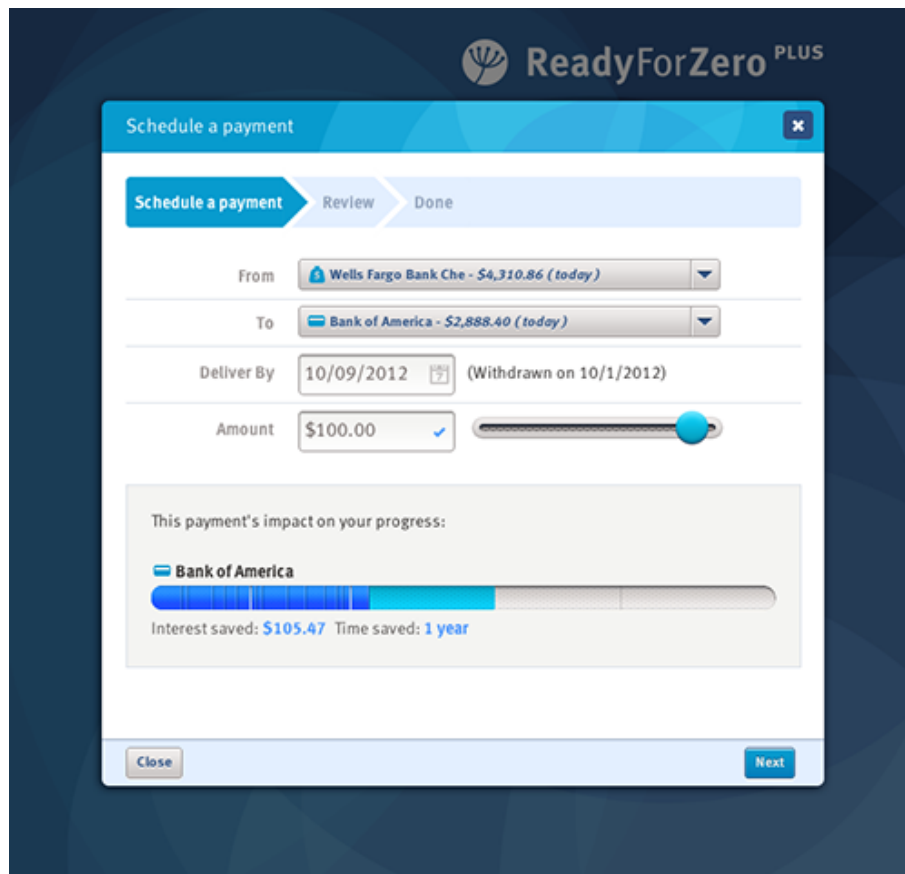
**Problem:** Feedback that isn't connected to the user's true motivations doesn't work sustainably.

"Gamified" feedback loops are able to make an impact, but points and badges that don't connect to anything meaningful won't last. People don't care about points; they care about things such as health, happiness, friendship and wealth. Right now, which would you rather have: 30 Foursquare points, 30 likes on your next Facebook update, 30 extra minutes in your day or \$30? The answers may vary, but I'll bet the Foursquare points would be last on the list for everyone.



**Solution:** Discover what your users care about, give them the ability to reach those goals, and present feedback connected to that. Social rewards, such as comments on our thoughts and photos, connect to the need for human relationships. Feeling fit and looking great connects to the goal of being healthy. Money connects to the many things people spend it on.

On ReadyForZero, we help people pay down their debt by showing them a number of bits of important feedback. First, we tangibly show the progress that a potential payment would make towards their goal on a progress bar. The visual connection enhances the feeling of achievement. Secondly, we tell them the long-term implications of that payment in terms of interest and time saved.



*ReadyForZero shows the impact of a payment to credit-card debt, including real time and money saved.*

Fully grasping what your users' motivations are can be difficult. There are many theories<sup>82</sup> about how motivation works, each with implications for design. A useful framework is the theory of self determination<sup>83</sup>, which posits that people who have most of their basic needs met can begin to look for autonomy, relatedness and competence. The theory is that these motivations arise not from the outside world, but from inside, and are more powerful. Theory aside, however, the tried and true approach of speaking candidly and in depth with your users will reveal what they truly care about.

## Two Things To Remember

With these categories in focus, you're on your way to designing better feedback loops. However, keep two things in mind as you go.

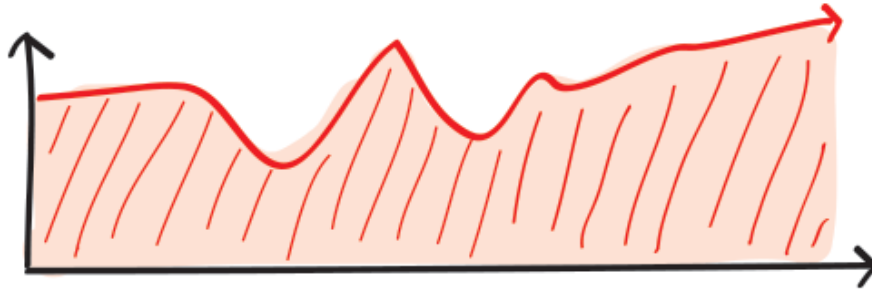
---

<sup>82</sup>. <http://en.wikipedia.org/wiki/Motivation>

<sup>83</sup>. [http://en.wikipedia.org/wiki/Self-determination\\_theory](http://en.wikipedia.org/wiki/Self-determination_theory)

## 1. MEASURE YOUR SUCCESS

Now that you've identified your areas for improvement and made changes to them based on our criteria, it's important to measure the outcomes of these changes using analytics and user research. As stated, you can't improve what you can't measure.



Analytics and research are, in fact, your own internal feedback loop as a designer. Consider the key metrics to track and report, how often they are reported, and in what context. The design of this feedback loop will affect how you respond to user behavior, and it's important that your team is optimizing for the right things.

## 2. BE ETHICAL

Finally, throughout the process, evaluate the ethics involved in each decision. The power of behavioral techniques should always inspire a discussion within your team as well as within yourself. Are you designing a health improvement app, a utility or a slot machine? Which of these is acceptable? Taking responsibility for influencing behavior requires a strong moral compass. Products that have taken the low road have been both successful and harmful.

Zynga has created a business model around extremely effective feedback loops, but the values informing many of these loops have been routinely<sup>84</sup> criticized<sup>85</sup> as being unethical. For example, Zynga uses progress bars, goals, levels and social pressures very effectively to monetize its signature game, Farmville. However, such games arguably succeed at the expense of the user's time, cash and creativity. They focus on achieving business goals by manipulating user emotions.

<sup>84</sup>. [http://en.wikipedia.org/wiki/Zynga#Reception\\_and\\_controversies](http://en.wikipedia.org/wiki/Zynga#Reception_and_controversies)

<sup>85</sup>. <http://www.theatlantic.com/technology/print/2012/01/the-zynga-abyss/251920/>



*Cityville<sup>86</sup>, another popular Zynga game, uses progress bars, levels and cleverly timed loops to drive engagement.*

Defining “fun” is beyond the scope of this chapter, but the clever timing of cash-based advances in gameplay is clearly designed for profit rather than for the player’s experience. Read about the fascinating Cow Clicker saga<sup>87</sup> for an amusing critique of this model.

For a more rational approach to designing ethically, review Stephen Anderson’s guidelines<sup>88</sup>, which take a deeper look at how to calibrate your moral compass.

In general, use the golden rule, and try to make the world a better place for everyone.

## *In Summary*

Design can be used to nudge behavior, and great feedback loops are a vital piece of pulling it off. Assist your users in understanding and achieving their goals by continually optimizing your feedback loops. Focus on making them fast, meaningful, quantified and contextual, and your users will thank you for improving their lives. 🐾

<sup>86</sup>. <http://company.zynga.com/games/cityville>

<sup>87</sup>. <http://kotaku.com/5846080/the-life+changing-20-rightward+facing-cow>

<sup>88</sup>. <http://uxmag.com/articles/towards-an-ethics-of-persuasion>

# Sketching For Better Mobile Experiences

BY LENNART HENNIGS 🐼

Mobile user experience design is maturing. One way to gauge this is to look at the tools at our disposal. Prototyping tools such as Balsamiq, Axure and Fireworks enable us to build wireframes and click-dummies, helping us to explain the targeted user experience. Cross-browser frameworks such as PhoneGap, Zurb Foundation and jQuery Mobile help us to create prototypes using the native languages of the Web: HTML, CSS and JavaScript.

## Why? The Case For Sketching

We seem to be in a better position than ever to design great experiences in virtually no time. However, these tools come with a hidden cost: they tempt us to skip the key step necessary to creating a well-designed product – which is to take the time to understand the problem we are given.

That's why I suggest to start using sketches to understand the problem and to come up with a concept before firing up your favorite tool.

Right now, you are probably thinking one of the following:

*“We have a tight deadline and there is no time for doodling. We need to get started.”*

*“Designing in [insert your favorite tool] is faster by far. I need to get results quickly.”*

*“We need to present results to our [client or project partner or peers]. We cannot show them hand-drawn stuff.”*

*“Sorry, but I really can't draw.”*

Not only have I heard these objections a couple of times before, I've also said all of them myself.

## DEFINING THE PROBLEM AND THE SOLUTION

Sketching enables us to explore the problem space and define the solution space at the same time. It frames our current understanding of the problem, while also helping us to figure out possible solutions.

