

Practical Approaches for

Designing Usable Websites

A large, stylized, maroon-colored letter 'U' is positioned in the bottom right corner of the cover. The background features a network of white lines connecting circular nodes, with several overlapping light pink circles.

Imprint

Copyright 2012 Smashing Media GmbH, Freiburg, Germany

Version 1: May 2012

ISBN: 978-3-943075-28-1

Cover Design: Ricardo Gimenes

PR & Press: Stephan Poppe

eBook Strategy: Thomas Burkert

Technical Editing: Andrew Rogerson, Talita Telma

Idea & Concept: Smashing Media GmbH

ABOUT SMASHING MAGAZINE

[Smashing Magazine](#) 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. 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.

ABOUT SMASHING MEDIA GMBH

[Smashing Media GmbH](#) is one of the world's leading online publishing companies in the field of Web design. Founded in 2009 by Sven Lennartz and Vitaly Friedman, the company's headquarters is situated in southern Germany, in the sunny city of Freiburg im Breisgau. Smashing Media's lead publication, Smashing Magazine, has gained worldwide attention since its emergence back in 2006, and is supported by the vast, global Smashing community and readership. Smashing Magazine had proven to be a trustworthy online source containing high quality articles on progressive design and coding techniques as well as recent developments in the Web design industry.

About this eBook

The User Center Design process is based on various steps, and for each of these steps there are appropriate methods. These methods can help improve the usability (and usefulness) of your website. The Smashing eBook **Practical Approaches For Designing Usable Websites** provides you several techniques that will help make your Web applications appeal to the masses, transforming them into lightweight user experiences.

Table of Contents

[Evolve Your User Interface To Educate Your Users](#)

[Optimizing Emotional Engagement In Web Design Through Metrics](#)

[Enhancing User Interaction With First Person User Interface](#)

[Minimizing Complexity In User Interfaces](#)

[A Guide To Heuristic Website Reviews](#)

[Stop Designing Pages And Start Designing Flows](#)

[The Data-Pixel Approach To Improving User Experience](#)

[About The Authors](#)

Evolve Your User Interface To Educate Your Users

Dave Shepard

The Web has changed. This isn't your neighbor's nerdy kid's internet anymore. Now the Web is home to your mom, your grandma and your technophobe sister. With computers as common a household appliance as televisions now, who might be using your web-application has expanded beyond the realm of *just* the power user.

Complicated menu systems, alert dialog messages that lock you out of the browser and flashy but confusing layouts aren't necessarily going to help you make conversions. The Web user demographic has changed and to make your web application appeal to the masses your user interface needs to teach and to guide.

User Interfaces Should Teach

User interfaces need to teach your user *how to use your application without resorting to a help screen*. Many developers think that a help section will suffice for teaching users how to operate most any application, but this is not the case.

The help section of most applications, Web-based or otherwise, end up being used by the power users who are already trying to figure out every nook and cranny of the application. The proper way to help users understand how your application works is to bring the help section to them and in a format that can easily be understood.

In-line tips and interface descriptions work as an excellent introduction to the interface and how it operates. By providing your user with assistive instruction aside the interaction, it is easier for the user to relate the instruction with how it applies to your web application's functionalities than a disconnected FAQ or help section.

[Nourish](#) by *Integral Impressions* is an application that creates an automated email newsletter of any RSS feed. The entire interface is documented in-line with descriptive help blocks that can easily be closed once the user decides they don't need the help anymore. A quick visit to the user profile section presents an option to re-enable all the help boxes in the account just in-case the user forgot how to do something.

The screenshot displays the Nourish dashboard with a dark green header containing 'Campaigns', 'Subscribers', and 'Dashboard View'. A light green notification bar states 'Logged in successfully'. A large yellow help block contains the following text: 'Welcome to Nourish. Here you will find a list of all the campaigns you have created. Each campaign represents an email that readers can subscribe to. You'll probably want to create a separate campaign for each RSS feed you want to publish to email. Another idea is to create a couple campaigns for each feed, which broadcast at different intervals (weekly vs. monthly). **Preview Campaign** You can preview your campaign by clicking on the magnifying glass. This will show what the campaign would look like if it were to go out today. In other words, it may be empty depending on where you are in your delivery cycle (i.e. if a mailing just went out). **Time Zone** You can set your time zone so that times are displayed correctly. Just click the user profile link at the top of the screen. [Hide this box](#)'. Below this, a grey bar indicates 'Your account allows you to send up to 5000 emails per month. So far you have sent 0 emails.' The 'Campaigns' section features a table with columns: Name, Active Subscribers, Open Rate, and Click-Through Rate. A '+ New Campaign' button is in the top right. A message states 'You don't have any campaigns right now. You can create a new campaign [here](#).'