While we sketch out our thoughts, new ideas will emerge. The ambiguity and lack of detail in sketches foster new ideas. Here, ambiguity is a good thing, because we will automatically try to fill in the blanks in our heads. That's what makes sketching "generative": it captures the ideas we already have and sparks new ideas. As Bill Buxton says in *Sketching User Experiences*:

*"Learning from sketches is based largely on the ambiguous nature of their representation. That is, they do not specify everything and lend themselves to, and encourage, various interpretations that were not consciously integrated into them by their creator."*

By sketching different solutions to a problem, we can explore them without immediately committing to one of them. This gives us new insight and raises new questions. Sketching is essentially a brainstorming exercise.

## THE "COST" OF SKETCHING

The fastest way to get something out of our heads is to grab a pen and a sheet of paper and quickly sketch it out. Trying to do the same in our favorite prototyping tool takes more time: We have to create a new project, decide on the right library to use, place our little rectangles on the canvas, draw little arrows connecting our little rectangles, rearrange everything a bit to make it look nice – and suddenly 30 minutes or more are gone.

If we don't like what we see on a sheet of paper, we can just toss it in the bin and start over. Using our prototyping tool, however, makes throwing stuff away harder because of the time and effort we've spent creating it – even if the idea wasn't a great one. Starting over is harder. Paper sketches, on the other hand, are cheap and fast.



*The fastest way to capture and evaluate a concept is to sketch it out.*

## SKETCHING AS A COMMUNICATION TOOL

However worn out the saying is, a picture does say a thousand words. Sketching enables us to walk our peers and stakeholders through our current understanding of a project. We can use various visual representations to explain different aspects of the project. In *The Back of the Napkin*, Dan Roam says:

*“Pictures can represent complex concepts and summarize vast sets of information in ways that are easy for us to see and understand, they are useful for clarifying and resolving problems of all sorts.”*

Our peers will be able to point out gaps in our thought process and discuss possible solutions and alternatives with us. They will most likely understand our shorthand and be able to give us feedback on the details of our concept. Presenting our idea to others also allows us reflect on our concept; we see mistakes and can find better alternatives just by walking through it.

## SKETCHING AS A COLLABORATION TOOL

Sketching is a powerful tool for involving stakeholders early on in the design process. The following chicken-egg problem is typical when defining the process: Stakeholders are not able to formulate a complete set of requirements before they have seen some visual representation of the solution, but we designers are reluctant to start working on a so-



lution before we are sure that the requirements we have been given are stable – we want to avoid unnecessary revision, which would result in extra work.

A way out of this dilemma is to start sketching possible solutions in a collaborative workshop, such as a design studio session (see Will Evans’ article “[Introduction to Design Studio Methodology](#)<sup>89</sup>”). We can walk stakeholders through our concept (and alternatives) step by step and explain to them the impact of their requirements on the design.

Sketching also has a low barrier to entry, allowing non-designers to participate. (Whether you hand out pens to stakeholders is a matter of personal taste and of much debate. You’ll have to make up your own mind.)

Because sketches are rough and unfinished, giving feedback is also easier. Some people hold back upon seeing a pretty finished-looking version of a page because they think of all the work that must have surely gone into it. High-fidelity visuals also tend to distract people – instead of focusing on the concept, they are tempted to comment on the visual design or on minor details. Sketches keep them focused on the core concept.

The only thing you’ll need to overcome is your reluctance to show your unpolished work to others.

## SKETCHING IS NOT DRAWING

A common misunderstanding is that your sketches have to be pretty. But interaction design is not art class. Your sketches don’t need to look good; they only have to convey your idea. They are supposed to spark discussion and generate ideas; they are unlikely to be framed on a wall. As long as you are able to draw boxes, arrows, circles and stick figures, you are good to go. As Joshua Brewer notes in “[Sketch, Sketch, Sketch](#)<sup>90</sup>”:

*“The sketch is not the end goal. The end goal of the drawing process is what you learn while sketching. So don’t worry if you can’t sketch.”*

Learning how to sketch is beyond the scope of this chapter, but look at the following presentations, article and book to get the basics:

- “The ‘Art’ of Sketching Interfaces<sup>91</sup>” (slides), Jason Mesut and Sam Smith

---

89. <http://uxmag.com/articles/introduction-to-design-studio-methodology>

90. <http://52weeksofux.com/post/346650933/sketch-sketch-sketch>

91. <http://de.slideshare.net/pubsmith/sketching-interfaces-workshop-interactions12-dublin>

- “Sketching 101<sup>92</sup>” (slides), Jackson Fox
- “The Messy Art of UX Sketching<sup>93</sup>,” Peiter Buick
- *Sketching User Experiences: The Workbook*, Saul Greenberg, Sheelagh Carpendale, Nicolai Marquardt, Bill Buxton (Morgan Kaufman: 2011)

## How To Sketch

Now that I have (hopefully) convinced you that sketching should be a part of your workflow, let’s talk about what to sketch in a mobile project.

### DIVERGENT SKETCHING

In the beginning, you will want to capture different ideas for your mobile app or website. Start sketching different versions of a single key screen or parts of it. Your goal is twofold: to generate a lot of different ideas, and to explore and evaluate them — at the same time. The more ideas you come up with, the more options you will have to choose from later on. This process is often called “divergent sketching.”

A useful tool for this type of sketching is a multi-page template. It offers enough space to sketch six different versions on a single page, and it provides some guidance on the available screen space. By showing different ideas side by side, it makes comparing and discussing them later on easier.

A lot of templates are available online<sup>94</sup> (I use Erik Loehfelm’s<sup>95</sup> most often). Pick the one you like.

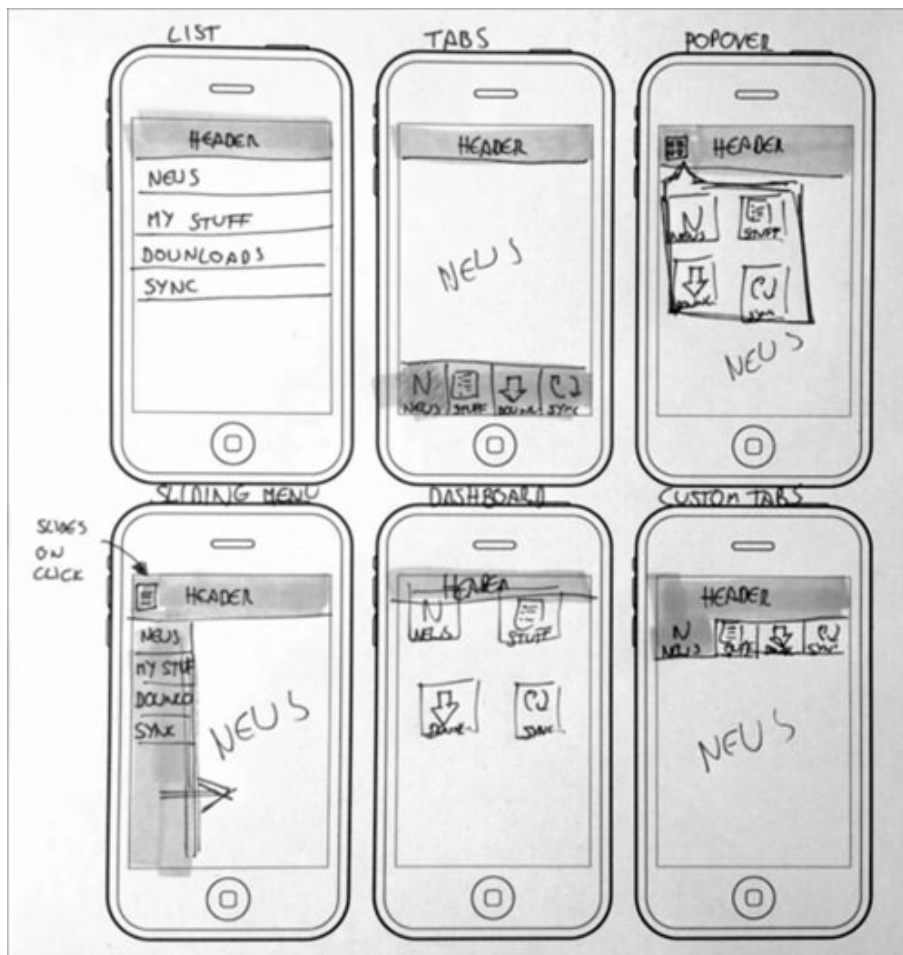
---

<sup>92</sup>. <http://de.slideshare.net/jacksonfox/sketching-for-design>

<sup>93</sup>. <http://uxdesign.smashingmagazine.com/2011/12/13/messy-art-ux-sketching/>

<sup>94</sup>. <http://www.smashingmagazine.com/2010/03/29/free-printable-sketching-wireframing-and-note-taking-pdf-templates/>

<sup>95</sup>. <http://erikloehfelm.blogspot.de/2009/05/iphone-ux-sketch-templates.html>



Wireframe variants for global navigation. (Large version<sup>96</sup>)

The screenshot above shows six different ideas for an app's main menu (plus vivid proof that my sketches are not pretty). I don't often come up with six versions (I like to tell myself that I am a structure-first person), but I do try to create at least three variants of a single key screen. Because this is brainstorming and you will want a lot of options to discuss later, quantity is your friend here.

If you happen to be short of ideas, browse some mobile UI galleries for inspiration (such as [Inspired UI](http://inspired-ui.com/)<sup>97</sup>, [Pptrns](http://pttrns.com/)<sup>98</sup>, [Lovely UI](http://www.lovelyui.com/)<sup>99</sup> and [UI Parade](http://www.uiparade.com/)<sup>100</sup>). They showcase solutions grouped by topic and feature (for example, main menus, chat windows). Include the things you like in your sketches, and mix them with your own ideas.

Make a habit of titling each sketch. This will make it easier to differentiate between them and refer to them later on. I add notes and anno-

<sup>96</sup>. [http://media.smashingmagazine.com/wp-content/uploads/2013/06/6variants\\_large\\_mini.png](http://media.smashingmagazine.com/wp-content/uploads/2013/06/6variants_large_mini.png)

<sup>97</sup>. <http://inspired-ui.com/>

<sup>98</sup>. <http://pttrns.com/>

<sup>99</sup>. <http://www.lovelyui.com/>

<sup>100</sup>. <http://www.uiparade.com/>

tations to explain the sketches and the thought processes behind them (such as strengths, weaknesses, trade-offs compared to other concepts, questions, new features, etc.). Labelling and dating the sheets are also a good practice.

## CONVERGENT SKETCHING

Once you have sketched multiple versions of a screen, pick the one that best solves the problem. Unfortunately, a single sketch will rarely check all of the boxes. In most cases, you will have to combine ideas or parts of ideas into a single winning concept. To do this, you will need to explore in more depth by making a more detailed sketch on a single page. Use the white space to annotate. Jot down all questions, new ideas, important or unclear areas, and things to discuss. This will help others to understand your thought process. This activity is often called “convergent sketching.”

For more information on divergent and convergent sketching, see Leah Buley’s presentation from FailCon 2010, “Good Design Faster<sup>101</sup>,” and Brandon Shauer’s article “Sketchboards: Discover Better + Faster UX Solutions<sup>102</sup>.”

## CREATING UI FLOWS

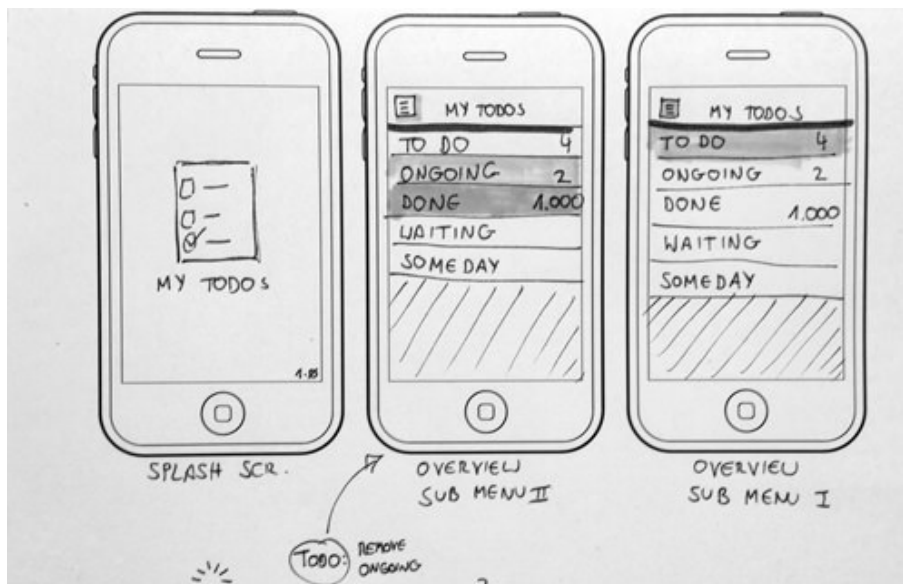
After boiling your ideas down to a set of key screens, the next step is to explore how they work together. For this, create a few UI flows (i.e. a series of key screens) that showcase how a user would use your solution to accomplish a task. UI flows highlight which interface elements are used (for example, which button is clicked or which gesture is used) and how the system responds (for example, with an animation, transition, pop-up dialog or new screen). They also show key screens in different states (for example, initially empty and then later filled with content).

You can visualize different outcomes within a UI flow (such as a list of search results versus an empty list). Your flow would then no longer be linear; different branches would show different outcomes. But try to limit the number of branches in a single UI flow. Each branch increases the complexity, making your flow harder to understand. It also makes them harder to explain.

---

<sup>101</sup>. <http://www.slideshare.net/webwallflower/good-design-faster-slides-failcon-2010>

<sup>102</sup>. <http://www.adaptivepath.com/ideas/sketchboards-discover-better-faster-ux-solutions>



UI flows visualize how a user gets from A to B. (Large version<sup>103</sup>)

You don't need to sketch all use cases; pick the ones that are important and will be used most. The Pareto principle<sup>104</sup> offers a good rule of thumb: prototype the 20% of functionality that will be used 80% of the time.

I usually start sketching one of the key use cases to try to come up with a way to solve it. I will highlight the interactions on the screens and (usually) connect the screens with arrows. I'll explain each step and label the key screens. As always, I annotate wildly.

For inspiration on how to sketch mobile screens and UI flows, look at Gisele Muller's article "Inspiring UI Wireframe Sketches"<sup>105</sup>; MOObileFrames<sup>106</sup>, a blog that showcases mobile wireframe sketches; and Wireframes<sup>107</sup>, a website by Jakub Linowski. You can also use Jakub's "Interactive Sketching Notation"<sup>108</sup> (PDF) in your UI flows.

## STEPS TO TAKE

To give you some more context and to show you how the three activities above relate to each other, here are the steps of a typical sketching session:

- List the information you have and what you want to capture.

<sup>103</sup>. [http://media.smashingmagazine.com/wp-content/uploads/2013/05/ui-flow\\_large\\_mini.jpg](http://media.smashingmagazine.com/wp-content/uploads/2013/05/ui-flow_large_mini.jpg)

<sup>104</sup>. [http://en.wikipedia.org/wiki/Pareto\\_principle](http://en.wikipedia.org/wiki/Pareto_principle)

<sup>105</sup>. <http://webdesignledger.com/inspiration/inspiring-ui-wireframe-sketches>

<sup>106</sup>. <http://moobileframes.tumblr.com/>

<sup>107</sup>. <http://wireframes.linowski.ca/>

<sup>108</sup>. [http://www.linowski.ca/downloads/ISN\\_1.2\\_Introduction.pdf](http://www.linowski.ca/downloads/ISN_1.2_Introduction.pdf)

- Make a first set of sketches (such as variations of a key screen or an initial UI flow).
- Review the sketches yourself.
  - What are the strength and weaknesses of each variant?
  - Are the UI elements and data consistent (i.e. the same elements are used for the same task)?
  - Do you show data the same way?
  - Are the interaction options clear?
- Get some feedback from others. Start with peers and continue with potential users if possible.
  - What are their first impressions?
  - What do they like best? Why?
  - What do they dislike? Why?
  - What is not clear in your sketch?
  - What improvements have they suggested? Why did they suggest them?
- Iterate on the concepts.
  - Can you combine different strengths of separate concepts into a new concept?
  - Can the sketches be clarified?
  - Can steps be reduced or simplified?
- Review the results in light of the initial feedback.
  - Have new questions come up?
  - What insights have you gained?

As you can see, a lot of these are “Why?” questions. This is because sketching is about trying to understand the problem while simultaneously working towards a solution. Write down all of the questions that come up during sketching next to the sketches themselves. This will guide you towards the right concept.

## BEYOND SKETCHING

After you've created sketches of the key screens and the main use cases, you will want to try out your concept live on a real phone. Apps such as Pop<sup>109</sup> and Protosketch<sup>110</sup> enable you to import pictures of sketches and turn them into a prototype. This very fast and low-fi method will give you a feel for the concept's interaction. Both apps allow you to define clickable hotspots and page transitions, which make a prototype's interaction more realistic.

Alternatively, import your screens into a more advanced tool such as Axure and build a prototype in there.

Whatever method you choose, your goal is to test the concepts quickly and with little effort on a real phone.

## Take-Aways

Sketching helps you better understand the problem you are trying to solve and lets you visualize possible solutions. It is a fast and inexpensive way to brainstorm and to test out a lot of UI ideas before committing to one. Sketching speeds up the concept creation and iteration phase and makes it possible to get feedback early on, when changes are easy to make.

For your next sketching session, keep the following principles in mind. They will help to keep you on track:

- **Be lazy.**  
Don't try to reinvent the wheel. Use a template to guide your sketching.
- **Get inspired.**  
Browse design pattern libraries and mobile UI galleries, and look at wireframe showcases. Take what you like, mix it with your own ideas, and come up with something new.
- **Stop at good enough.**  
Your sketches need to get your ideas across. Don't get lost in (unimportant) details.
- **What would Pareto do?**  
A small set of your solution's features will be used the most. Focus on them. Which 20% of the solution will be used 80% of the time?
- **Be consistent.**  
Use an existing sketching notation and/or develop your own. This will

---

<sup>109</sup>. <http://popapp.in/>

<sup>110</sup>. <http://protosketch.uistencils.com/>

make your sketches reliable. Peers will understand your style and will need less time to look through the sketches to give feedback.

- **Annotate everything.**

During sketching, new questions, ideas and problems will pop up. Write them down; otherwise, they will get lost.

- **Be open.**

Make your sketches visible. Pin them on a wall, and always show them to others to get feedback.

- **When in doubt, sketch some more.**

If you are unsure about a solution, start sketching variations. List their pros and cons and get feedback to see what works best.

Happy sketching! 🐼



# Converting Our Stories Into Multi-Screen Experiences

BY SENONGO AKPEN 🍷

Storytelling takes many forms. In the past, stories were told orally, with people telling and retelling myths, fables and even histories. As writing technology became more prevalent, we began to record our stories<sup>111</sup>, and we told them in the pages of books. Now, our society is awash in different devices and technologies, and those traditions of spoken stories and printed stories are blurring.