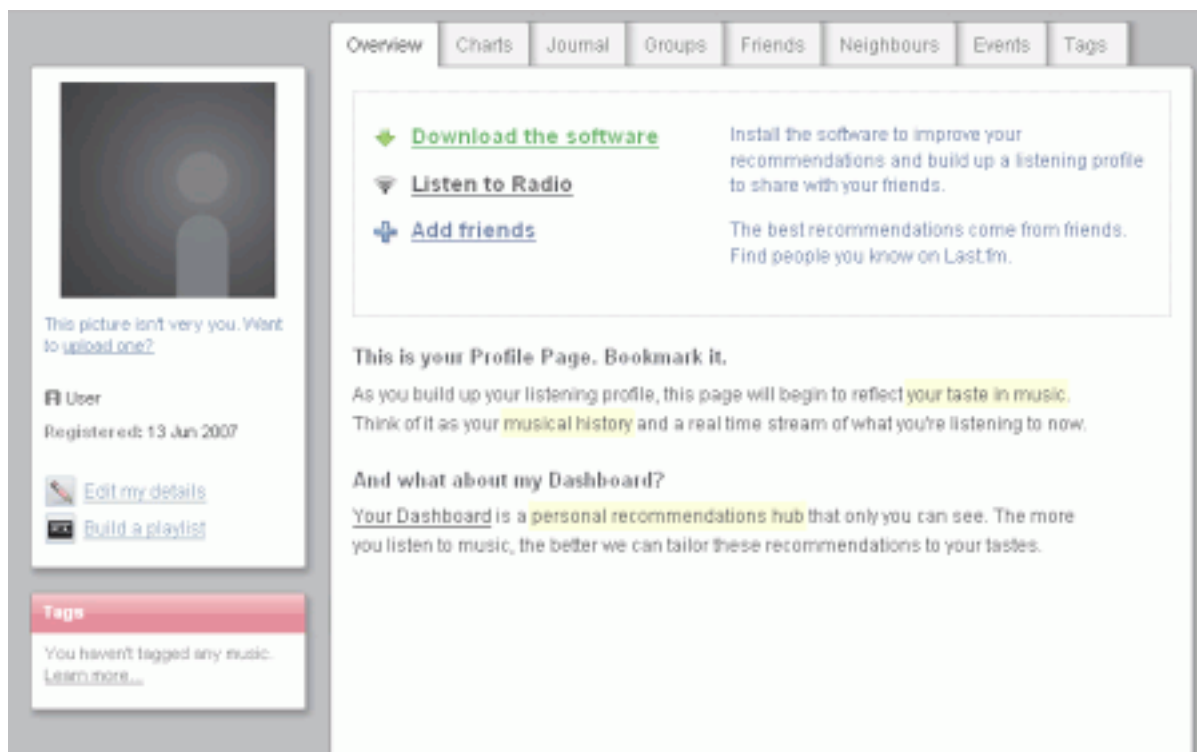
Nourish's interface is documented in-line with descriptive help blocks.

37Signals accomplishes helping their users with their communications management applications like [Basecamp](#) in a similar manner. Sections which do not have any content yet display a screenshot of the section, populated with data. This screenshot is even linked to an instructional video on how to create content for the section. That's user-friendly.



In Basecamp sections without any content display a screenshot of the section, populated with data.

[Last.fm](#) provides users with leads on what they can do with their social online radio application via contextual tips and descriptive defaults, teaching the user how use and understand the interaction without taking them away from it as a FAQ would do.



Last.fm uses contextual tips and descriptive defaults, teaching the user how use and understand the interaction.

To ease a user into your application, the application needs to teach the user *how to use it*. Once the user has moved beyond the beginner level with that application, the application should allow the user to skip all the help boxes and tool-tips or they should be un-obtrusive enough that they are never in the way to begin with. It is best to allow your users to re-enable the tips on how to use your application. Though never assume your user is done learning how to use your application the first time they read your helpful tips.

However, teaching a user how to use your interface is not just limited to providing tool-tips and lots of descriptive fields. You can easily teach a user by setting good default data. Setting good default data demonstrates to the user how a field should be formatted or what kind of information is expected in a particular text panel.

By entering in an email field *email@domain.com* as the default value it is perfectly clear what you are expecting to be entered in this field and how it is supposed to be formatted. To complete the interaction a bit of simple JavaScript can be used to make the form field even more usable by clearing the default value when the user gives focus to the field and then returning the default value if they did not enter any information.

This technique can even be taken into other fields where you may be expecting specific descriptive data to be entered, such as placing *Describe the details of this project...* as the default value in a textarea expecting the description of a project. Setting good defaults not only helps ensure that you'll get the data is entered correctly, but it helps teach the user how the pieces of your application function.

User Interfaces Should Guide

Setting good defaults in your user interface and providing helpful tips will also allow you to guide users through processes and prevent having to correct them. There is nothing worse for the user experience than filling out a form of non-descript fields, and then alerting, via JavaScript, back to the user about how they did not fill the form out properly. Utilizing debugging implementations for instructive communication is a very alienating way of communicating instruction to users. A better way of instructing a user is to guide them through your form and prevent errors from ever happening.

Yahoo places defaults in fields that are not self-descriptive, but these defaults disappear when the field is given focus so as not to interrupt the user's flow when filling out the form. When form fields are given focus, a simple description of the field and its parameters is given to the right, providing simple instruction without interrupting the user's flow through the form.

YAHOO! Yahoo! - Help

Hi there!
We'll get you set up on Yahoo! in three easy steps! Just answer a few simple questions, select an ID and password, and you'll be all set.

Already have an ID or Mail address?
[Sign in](#)
[Forgot your password or Yahoo! ID?](#)

I prefer content from Yahoo! U.S. in English

1. Tell us about yourself...

My Name Dave Shepard

Gender - Select One -

Birthday - Select Month - Day Year

I live in United States

Postal Code

2. Select an ID and password

Yahoo! ID and Email @ yahoo.com Check

Password Password Strength

Re-type Password

3. In case you forget your ID or password...

Alternate Email

Security Question - Select One -

Your Answer

Your postal code lets Yahoo! provide you with content that is relevant to where you live and help you reset your password if you ever forget it.

On Yahoo defaults in fields disappear when the fields are entering the focus state.

Microsoft's Live Mail signup also provides descriptive examples and parameters for fields that require them. Microsoft additionally groups fields in the signup form that are related to each other, providing a description of the section to the right of the form interaction area.

Create a Windows Live ID

The screenshot shows the 'Create a Windows Live ID' form. At the top, a green checkmark and text confirm that 'daveshep@live.com' is available. Below this, the email address is split into two input fields: 'daveshep' and 'live.com', separated by an '@' symbol. A 'Check availability' button is positioned below the email fields. The main section, titled 'Choose your password', contains a red error message: 'The new password and the confirmation password don't match. Please type the same password in both boxes.' This message is positioned above two password input fields. The first field is labeled '*Type password:' and the second is labeled '*Retype password:'. Below the first password field, a 'Password strength' indicator shows a green bar labeled 'Medium' with the text 'Six-character minimum; case sensitive' underneath. To the right of the password fields, a text block explains that strong passwords contain 7-16 characters and should include common words, names, and a combination of uppercase letters, lowercase letters, numbers, and symbols. A link 'Get help with this' is provided at the bottom of this text block.