Multi-screen narratives are being told across all kinds of platforms, pages and devices, making for truly immersive experiences. We are watching them, tapping them and learning from them. They immerse us in the storyteller's world. This chapter outlines what I believe are the five essentials of telling multi-screen stories.

## *How I Fell In Love With Interactive Storytelling*

First, a little background. My childhood was spent in Nigeria, West Africa. I am a member of the Tiv tribe, a group of about 6 million people clustered in Nigeria's Benue River Valley. As a child, I heard a lot of Nigerian folktales, about animals, humans and even magic. In Nigerian narrative tradition, stories are often told orally, in front of a gathered audience. During festivals<sup>112</sup> and cultural events, men even dress up in elaborate costumes<sup>113</sup> and perform stories for the crowds. I have vivid memories of these stories and have always been curious about how they could be translated into something digital and interactive.

---

<sup>111</sup>. [http://en.wikipedia.org/wiki/Brothers\\_Grimm](http://en.wikipedia.org/wiki/Brothers_Grimm)

<sup>112</sup>. [http://www.youtube.com/watch?v=Zx4NvT\\_DVX4&feature=player\\_embedded](http://www.youtube.com/watch?v=Zx4NvT_DVX4&feature=player_embedded)

<sup>113</sup>. <http://vimeo.com/18917127>



*The Kwagh Hir, or Thing of Magic, my tribe's largest cultural festival (Image: [Naptuz<sup>114</sup>](#)).*

Those fables are a piece of my cultural inheritance. They always seemed to contain essential truths about humans. Take the story of the Ear and the Mosquito. One day, the Ear steals food from the Mosquito and refuses to pay it back. In anger, the Mosquito visits the Ear every evening, demanding the food to be returned, annoying Ear all night with his buzzing. It's an old tale, [with many versions<sup>115</sup>](#), but the moral is consistent: don't steal from your friends.

Creating modern, interactive versions of these stories is possible, but how exactly do we do that? Let's begin by talking about what I mean by the word "multi-screen."

## *A Bit About Context And Screens*

When speaking about multi-screen storytelling, remember that screens have different contexts<sup>116</sup>, not only different capabilities. The same screen on which you carelessly watch videos at home becomes a closely guarded viewport when you're watching a movie on a crowded train. The context in which people view stories is more important than the

---

<sup>114</sup>. <http://www.nairaland.com/986883/some-funny-really-nice-nigerian/4#12362934>

<sup>115</sup>. <http://www.amazon.com/Why-Mosquitoes-Buzz-Peoples-Ears/dp/0140549056>

<sup>116</sup>. <http://www.cennydd.co.uk/2013/designing-with-context>

device's specifications. When we tell interactive stories, we need to be aware of this, and embrace it.

I like to focus on the following screens:

- Sensors ([Twine<sup>117</sup>](http://www.gimcrackd.com/etc/src/), GPS, [Arduino<sup>118</sup>](http://www.arduino.cc/), motion detectors, etc.)
- Mobiles and tablets (phones, tablets, laptops and everything in between)
- Flat-screens (desktops, TVs, etc.)
- Public and immersive displays (store kiosks, large stadium screens, projectors, Kinects, etc.)

Not all of these need to be used at the same time, because they won't all be appropriate to the story you are telling. Context is extremely important.

Now, as promised, here are the five essentials of multi-screen storytelling.

## *1. Divide Your Story Into Separate Content Blocks*

When we create multi-screen narratives, we need to find natural break-points in the story, places where the visual or narrative content can easily be separated. This enables us to deliver different segments to different devices, in different contexts.

Kolobok is a Slavic children's story about the adventures of a round yellow cake. For the Moscow International Festival, a large team of designers and animators from SilaSveta Studio incorporated it into a [truly fascinating demonstration<sup>119</sup>](http://portfolios.risd.edu/gallery/Kolobok/5648689) of storytelling. Before the show, the team set up a large touchscreen at the children's height. With their hands, the kids could manipulate parts of the animation by adding movement and color.

---

<sup>117</sup>. <http://www.gimcrackd.com/etc/src/>

<sup>118</sup>. <http://www.arduino.cc/>

<sup>119</sup>. <http://portfolios.risd.edu/gallery/Kolobok/5648689>



*A public display for children to play with.*

For the show itself, the full story was projected onto the facade of a large building, allowing the crowd of adults and their children to watch the narrative unfold. Along with sound, it made for another discrete content block, one that closely resembled a 3-D movie.



*The full animated story in front of a large festival crowd.*

While the touchscreen and the movie were different views of the same content, they could exist as independent pieces and did not have to appear next to each other. The SilaSveta team found the natural break-points in its story and created two separate visual experiences to match them.

## QUESTIONS TO ASK

- **Where are the breakpoints in the story?**

Divide your content so that it makes sense in context. The practice of responsive design gives us numerous guidelines<sup>120</sup> on how to do this.

- **Can those content blocks exist independently?**

Sometimes, the answer is no, but it depends on the story. In the Kolobok example above, they can. In other interactive stories, such as Snow Fall<sup>121</sup> from the New York Times, the blocks are chapters in a single story and should be kept together.

## 2. Offer People Multiple Perspectives

Bear 71<sup>122</sup> is an award-winning multi-screen experience created by Jeremy Mendes and Leanne Allison. The creators tell the story of a bear living in Banff National Park in Canada. It feels like a cross between a role-playing game and a TV documentary, and as a linear narrative with a non-linear interface<sup>123</sup>, it works beautifully.



*The Bear 71 story<sup>124</sup> is told in a highly abstracted interface.*

Multiple viewpoints are accessible. Online, you roam in a stylized landscape, watch crittercam footage from the forest, and otherwise live as

---

<sup>120</sup>. <http://www.markboulton.co.uk/journal/a-richer-canvas>

<sup>121</sup>. <http://www.nytimes.com/projects/2012/snow-fall/#/?part=tunnel-creek>

<sup>122</sup>. <http://bear71.nfb.ca/#/bear71>

<sup>123</sup>. <http://www.youtube.com/watch?v=58jHqrymqPM>

<sup>124</sup>. <http://bear71.nfb.ca/#/bear71>

bears do — freely. Even though it may look like a game interface, you are not so much “playing” as you are participating in a story. Watching real crittercam footage, you see what the forest silently sees. You also have the option to turn on your webcam (“stealth mode”) to see other users around the world, all watching the same story online.



During shows and installations, the team responsible for Bear 71 set up augmented-reality applications and motion-detection cameras, so that visitors could experience what it’s like to have their pictures taken<sup>125</sup> with one. By playing with the augmented-reality apps and the motion-detection cameras at the installations, users got a bit of the same physical experience that the bears had.

## QUESTIONS TO ASK

- **Does the narrative change when viewed from a different perspective?**

A variety of perspectives can make a narrative much more fascinating. Bear 71 forces us to see the world first from the bear’s perspective and to sympathize with its loss of habitat, but other viewpoints take a slightly different angle. The voyeurism common in our digital sharing

---

<sup>125</sup>. <http://iambear71.tumblr.com/post/16604858064/walls-of-surveillance-images-of-bear71-subjects>

culture takes on a different meaning when used for animal surveillance.

- **What data sets can be used in the narrative?**

Bear 71 cleverly combines crittercam video, GPS data, cell tower data, augmented reality, and topographical data. The photographs of visitors to the installation provide an additional emotional layer of data. The data we bring into our stories helps to define additional viewpoints and characters.

### *3. Redefine A Tradition*

As Western culture has moved more deeply into Nigeria, Nigeria's traditions are weakening. I wanted to take a piece of my culture and put it in the cloud, instead of leaving it locked in the heads of our oral storytellers. That meant redefining how the stories are relayed, how they are saved and, most importantly, what messages they convey to the audience.

In 2011, I started a project named Pixel Fable<sup>126</sup> in which I take those traditional stories and reinterpret them online. In essence, I'm creating an interactive archive of Nigerian stories. As mentioned earlier, the oral histories of Nigerians are rich, but capturing them and translating them into digital stories means they will reach a wider audience. About 25% of my website's visitors come from the US, while another 25% come from Japan. Canada, France and Germany also send a fair amount of traffic.

Pixel Fable uses responsive websites, iOS apps and augmented-reality animations to reinterpret Nigeria's oral history.

---

<sup>126</sup>. <http://pixel-fable.com/>





An introductory screen from “Cricket and Mud Brick<sup>127</sup>,” a new Pixel Fable story built with the Tapestry app.

I’ve relied on two primary contexts to reinterpret the old Nigerian storytelling tradition. The first is people on their tablets and phones, clicking on and reading the stories. The spread of mobile devices<sup>128</sup> makes this inevitable — why not tell African fables in a more accessible context? The second is my attempt to update the moral lessons for our modern age. While the original story of the Ear and the Mosquito may be a funny tale about annoying insects, the lesson can be updated to speak about how mosquitoes spread malaria in Nigeria. There’s room to redefine our old myths for the 21st century.

## QUESTIONS TO ASK

- **Will people love or hate the reimagined version?**

Not every fable or myth can (or should<sup>129</sup>) be recreated digitally. However, if people have an emotional reaction to a story that you have designed and pushed out to multiple screens, that is usually a good sign.

- **If people talk about your narrative, will it bring about change in society?**

Each Pixel Fable story has a message. Most fables do. Some revolve

---

<sup>127</sup>. <http://tapestry.is/yJe4E/>

<sup>128</sup>. <http://www.nielsen.com/us/en/newswire/2013/the-teen-transition--adolescents-of-today--adults-of-tomorrow.html>

<sup>129</sup>. <http://www.npr.org/2012/03/02/147573582/the-lorax-a-campy-and-whimsical-seussical>



around love triangles, others around the wisdom of elders, and there's even one about why you shouldn't get angry at your friends. They are small messages, but put together, they force us to reconsider how we treat the narrative history of people in Nigeria and West Africa.

#### 4. Immerse People In The Narrative

The Walking Dead, the famous comic and now TV show, used a polling Web app<sup>130</sup> (AMC's Story Sync, if you like marketing-speak) to ask viewers questions and show related content as an episode was being broadcast on TV. While the app was simply timed to each scene, it was an experiment in multi-screen storytelling that invited audience participation, not just audience attention. Polling has a gimmicky feel to it<sup>131</sup>, but that probably came about as a result of Hollywood pressure and doesn't reflect the value of the concept in general.



*Polling and syncing apps extend narratives from the TV to the couch.*

The creators also added mobile gaming<sup>132</sup> to the mix, bringing viewers “into” the story in a completely different way, in different contexts.

All of these facets of each story's arc enabled people to immerse themselves in this apocalyptic narrative. Jason Spero of Google notes<sup>133</sup> the need for a seamless experience as users move between devices. Oth-

---

<sup>130</sup>. <http://www.amctv.com/shows/the-walking-dead/story-sync>

<sup>131</sup>. <http://www.engadget.com/2013/01/11/the-state-of-the-second-screen/>

<sup>132</sup>. <http://www.amctv.com/shows/the-walking-dead/mobile>

<sup>133</sup>. <http://gigaom.com/2012/08/29/multi-screen-mania-how-our-devices-work-together/>

er people, however, say that a second-screen experience can be extremely distracting, forcing viewers to miss key parts<sup>134</sup> of the TV show. It is the opposite of an immersive experience, they say, and is confusing to use. In my opinion, each content block should work independently to avoid putting users in this position.

## QUESTIONS TO ASK

- **Will people forget where they are?**

I'll be the first to admit that this is not always a good thing. I can't count the number of times that I've almost missed my train stop because my head was buried in my phone. Context, not only device capability, is key. Do you want users to get lost in the story or just engage in a manageable chunk?

- **Do the screens you have chosen feel natural?**

By this, I'm not referring to pixel density. That is simply impossible to control, and if the story is good, it won't matter anyway. The screens that people choose will depend a lot on the tasks they want to complete, so make your story feel natural for whatever content block they are interacting with.

## 5. *Design Contextual Interactions*

Recently, a number of storytelling apps have relied on location to serve additional content, much in the same way that Foursquare or Google Maps do. The Silent History<sup>135</sup> is a dystopian science-fiction story about children who do not speak. The iPad app contains the whole story, but by visiting certain geo-tagged locations, users can access additional content.

---

<sup>134</sup>. <http://gigaom.com/2012/09/09/can-breaking-bads-story-sync-get-viewers-to-give-up-their-dvrs/>

<sup>135</sup>. <http://www.thesilenthistory.com/>



For a novel about children all over the US, inviting readers to physically go to where the story's kids are makes perfect sense. The additional contextual interaction makes the story more layered and thought-provoking, in a way that a simple app would not be.

We use map data every day, to look for restaurants, check the weather, see road conditions and even check for public transit delays. Other contextual interactions make sense when creating digital stories: taking photographs, texting, sharing and saving information, even body motion<sup>136</sup>. Use these, along with your UI and UX skill sets, to devise new storytelling methods.

## QUESTIONS TO ASK

- **Does the device matter?**

With the rise of responsive design workflows, our content should not be device-dependent. Some things, however, such as camera or GPS functionality, may be integral to a part of your story, and so the device would need to be factored in.

- **Should the interface be designed as a seamless part of the narrative?**

As people who work on the Web, we really have a strength here. If we choose to make the interface part of the story, then we can rely on our experience in building websites and content management systems.

- **Will your story “remember” anything?**

As a child, I folded over the top corner of the page when I had to put a book down. It was a simple way to keep my place. With a narrative split

---

<sup>136</sup>. <https://getmyo.com/>

across multiple devices, it might be necessary to design an interaction that flags where you've gotten to and then returns you there when you visit again. That all depends on the content, but the question does need to be asked. Everyone hates losing their place on the Internet and having to navigate back, so perhaps we should enable a memory in our stories as well.

## *Conclusion*

We have conceptualized different uses of multiple screens to tell stories. All of us, from every corner of the globe, have intensely rich cultures filled with stories and fables. Using them to create interactive narratives is another way to explore the power of the Web<sup>137</sup>, to wow people and to record our cultural history.

I would love to see what you come up with, or hear about other examples of clever digital storytelling. 🐼

---

<sup>137</sup>. <http://www.react-hub.org.uk/books-and-print-sandbox/commissions/2013/>

# Creating An Adaptive System To Enhance UX

BY AVI ITZKOVITCH 🍷

In computer science, the term “adaptive system” refers to a process in which an interactive system adapts its behavior to individual users based on information acquired about its user(s), the context of use and its environment. Although adaptive systems have been long-discussed in academia<sup>138</sup> and have been an aspiration for computer scientists and researchers<sup>139</sup>, there has never been a better time than today to realize the potential of what future interaction with computer systems will be like.

The abilities of today’s network information technologies to create rich, immersive personalized experiences to track interactions and aggregate and analyze them in real time, together with the data collected by the sensors we carry in our smart devices, provides us an opportunity like never before to design adaptivity in order to ultimately offer a better user experience that is both unobtrusive and transparent.

This chapter will cover the fundamental concepts for utilizing smart device technologies and sensor data in order to understand context and introduce “adaptive thinking” into the UX professional’s toolset. I will demonstrate the importance of context when designing adaptive experiences, give ideas on how to design adaptive systems, and perhaps inspire designers to consider how smart devices and context aware applications<sup>140</sup> can enhance the user experience with adaptivity.

## *Examples Of Adaptive Systems*

An early example of an adaptive feature can be found in GPS navigational devices. Using one of these devices, a user is able to easily locate and navigate to any location they can drive to. When the sun sets or while driving through a tunnel, the system automatically changes the interface color to a dark “night mode” so as not to blind the driver with a bright light from the device. The system knows the user’s exact location and the position of the sun, and by understanding these two fac-

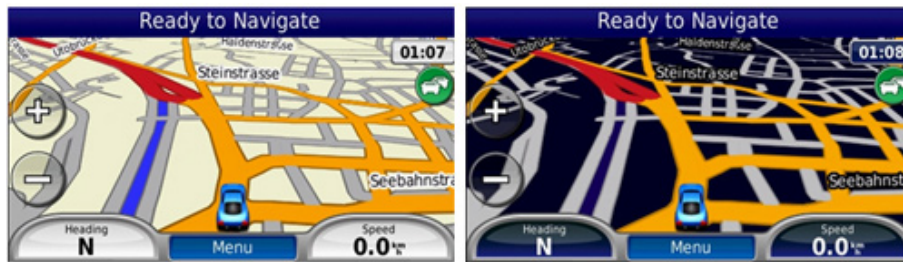
---

<sup>138</sup>. [http://scholar.google.com/scholar?hl=en&q=adaptive+system&btnG=&as\\_sdt=1%2C20&as\\_sdtp=](http://scholar.google.com/scholar?hl=en&q=adaptive+system&btnG=&as_sdt=1%2C20&as_sdtp=)

<sup>139</sup>. <http://research.microsoft.com/en-us/groups/adapt/>

<sup>140</sup>. [http://en.wikipedia.org/wiki/Context\\_awareness](http://en.wikipedia.org/wiki/Context_awareness)

tors, the system maintains a safe driving environment by adapting to the user's needs.



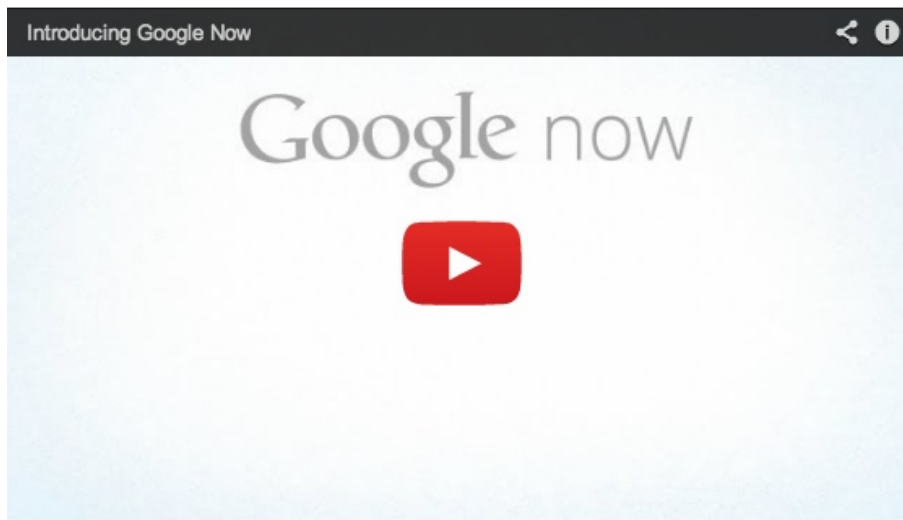
*The day and night interfaces in the GARMIN Zumo 660<sup>141</sup> adapt the interface color so the user isn't blinded with a bright light.*

Adaptive design is about listening to the environment and learning user patterns. Combining smart device sensor data, network connectivity and analysis of user behavior is the secret sauce behind creating an adaptive experience. By combining these capabilities, we not only understand the context of use, we can also anticipate what the user needs at a particular moment.