Microsoft groups fields in the signup form that are related, providing a description of the section to the right of the form interaction area.

The form only displays the description for the section the user is currently interacting with to keep the form area from becoming cluttered, allowing the user to focus on the section they are in before moving to the next.

Squidoo's lens signup wizard is another example of a guiding interface. Their lens signup process provides human descriptions of each step making it easy to understand what is being done. The contact form seems to communicate with the users throughout their registration.

Step 2: What do you want to do with your lens?

A few real examples: There's Damon, a teacher who makes learning lenses with his students. And Jane Goodall, supporting her nonprofit organization. And Kimberly, an author promoting her books. And Raj, who wants traffic back to his blog. These are just 4 ways people use Squidoo. How will you?

- ☒ I want to get the word out about Design (easiest option)
- ☐ I'm gonna sell Design stuff and earn \$\$, for me or for charity (money maker!)
- ☐ I want to make a list of my favorite things about Design (most popular)
- ☐ I just want to do my own thing.

[Back](#)[Go to Step 3](#)

just 2 easy steps to go!

Squidoo also gives feedback to the user in their multi step process of how many steps they have to go providing the user an understanding of the scope of the signup process.

Setting up a form with these implementations humanizes the experience and hand-holds the user as they are walked through the form.

With AJAX and JavaScript DHTML, form fields can now be validated in real-time — no more submitting and going back is necessary. For instance, Yahoo's account signup form display's a green check mark if the data is correct, and highlights the invalid field with a friendly description of what was filled out incorrectly and why.

Sign-Up

New around here? Welcome!

Now then, Let's get you all signed up.

Email Address

dave@digital-telepathy.com

✔ Nicely put!

First Name

Dave

✔ So that's what they call you!

Last Name

Shepard

✔ Classy last name, I have to admit!

Username (letters & numbers only, please!)

⚠ Hmm, please be more creative...

Password (Must be more than 5 characters, without spaces)

⚠ So, that's not really a password...

Zoomr.com validates their signup form in real-time with personable, yet helpful descriptions of field parameters

[Zoomr](#), an online photo storage and sharing service, validates their signup form in real-time with personable, yet helpful descriptions of field parameters and valid (or invalid) entered data.

Conclusion

The average web user demographic has changed and so must the user interface. By creating a more usable, easier to understand and less intimidating experience your web application will encourage people to use it. Designing a web interface isn't about how many features you can stick in one navigation or how pretty that subtle gradient might be.

Designing a web interface is about creating a conduit for users which is easy to use and understand, helps to to accomplish tasks and improves their experience. Show the users how you would like them to use your application, but don't force it on them. Allow your web application to guide your users through the process, teaching them how to become experts at it and make the application work for *them*.

Optimizing Emotional Engagement In Web Design Through Metrics

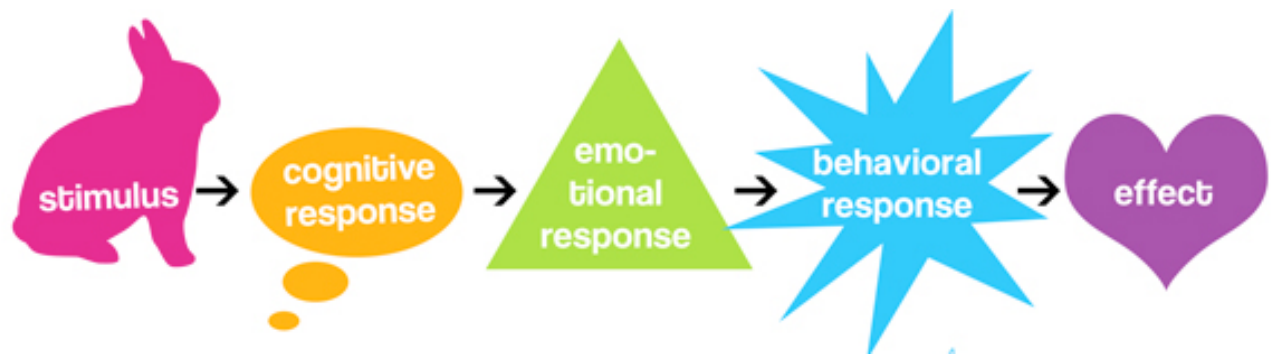
Shell Greenier

Think about what keeps you coming back to your favorite store, your favorite person or even your favorite website. It's not just a mindless buy-go, hug-go or click-go relationship. It is a complicated, emotional connection. It is what makes relationships with people and brands intoxicating. User engagement must have an equally complex emotional connection. It must affect the user in mind, body and spirit. Anything less is a 1990s brochure website.

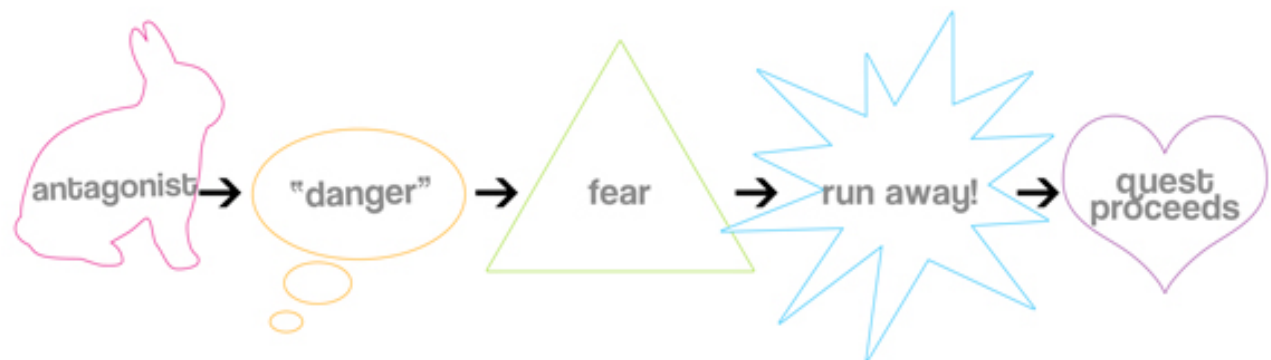
You can create strong storytelling strategies based on user personalities and segmentation. However, it seems almost impossible to measure those efforts, let alone know how to optimize them, without access to a [neuroscience laboratory](#). In fact, emotional engagement can be optimized, and quite effectively, using something already at your disposal: [performance metrics](#).