Google Now<sup>142</sup> is an interesting example of an adaptive application that gives users answer to questions they've thought rather than typed. Through a series of smart cards that appear throughout the day on the user's mobile phone, Google Now tells you the day's weather before you start your day, how much traffic to expect before you leave for work, when the next train will arrive as you're standing on the platform or your favorite team's score while they're playing. It does this by recording and analyzing your preferences while you're using your phone. For example, updates on your favorite sports team are based on your Web browsing and search history. And by analyzing your current location, previous locations and Web history, Google Now presents a card with traffic conditions on route to your next likely destination.

<sup>141</sup>. [http://www.tramsoft.ch/gps/garmin\\_zumo660\\_en.html](http://www.tramsoft.ch/gps/garmin_zumo660_en.html)

<sup>142</sup>. <http://www.google.com/landing/now/>



As UX professionals, we understand that some mobile users do not like to use the virtual keyboard and we try to avoid that necessity as much as possible. By utilizing the user's personal behavior as a sensor together with smart device capabilities and enabling voice commands<sup>143</sup> (similar to iOS's Siri), Google Now creates an adaptive experience that helps users avoid using the virtual keyboard, thus further adapting to the mobile user's needs and helping users quickly get the information they require on the go.

Adaptive systems are not only limited to mobile devices. Ubiquitous computing<sup>144</sup> (ubiquomp) is the idea of being surrounded by smart devices and networked digital objects that are carefully tuned to offer us unobtrusive assistance as we navigate through our work and personal lives. Similarly, ambient intelligence<sup>145</sup> (AmI) refers to digital environments that are sensitive and responsive to the presence of people.

---

<sup>143</sup>. <http://www.youtube.com/watch?v=fHkhp6BwnGo>

<sup>144</sup>. [http://en.wikipedia.org/wiki/Ubiquitous\\_computing](http://en.wikipedia.org/wiki/Ubiquitous_computing)

<sup>145</sup>. [http://en.wikipedia.org/wiki/Ambient\\_intelligence](http://en.wikipedia.org/wiki/Ambient_intelligence)





*Nest uses sensors to adapt the temperature to activity in the home.*

Nest<sup>146</sup>, The Learning Thermostat, is a great example of an adaptive system integrated to home environments. Using a variety of sensors<sup>147</sup> for temperature, humidity, touch, near-field activity, far-field activity and even ambient light, it can detect whether there are people home and how active the home is at any time. By adjusting the temperature to adapt to this information, it can automatically cut up to 20% off a home's heating and cooling bills.

When no one is around, Nest learns to turn the heat down<sup>148</sup>. When you come home from work, it knows that the heat should go back up. After the first few weeks, it learns when you come home from work and can turn the heat up before you arrive so that you come home to a warm house.

In 1991 Mark Weiser<sup>149</sup>, widely considered to be the father of ubiquitous computing, wrote:

*“The most profound technologies are those that disappear. They weave themselves into the fabric of everyday life until they are indistinguishable from it.”*

Nest is a great example of ubicomp and how technology can disappear into our surroundings until only the user interface remains perceivable to users.

---

<sup>146</sup>. <http://www.nest.com/>

<sup>147</sup>. <http://www.nytimes.com/interactive/2012/10/04/business/inside-the-nest-learning-thermostat.html>

<sup>148</sup>. <http://www.digitaltrends.com/hands-on-videos/how-the-nest-learning-thermostat-works/>

<sup>149</sup>. [http://en.wikipedia.org/wiki/Mark\\_Weiser](http://en.wikipedia.org/wiki/Mark_Weiser)



These devices create contexts of sensor and user data that provide a superior user experience by anticipating what the user might need before the need is expressed. This is the future of UX design.

## *Adaptive Thinking*

In contrast to traditional desktop systems, mobile devices are normally used in many different situations. However, mobile applications nowadays do not often take advantage of information about the context of their use, and hence are only usable for very specific purposes. For example, an application with city maps for local businesses can be used in different contexts: walking through town or at home, and with or without network connectivity.

Today's users can customize their device's system through preferences and settings, and by choosing what applications work best for their needs. Even after the implementation of user-centered design processes that assure a certain degree of user acceptance and yield a richer understanding of the context, it is impossible to anticipate the requirements of all users and map those requirements to a single best or optimal system configuration.

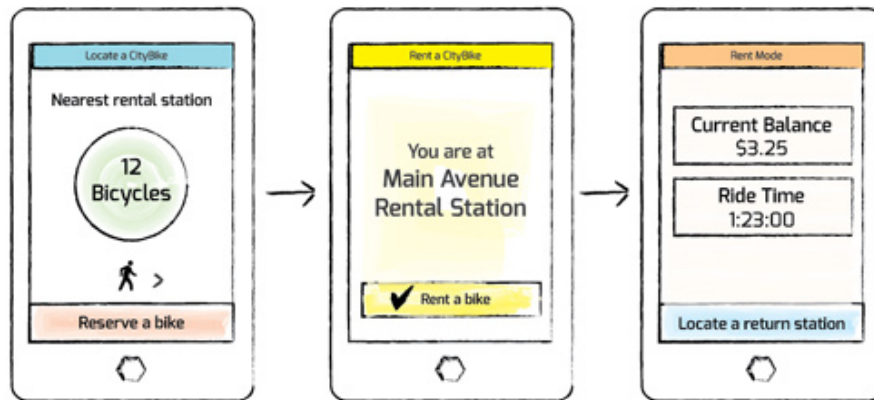
"Adaptive thinking" is a mindset that provides the tools necessary to significantly improve the user experience and enhance the intended purpose of the product by utilizing the technology that is readily available in every pocket. It is about learning the environment and the user and adapting to their current needs and situation. Therefore, designers should first design for the context of use and then design the set of functions that are triggered in relevant situations.

Here is an instructive case where *adaptive thinking* was used to create a mobile application for a bike sharing program. Bicycle sharing systems, also known as bike rental, are becoming more and more common in major cities around the world. Bicycle sharing helps reduce traffic congestion and air pollution, and encourages local residents to maintain a healthy lifestyle.

A user who wants to rent a bike can use a mobile application to look for the nearest bike rental station that has bikes available to rent. If the user is unfamiliar with the city, they can use the application to get directions to the rental station; this is the core functionality of the application.

An adaptive system will realize when the user has arrived at the bike rental station and automatically offer additional options, i.e., adapt to the current situation. For example, it may offer the user a quick way to rent a bike, a feature that was not available in the application before arriving at the rental station. During the rental period, the system will an-

anticipate the user's needs and offer nearby bike rental stations with available parking spots where the bike can be returned, and show the user the current balance for the rental time.



*A bicycle sharing application can adapt to show the user different options depending on location, and whether the user is currently renting a bike.*

By using the assisted GPS<sup>150</sup> device capabilities, using the network connectivity and understanding the user's story at any given time through the product lifecycle, adaptive design will provide users of the mobile application a reliable extension to the bike rental program.

## ADAPTIVE AND RESPONSIVE DESIGN

An adaptive system is one that adapts automatically to its users according to changing conditions. Responsive design (or adaptive layout<sup>151</sup>) is a subset of adaptive design, an approach to Web design in which a website is crafted to provide an optimal viewing experience across a wide range of devices. In my UX magazine article The Multiscreen Ecosystem<sup>152</sup>, I discuss how responsive websites can also be adaptive by understanding the context of using a mobile device and by designing contextual paths.

## Context For Adaptivity

I quote below from the 2007 book The Adaptive Web<sup>153</sup>, which talks about the importance of context for adaptive mobile guides. It explains

<sup>150</sup>. [http://en.wikipedia.org/wiki/Assisted\\_GPS](http://en.wikipedia.org/wiki/Assisted_GPS)

<sup>151</sup>. <http://uxdesign.smashingmagazine.com/2012/11/08/ux-design-qa-with-christian-holst/>

<sup>152</sup>. <http://uxmag.com/articles/designing-for-context-the-multiscreen-ecosystem>

<sup>153</sup>. [http://www.amazon.com/Adaptive-Web-Personalization-Information-Applications/dp/3540720782/ref=sr\\_1\\_1?s=books&ie=UTF8&qid=1347398939&sr=1-1&keywords=The+Adaptive+Web](http://www.amazon.com/Adaptive-Web-Personalization-Information-Applications/dp/3540720782/ref=sr_1_1?s=books&ie=UTF8&qid=1347398939&sr=1-1&keywords=The+Adaptive+Web)

adaptivity in the scope of mobile systems as context-aware computing<sup>154</sup>, i.e., the ability to use information in the current context to adapt the user interaction and the presentation of information to the current situations of the user.

*“Understanding the context is an important prerequisite for the adaptation process. Context is not just the location, but encompasses also information like the ambient noise or lighting level, the network connectivity or bandwidth, and even the social circumstances of the user. Furthermore, systems have to anticipate the user’s goals and intentions, which might be inferred from their actions or from physiological sensors and appropriate environmental sensors (e.g. light, pressure and noise sensors).*

*One prerequisite for adaptive systems is the proper assessment of the user’s situation. For this purpose, systems need to rely on a representation of relevant situations. Depending on the supported task, situations can be characterized by many different attributes. Therefore, designers of suitable adaptation for mobile devices need to look at a variety of spatial, temporal, physical and activity related attributes to provide effective assistance.*

*For example, a mobile application that assists users in a shop needs to know about the current spatial environment of the users (e.g. which products are nearby), the temporal constraints of the user (e.g. how much time is available for shopping), the general interests of the users and their preferences (e.g. if the user prefers red or white wine with tuna), details about the shopping task itself (e.g. which items are on the shopping list and for which purpose the products are needed) and maybe even about the physiological and the emotional state of users (e.g. whether users are enjoying the shopping or not).”*

That said, understanding the locational context and the user story is now easier than ever before. We can take advantage of the fact that we carry our phones wherever we go. A smartphone is packed with technology and with information about the user that designers can use to understand context. The highly sophisticated advanced technology in a user’s pocket not only allows designers to analyze if the user is walking, standing or in a loud or quiet environment, but also can help designers understand the precise location of a person within a department store, such as a specific aisle.

---

<sup>154</sup>. [http://www.interaction-design.org/encyclopedia/context-aware\\_computing.html](http://www.interaction-design.org/encyclopedia/context-aware_computing.html)



*An application can analyze the user's precise location within a store to provide information that is adapted to the current content.*

AislePhone<sup>155</sup>, an Israeli start-up currently in the beta stage, is developing a platform for precise in-store positioning that can determine the exact position of a person down to the specific aisle. With this technology, shopping with your mobile phone in hand will be a common experience, as mobile apps for supermarkets and other large retail stores will use locational and user data to enhance the shopping experience, much like a personal shopping assistant in your pocket.

Google Indoor Maps<sup>156</sup> allows users to view and navigate floor plans of several kinds of commercial locations such as airports, department stores or malls, all within Google Maps.

This technology not only knows your indoor location, but also what floor you're on in a building. Depending on what data is available, the map shows notable places in the building you're currently in, such as stores, restrooms or the nearest food court.

With this type of technology, "you are here" directory maps will no longer be needed in malls or department stores. You will be able to determine your location and orient yourself using a smartphone, and this experience will adapt to your specific needs. For example, apps will offer you relevant discounts as you walk through the mall or highlight shops based on your gender and age.

---

<sup>155</sup>. <http://www.aislephone.com/>

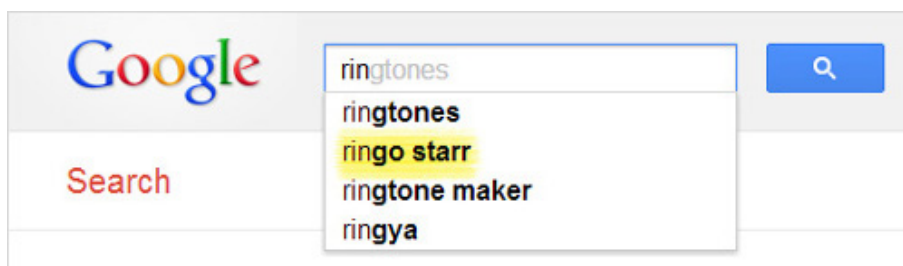
<sup>156</sup>. <http://www.google.com/mobile/maps/?hl=en>



## Designing An Adaptive System

Adaptive design integrates both subtle and obvious features. Often, adaptive qualities can be very subtle and unobtrusive: sometimes a seemingly small adaptive feature can greatly improve the overall experience. For example, did you ever notice that Google Search can read your mind? When you start typing, Google Instant<sup>157</sup>, using autocomplete<sup>158</sup>, knows what you're thinking even when you enter only three letters of a search term. It does this because Google Search considers and records all search queries within a session in order to have a better understanding of the user's intent.

When a user searches for "The Beatles," Google understands this as part of a research session and will help you quickly discover Ringo Starr or Paul McCartney as you enter the first three letters of their name; it understands the context of your search and compares it with other similar popular relevant results.



*Google Instant understands the context of your search.*

<sup>157</sup>. <http://www.google.com/insidesearch/features/instant/about.html>

<sup>158</sup>. <http://en.wikipedia.org/wiki/Autocomplete>

Another example of a subtle feature that helps enhance the user experience is a testing system for students that adjusts the difficulty of test questions according to whether prior questions were answered correctly. Or a music discovery application that looks into your current play list and adapts to your taste, helping you discover additional music you may like.

Although the experience should always be unobtrusive, adaptive interfaces need to be obvious so users understand the context for the adaptation and always feel in control. For a better experience, applications should also allow users to manage adaptive features. For example, if at nighttime the interface changes to a darker night mode (like in navigational devices), the user should always be able to change it back manually. Or, if entering a shopping mall triggers a different experience, the user must understand the context for this adaptivity and want to embrace the added functionality.

Charles Darwin wrote:

*“It is not the strongest of the species that survives, nor the most intelligent that survives. It is the one that is the most adaptable to change.”*

As human beings, we adapt to our surroundings naturally; it is the key to our survival. As designers, we can use this inherent ability and our physical senses and the powers of the brain to analyze and design what we would do in adaptable situations. For example, to communicate in a loud environment, we adapt by raising up our voice up to be heard. Similarly, an adaptive system will raise a device’s volume.

In an even louder environment, we use hand gestures to get attention and focus our eyes on the other person’s mouth to try to read their lips. However, unlike computers that can process multiple layers of data, human beings have limited sensory resources and a limited cognitive workload<sup>159</sup>.

In today’s world, a person carries in one pocket more advanced technologies than ever before possible. An intelligent device like a smartphone is embedded with highly sophisticated sensors. These sensors, together with advanced computing power and network connectivity, can help us analyze and understand the context of use. The smart device’s ability to analyze the context of use in real time, together with understanding the user’s story, allows opportunities to provide an even greater user experience by adapting to the user needs.

I will illustrate some of the key points in using these technologies.

---

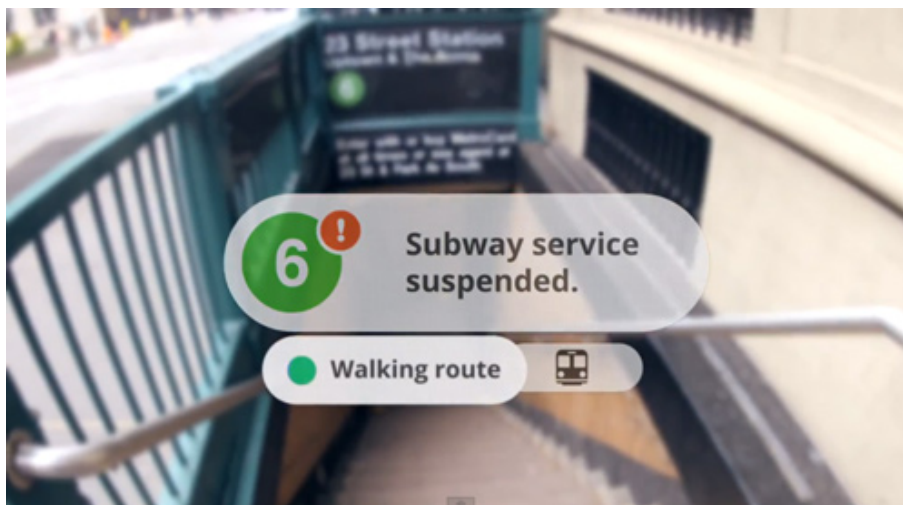
<sup>159</sup>. <http://en.wikipedia.org/wiki/Workload>



## ANALYZING USER BEHAVIOR

Similar to the Google Now example, analyzing user behavior and the user's interaction with the digital world can yield a great understanding of the user's context. Analyzing the user's search patterns or what applications they download can tell us about their preferences and hobbies. Tracking current location and location history can give us the user's surroundings and the physical boundaries of their life, so we can understand what subway station they take to work or where they like to eat their lunch. Note that when this is done without the knowledge of users, it may be considered a breach of browser security and illegal in many countries.

Here is a practical example of how analyzing the user's behavior could help in creating an adaptive system. In the now famous Google Glasses video<sup>160</sup>, we follow the user throughout his morning as he eats his breakfast and then leaves his house heading for the subway. Upon arriving at the subway, he receives a message that subway service is suspended and is offered a walking route. As useful as this may be, a *true* adaptive system will analyze the user behavior as the user gets up and will warn the user ahead of time that the subway service is suspended.



*Google Glasses uses information about the user's location to provide relevant information.*

Understanding the user's behavior (whether he takes a subway or walks to work) and connecting it with available information online allows us to understand and adapt to his needs. Most times, using one data source is not enough; combining the technologies (network con-

---

<sup>160</sup>. <http://www.youtube.com/watch?v=9c6W4CCU9M4>

nectivity, user behavior and sensor data) is the only way to understand context. For example, we can gauge the outside temperature combining the user's current location with online weather information, and then use this data to offer phone numbers for nearby cab companies in addition to a walking route, assuming the user may not wish to walk to work in the rain.

## MAKING USE OF THE USER'S STORY

Behavioral targeting<sup>161</sup> or personalization refers to a range of technologies used by online website publishers and advertisers that allows them to increase the effectiveness of their campaigns by capturing data generated from website and landing page visitors and then adapting to their needs. Personalization<sup>162</sup> technology enables the dynamic insertion, customization or suggestion of content in any format that is relevant to the individual user, based both on the user's explicitly provided details and their implicit behavior and preferences.

Another aspect of personalization is the increasing prevalence of open data on the Web. Many companies make their data available on the Web via APIs<sup>163</sup>, Web services and open data standards. For example, Pipl<sup>164</sup> is a search engine designed to locate people's information across the web. Pipl uses identity resolution<sup>165</sup> algorithms to aggregate information and cross-link various sources before delivering an online profile containing a summary report of everything that's publicly available for each individual. Pipl offers all that wealth of information<sup>166</sup> to developers via an API. One useful application for this would be running an API request for an email address; one can determine the user's gender, age, location and interests and provide an adaptive experience based on the individual user.