Emotional-Behavioral Response Relationship

Let's start with the basics: an emotion is a psychophysiological response in your body to a stimulus. It's an internal process that in turn triggers an external behavioral response. Behavioral responses help you decipher the emotional responses of others. Things like facial expressions and body language give you clues to whether the chef wielding the knife is angry and going to attack you or happy and going to make you dinner.



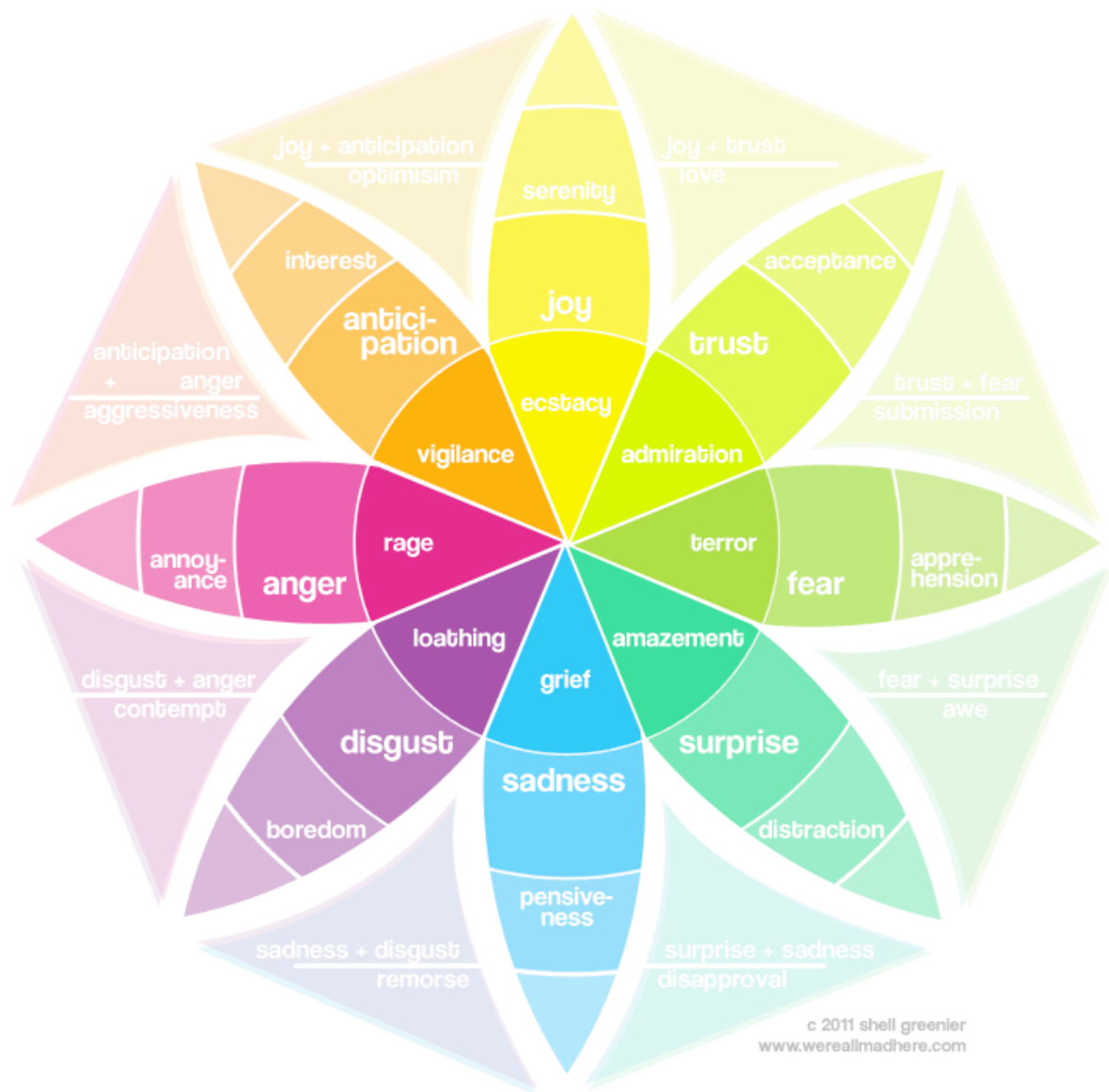
For example:



But you don't have to be face to face in order to read a person's behavioral clues. In digital environments, users' behavioral interactions with the platform can give you insight into their emotional states. Instead of reading facial cues to analyze how your UX efforts affect users, you can measure their responses via metric data. Metric data is a user's behavioral response quantified. With a little reverse engineering, you can identify strong emotional triggers, optimize weak elements and create stronger user experience strategies, using psychology as your secret weapon.

By The Numbers: Behavioral Response

Behavioral psychologists have classified emotions in numerous different theories. A large majority of these theories agree that emotions manifest in various intensities and can even combine with others to build new emotional states. One example of such a theory is Robert Plutchik's emotion wheel.



Plutchik's emotion wheel.

When it comes to user experience, emotional engagement builds on itself as the user continues to interact not only with your platform, but with all aspects of your brand online, including SEM, press coverage and social networks. Emotional engagement with a digital product can be divided into four categories based on how much information and engagement the user has with your website: awareness, attraction, investment and adoption.

AWARENESS

Emotion Map of a Coffee Addict



Emotion Maps: Legend



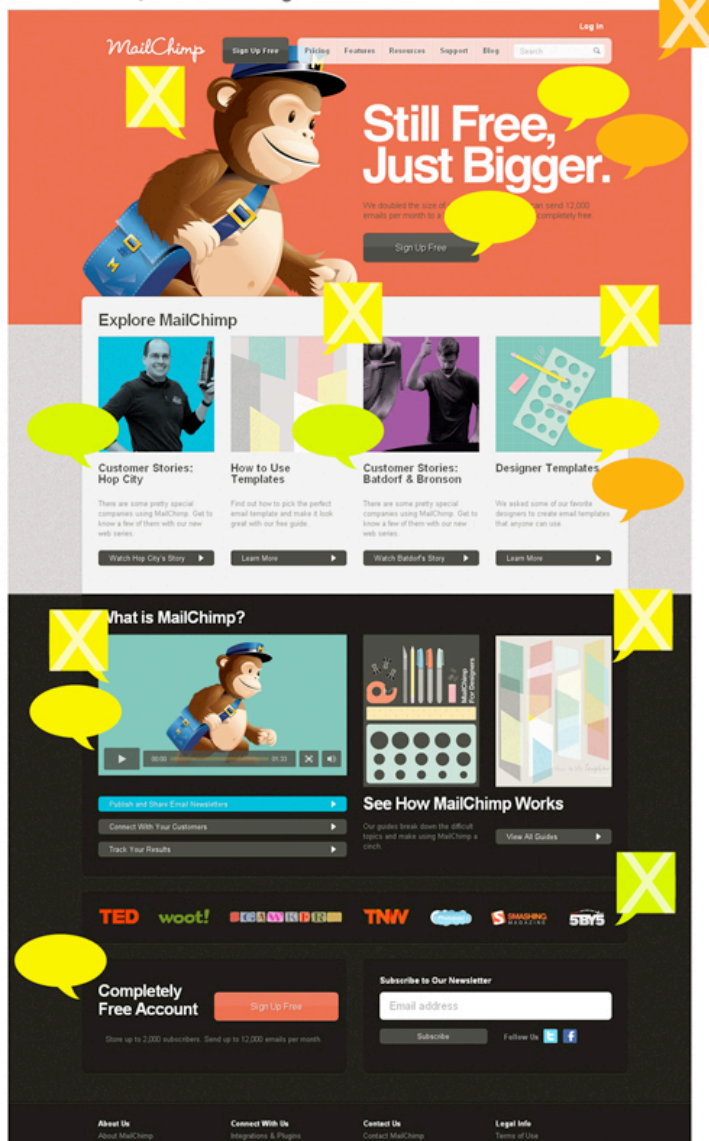
User experience doesn't start when they hit your landing page or start your app, but prior to it. Visitors have to make their way to your website in the first place. They're navigating through websites full of frustration-inducing elements, trust-busting perils and anticipation-inducing amusements. The focus at this point is on building trust, anticipation and whatever other emotional responses you'll target in your copy, imagery and overall [storytelling](#).

- Track awareness-level engagement using metrics like page views, page hits, video views, impressions and click-through rates.
- Identify emotional image and copy triggers during the development phase by A/B testing on micro-sites, as well as using ad content and email campaigns.
- Create dynamic content that spotlights your SEM, SEO and advertising goals, giving users exactly what they want as soon as they hit your page.
- Leverage strong referral sources and advertising platforms to build trust and credibility. The coffee vendors featured in the screenshot above use Google, and the companies featured to the right of this column you're reading use Smashing Magazine to reach customers. By aligning their messages with these brands, they are able to build more trust in the user than an ad on a less relevant platform (like Craigslist) might have.
- Use emotionally rich imagery in your advertising messages, and carry that messaging through to the website itself. "Fresh-roasted coffee" begins to paint an emotional picture for the user of that perfectly roasted cup of steaming delight first thing in the morning. If your click-through rate is low, then your ad may not be compelling enough.

- Develop intuitive and relevant architecture to decrease frustration and increase trust. First impressions count. If the bounce rate is high or the time on site is low, then the story you are telling in the awareness phase might not be carrying through to the user's interaction on the platform.