---

<sup>161</sup>. [http://en.wikipedia.org/wiki/Behavioral\\_targeting](http://en.wikipedia.org/wiki/Behavioral_targeting)

<sup>162</sup>. <http://www.quora.com/What-is-the-definition-of-personalization>

<sup>163</sup>. [http://en.wikipedia.org/wiki/Application\\_programming\\_interface](http://en.wikipedia.org/wiki/Application_programming_interface)

<sup>164</sup>. <https://pipl.com/>

<sup>165</sup>. <https://pipl.com/help/identity-resolution/>

<sup>166</sup>. <http://dev.pipl.com/>





*Pipl Search aggregates information that's publicly available on any individual.*

Understanding the user story is possible with a network connection. However, network connectivity is not only important to understand the user and his online record, it is a vital instrument that connects all other technologies together – cloud computing<sup>167</sup>, understanding local weather, traffic conditions or even the type of connection itself (Wi-Fi<sup>168</sup> or G3<sup>169</sup>) can help us understand context. Ultimately, the possibilities inherent in understanding and designing to the user's story – their context – are possibilities built upon the collection of sensor data and user data via the network.

## SENSOR DATA

A *sensor* for adaptive systems is any technology that allows a device to understand and evaluate context. It includes a built-in accelerometer in smart devices, a camera, a clock or even a microphone. We can use the various sensors embedded in smart devices to better understand the user's environment. For example, the built-in accelerometer can be used to gauge if a user is walking or running.

There are two main scenarios for using sensors: everyday objects transmitting data like temperature or noise level to other devices, for example, *iGrill*<sup>170</sup>, a cooking thermometer and application that communicates with smart devices via a secure, long-range Bluetooth connection. Or smart device applications, utilizing the built-in sensors to receive, process and output data to the user. By using these sensors and mixing other technologies discussed above, we can often obtain powerful information on the context of use and use it to create adaptive systems.

---

<sup>167</sup>. [http://en.wikipedia.org/wiki/Cloud\\_computing](http://en.wikipedia.org/wiki/Cloud_computing)

<sup>168</sup>. <http://en.wikipedia.org/wiki/Wi-Fi>

<sup>169</sup>. <http://en.wikipedia.org/wiki/3G>

<sup>170</sup>. <http://www.igrillinc.com/>



*iGrill Cooking Thermometer.*

Sensors can be a powerful design tool of the future. For example, with the aid of sensors, e-commerce checkout will be as easy as logging into a bank account with no password. Here is an example of how using four layers of sensor data to secure the user's identity with a degree of certainty to create passwordless banking that would present the user a "light" version of his bank account, so he quickly could check his account balance. Imagine a user is at home surfing to his bank account through his tablet computer.

The first layer of security is the username associated with the tablet. Second is the location sensor, which will give us a greater degree of certainty that the user is in his home vicinity, cross-checked with his registered address with the bank. The third layer is the Wi-Fi connection (its MAC address<sup>171</sup>, a unique identifier assigned to a network) the user is surfing on. For the fourth layer, we can check for other nearby Wi-Fi connections (The neighbors are sure to have a unique Wi-Fi MAC address) that can also be used as a security verification. If these bits of data are consistent across several password logins, the system can adapt and allow the user to enter without any password.

To learn more about adaptive design and how to get from sensors to context, I highly recommended you read this paper by Albrecht Schmidt about Context-Aware Computing<sup>172</sup> (Interaction Design Foundation Encyclopedia).

## Conclusion

Today, we're just starting to see the potential of using sensors and technology to connect between devices and people. The term "Internet of

<sup>171</sup>. [http://en.wikipedia.org/wiki/MAC\\_address](http://en.wikipedia.org/wiki/MAC_address)

<sup>172</sup>. [http://www.interaction-design.org/encyclopedia/context-aware\\_computing.html](http://www.interaction-design.org/encyclopedia/context-aware_computing.html)

things<sup>173</sup>” refers to uniquely identifiable objects that are network-connected. For example, a smart flowerpot that sends a signal when it’s time to water the flowers. There is no doubt that adaptive design will play a key role in making future devices and functional user interfaces that give users an intuitive control over their environments in any situation or context. 🐼

---

<sup>173</sup>. <http://jenson.org/of-bears-bats-and-bees-making-sense-of-the-internet-of-things/>

# About The Authors

## *Anastasios Karafillis*

Anastasios is a freelance user interface designer based in Greece. Having studied Social Sciences and Philosophy, he believes that a proper philosophy of language is crucial to any well-designed social or digital information system. My main areas of expertise are: Interaction Design, Information Architecture, Empathy, Usability, Nomenclature. Website: <http://anastasiosk.deviantart.com/><sup>174</sup>.

## *Avi Itzkovitch*

Avi (@xgmedia<sup>175</sup>) is a longtime interactive and Web design professional with over thirteen years of experience. Author of *Design-It-Yourself: Web Sites* (Rockport Publishers, 2003), Avi has published several articles about user experience, he is an avid enthusiast of technology and is often invited to speak about emerging design and UX trends. Avi is currently working as an Independent UI/UX Consultant and is the owner of XG Media.

## *Bill Beard*

Bill Beard is a freelance copywriter, creative director and content strategist who connects companies with customers at every touchpoint. He's worked with startups like *TheLadders*<sup>176</sup> and agencies like *Digitas* and *mcgarrybowen*. He's a believer in Lean UX and recently spoke at *LeanUXNYC*<sup>177</sup>. Bill loves building brands and creating great experiences, but hates talking about himself in the third person. You should follow him on *Twitter*<sup>178</sup>.

## *Lennart Hennigs*

Lennart Hennigs is a Senior Interaction Designer at Deutsche Telekom, where he designs for mobile and fixed-line user experiences. He has

---

<sup>174</sup>. <http://anastasiosk.deviantart.com/>

<sup>175</sup>. <https://twitter.com/xgmedia>

<sup>176</sup>. <http://www.theladders.com/>

<sup>177</sup>. <http://leanuxnyc.co/nyc/>

<sup>178</sup>. <https://twitter.com/writebeard>

worked in the field for over ten years — beginning with classic usability engineering; he shifted his focus to conception and interaction design in 2003. Lennart studied Computer Science and Philosophy. He is active in the German Chapter of the UPA and when he is not busy commuting he writes about UX topics. He recently wrote an ebook about [mobile prototyping with Axure RP](#)<sup>179</sup>. Twitter: [@LennartHennigs](#)<sup>180</sup>.

## *Loren Baxter*

Loren leads the product design team at [ReadyForZero](#)<sup>181</sup>, an [award winning](#)<sup>182</sup> startup that helps people pay off debt and build wealth. He occasionally writes at [A Clean Design](#)<sup>183</sup>, mentors at [The Designer Fund](#)<sup>184</sup>, and accepted an Interaction Design Award at the Interaction '12 conference for his team's work. He can otherwise be found paddling around the ocean, off on adventures, or chasing a soccer ball. Twitter: [@lorenbaxter](#)<sup>185</sup>.

## *Senongo Akpem*

Senongo is a Nigerian-American designer and illustrator in New York City. He designs and builds digital experiences, and founded [Pixel Fable](#)<sup>186</sup>, an interactive archive of African fables. Follow him on Twitter [@senongo](#)<sup>187</sup>.

## *Yogev Ahuvia*

Yogev Ahuvia is a Front End Team Leader at [Netcraft Israel](#)<sup>188</sup>, an industry leading UX agency. You can follow him on [his website](#)<sup>189</sup>, his [Twitter](#)<sup>190</sup> or [Google+](#)<sup>191</sup> accounts, or see him painting with code on [CodePen](#)<sup>192</sup>.

---

<sup>179</sup>. <http://www.axureformobile.com/>

<sup>180</sup>. <http://www.twitter.com/LennartHennigs>

<sup>181</sup>. <https://www.readyforzero.com/>

<sup>182</sup>. <http://lifehacker.com/5951814/readyforzero-calculates-your-optimal-debt-reduction-plan-and-pays-your-bills-automatically>

<sup>183</sup>. <http://www.acleandesign.com/>

<sup>184</sup>. <http://www.designerfund.com/>

<sup>185</sup>. <http://www.twitter.com/lorenbaxter>

<sup>186</sup>. <http://pixel-fable.com/>

<sup>187</sup>. <http://www.twitter.com/senongo>

<sup>188</sup>. <http://www.netcraft.co.il/en/>

<sup>189</sup>. <http://www.yogevahuvia.com/>

<sup>190</sup>. <https://twitter.com/kindofone>

<sup>191</sup>. <http://gplus.to/yogevahuvia/>

<sup>192</sup>. <http://codepen.io/kindofone>

## *About Smashing Magazine*

Smashing Magazine<sup>193</sup> is an online magazine dedicated to Web designers and developers worldwide. Its rigorous quality control and thorough editorial work has gathered a devoted community exceeding half a million subscribers, followers and fans. Each and every published article is carefully prepared, edited, reviewed and curated according to the high quality standards set in Smashing Magazine's own publishing policy<sup>194</sup>.

Smashing Magazine publishes articles on a daily basis with topics ranging from business, visual design, typography, front-end as well as back-end development, all the way to usability and user experience design. The magazine is — and always has been — a professional and independent online publication neither controlled nor influenced by any third parties, delivering content in the best interest of its readers. These guidelines are continually revised and updated to assure that the quality of the published content is never compromised. Since its emergence back in 2006 Smashing Magazine has proven to be a trustworthy online source.

---

<sup>193</sup>. <http://www.smashingmagazine.com>

<sup>194</sup>. <http://www.smashingmagazine.com/publishing-policy/>