ATTRACTION

Emotion Map: The Monkey



Emotion Maps: Legend



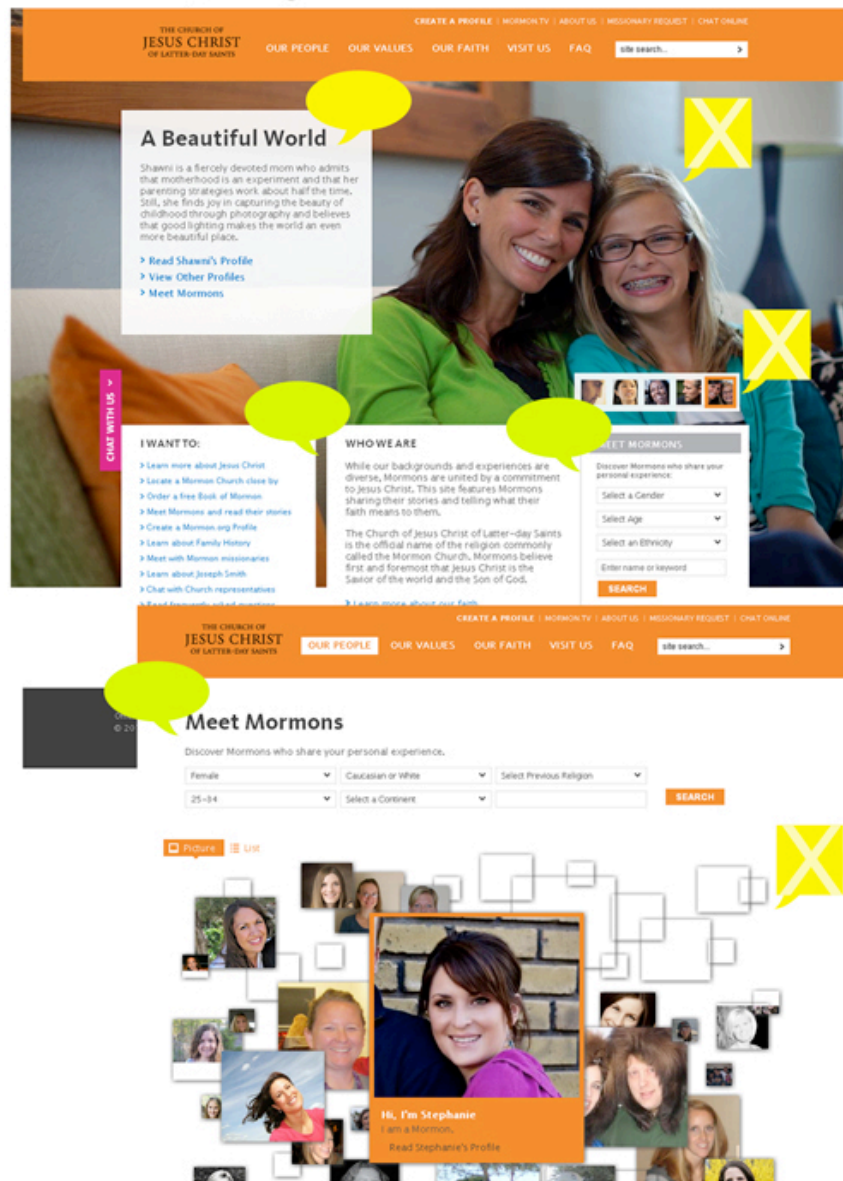
Attraction-level engagement keeps users interested in your platform. Cohesion of the UX elements and usability is the name of the game. Building emotional engagement at this level is critical because behavioral engagement consists mainly of superficial interaction such as navigation and content absorption.

- Track attraction-level engagement by looking at bounce rates, session lengths, pages per visit, abandonment rates, email opening rates and click-through rates.
- Stay up to date on current trends in usability to create systems that are user-centric.
- Identify high-focus areas in your user interface through eye-tracking, heat maps and software that records mouse movement. Optimize your framework, and place emotion-building content in strategic areas.
- Continue building and reinforcing motivational and emotional triggers through engaging imagery, emotionally charged words for headings and main copy, and persuasive triggers. [@Mailchimp](#) publicizes a lot of “free”, “big” offers and uses bright imagery to boost joy and anticipation levels. Negative emotions, like sadness, can also build strong emotional engagement in users. Examples are the powerful images of animal cruelty often seen in PETA and ASPCA campaigns.
- Highlight brand relationships, security measures, press and endorsements. Aligning your product with other respected brands builds trust. The list of brands that Mailchimp uses to show its press features builds an impressive amount of credibility. This is immediately followed by the call to subscribe by email. Putting the call to action after the trust-building credentials is more effective than the other way around.

- Develop proper system-generated feedback and error handling. This could include loading notices and 404 pages with content referrals, which reduce abandonment due to user frustration.

INVESTMENT

Emotion Map: Religious Site



Emotion Maps: Legend



Investment-level engagement involves a commitment from the user. It moves beyond a simple navigational interaction; the user is no longer behaving based on curiosity and anticipation alone. The user is interested in what you are offering, they trust your credibility, and they anticipate further interaction; thus, they act appropriately. They are now engaged enough to invest time, or do something risky like download a file or submit credit card information, or assume an identity (real or fictitious), stepping out of the role of anonymous Web surfer.

- Track investment-level engagement by tracking your social network followers, RSS feed or podcast subscribers, email newsletter subscriptions, file downloads, e-commerce conversion rates, purchase line items (both items and amount), community sign-ups, and warm leads.
- Leverage your existing community to motivate others to action. @Mormon.org leverages its community to build trust, and its strong messaging of love and belonging and its interesting interface build joy in users.
- Minimize frustration by requiring the fewest steps possible to achieve the objective. For example, allow users to order without logging in. The more information you require (such as Social Security or credit card numbers), the more trust you will have to build.
- Reiterate trust, security and credibility elements during the check-out or registration process.
- Monitor your online reputation.

ADOPTION

Emotion Map: Fan Building



Emotion Maps: Legend



Adoption entails users accepting the website structure as a common platform for interaction or knowledge on a subject. Emotional engagement is extremely high, and interaction is consistent. If investment-level engagement gets users involved, then adoption-level engagement makes them your cheerleaders.

- Adoption-level engagement can be seen in return customers, unique versus returning visitor ratios, geo-location check-ins, and participation in “karma” systems (badges, etc.), to name a few.
- Use [strong social interfaces](#) to spotlight followers, supporters and die-hard addicts.
- Put community-building messaging in your copy. “Colbert Nation,” “Psychos,” “Gleeks” and “Chipotle for Life” are elements that brands use to add excitement and enthusiasm to their products and reinforce the emotional engagement of their communities.
- Make it easy for users to integrate the product into their daily lives, as Foursquare, Facebook and Twitter do via apps and open APIs.
- Invite fans to help shape your future, and make it easy for them to share information, spread your brand and recruit their network.
- Let users play. It’s just a burrito wrapped in foil, but [@Chipotle](#) builds an addictive community element into its website by combining joy, trust and anticipation with quick, simple interactive elements.

Summary

So, what keeps people coming back? Or more to the point, what keeps them from coming back? Using metric data, we are able to trace behavioral and emotional responses to identify the weak spots in our storytelling strategies. By identifying and optimizing these areas, we're able to make our products better, faster, stronger.

These are just some of the ways to quantify emotional engagement strategies. How do you measure your engagement successes?

Enhancing User Interaction With First Person User Interface

Luke Wroblewski

Though many computer applications and operating systems make use of real-world metaphors like the desktop, most software interface design has little to do with how we actually experience the real world. In lots of cases, there are great reasons not to directly mimic reality. Not doing so allows us to create interfaces that enable people to be more productive, communicate in new ways, or manage an increasing amount of information. In other words, to do things we can't otherwise do in real life.

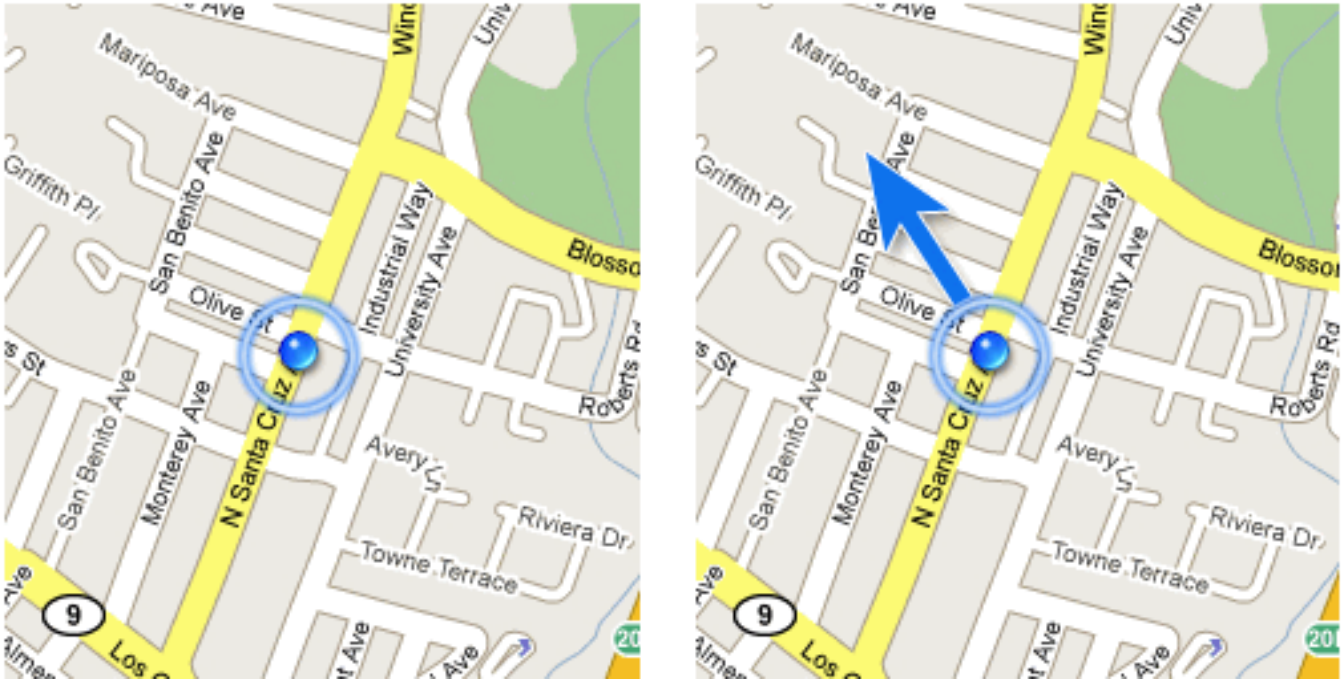
But sometimes, it makes sense to think of the real world as an interface. To design user interactions that make use of how people actually see the world to take advantage of first person user interfaces.

First person user interfaces can be a good fit for applications that allow people to navigate the real world, “augment” their immediate surroundings with relevant information, and interact with objects or people directly around them.

Navigating the Real World

The widespread integration of location detection technologies (like GPS and cell tower triangulation) has made mobile applications that know where you are commonplace. Location-aware applications can help you find nearby friends or discover someplace good to eat by pinpointing you on a map.

When coupled with a digital compass (or a similar technology) that can detect your orientation, things get even more interesting. With access to location and orientation, software applications not only know where you are but where you are facing as well.



This may seem like a small detail but it opens up a set of new interface possibilities that are designed from your current perspective. Consider the difference between the two screens from the TomTom navigation system shown below. The screen on the left provides a two-dimensional, overhead view of a driver's current position and route. The screen on the right provides the same information but from a first person perspective.



This first person user interface mirrors your perspective of the world, which hopefully allows you to more easily follow a route. When people are in motion, first person interfaces can help them orient quickly and stay on track without having to translate two-dimensional information to the real world.



TomTom's latest software version goes even further toward mirroring our perspective of the world by using colors and graphics that more accurately match real surroundings. But why re-draw the world when you can provide navigation information directly on it?

[Nearest Tube](#) is a first person navigation application that tells you where the closet subway station is by displaying navigation markers on the real world as seen through your phone's camera.

As you can see in the video above, the application places pointers to each subway station in your field of vision so you can quickly determine which direction to go. It's important to note, however, that the application actually provides different information depending on your orientation.

When you hold the phone flat and look down, a set of arrows directs you to each subway line. Holding the phone in front of you shows the nearest

subway stations and how far away they are. Tilting the phone upwards shows stations further away from you.

LOOKING DOWN



LOOKING AHEAD

LOOKING UP

Making use of the multiple perspectives naturally available to you (looking down, looking ahead, looking up) is an example of how first person interfaces allow us to interact with software in a way that is guessable, physical, and realistic. Another approach (used in [Google Street View](#)) is to turn real world elements into interface elements.

Street View enables people to navigate the World using actual photographs of many major towns & cities. Previously, moving through these images was only possible by clicking on forward and back arrows overlaid on top of the photos. Now, (as you can see in the demo video below) Street View allows you to use the real-world images themselves to navigate around. Just place a cursor on the actual building or point on the map that you want to view and double-click.

Augmented Reality

Not only can first person user interfaces help us move through the world, they can also help us understand it. The information that applications like Nearest Tube overlay on the World can be thought of as "augmenting" our view of reality. Augmented reality applications are a popular form of first person interfaces that enhance the real world with information not visible to the naked eye. These applications present user interface elements on top of images of the real world using a camera or heads up display.



For example, an application could augment the real world with information such as ratings and reviews or images of food for restaurants in our field of vision. In fact, lots of different kinds of information can be presented from a first person perspective in a way that enhances reality.

[IBM's Seer](#) application provides a way to navigate this year's Wimbledon tennis tournament more effectively. Not only does the application include navigation tools but it also augments your field of vision with useful information like the waiting times at taxi and concession stands.



[Wikitude](#) is an application that displays landmarks, points of interest, and historic information wherever you point your phone's camera. This not only provides rich context about your surroundings, it also helps you discover new places and history.



These augmented reality applications share a number of design features. Both IBM Seer and Wikitude include a small indicator (in the bottom right corner of the screen) that lets you know what direction you are facing and how many points of interest are located near you. This overview gives you a quick sense of what information is available. Ideally, the data in this overview can be manipulated to zoom further out or closer in, adjust search filters, and even select specific elements.

Wikitude allows you to manage the size of this overview radius through a zoom in/out control on the left side of the screen. This allows you to focus on points of interest near you or look further out. Since it is dealing with a much smaller area (the Wimbledon grounds), IBM Seer doesn't include (or need) this feature.

In both applications, the primary method for selecting information of interest is by clicking on the icons overlaid on the camera's view port. In the case of IBM Seer, different icons indicate different kinds of information like concessions or restrooms. In Wikitude, all the icons are the same and indicate information of interest and distance from you. Selecting any of these icons brings up a preview of the information. In most augmented reality applications, a further information window or screen is necessary to access more details than the camera view can display.

When many different types of information can be used to augment reality within a single application, it's a good idea to allow people to select what kinds of information they want visible. Otherwise, too many information points can quickly overwhelm the interface.

[Layar](#) is an augmented reality application that allows you to select what kinds of information should be displayed within your field of vision at any time. The application currently allows you to see houses for sale and rent, local business information, available jobs, ATM locations, health care providers, and more. As the video below highlights, you can switch between layers that display these information points by clicking on the arrows on the right and left sides of the screen.

Layar also provides quick access to search filters that allow you to change the criteria for what shows up on screen. This helps narrow down what is showing up in front of you.

Interacting with Things Near You

First person user interfaces aren't limited to helping you navigate or better understand the physical space around you they can also enable you to interact directly with the people and objects residing within that space. In most cases, the prerequisite for these kinds of interactions is identifying what (or who) is near you. As a result, most of the early applications in this category are focused on getting that part of things right first.

One way to identify objects near you is to explicitly provide information about them to an application. An application like [SnapTell](#) can identify popular products like DVDs, CDs, video games, and books when you take a picture of the product or its barcode. The application can then return prices, reviews, and more to you.



This approach might eventually also be used to identify people as illustrated in the [augmented ID concept application from TAT](#) in the video below. This

proposed application uses facial recognition to identify people near you and provides access to digital information about them like their social networking profiles and updates.

While taking pictures of objects or people to get information about them is a more direct interaction with the real world than typing a keyword into a search engine, it only partially takes advantage of first person perspective. Perhaps it's better to use the objects themselves as the interface.

For example, if any of the objects near you can transmit information using technologies like RFID tags, an application can simply listen to how these objects identify themselves and act accordingly. Compare the process of inputting a barcode or picture of an object to the one illustrated in this video from the [Touch research project](#). Simply move your device near an RFID tagged object and the application can provide the right information or actions for that object to you.

This form of first person interface enables physical and realistic interactions with objects. Taking this idea even further, information can be displayed on the objects themselves instead of on a device. The [6th Sense project](#) from the MIT Media Lab does just that by using a wearable projector to display product information on the actual products you find in a library or store.

Though some of these first person interfaces are forward-looking, most are available now and ready to help people navigate the real world, “augment” their immediate surroundings, and interact with objects or people directly around them. What's going to come next is likely to be even more exciting.

The next time you are working on a software application, consider if a first person user interface could help provide a more natural experience for your users by bringing the real world and virtual world closer together.

Minimizing Complexity In User Interfaces

Tyler Tate

Clean. Easy to use. User-friendly. Intuitive. This mantra is proclaimed by many but often gets lost in translation. The culprit: complexity. How one deals with complexity can make or break an application. A complex interface can disorient the user in a mild case and completely alienate them in an extreme case. But if you take measures first to reduce actual complexity and then to minimize perceived complexity, the user will be rewarded with a gratifying experience.

We recently faced this very challenge while designing two Web-based applications, including an enterprise-level content management system. What follows are several techniques that have helped us streamline complex applications into lightweight user experiences.

Grab The Hatchet

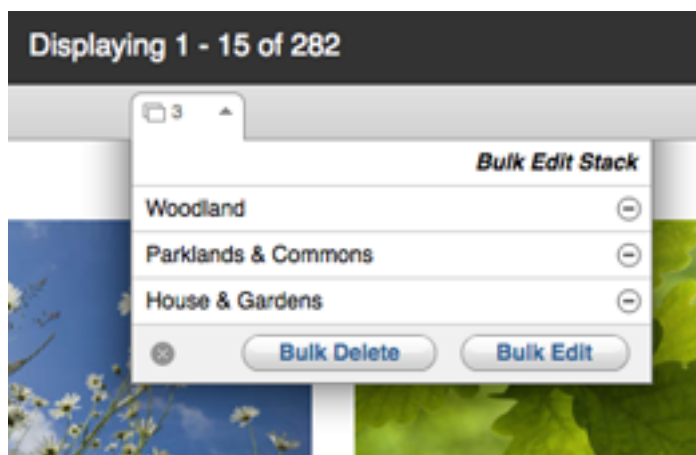
The first weapon for fighting the villain of complexity is a hatchet. [Studies](#) have found that 80% of users use only 20% of software features. Not only are all those unused features a waste of development time, but they frequently detract rather than add value by making the application more difficult to use. Applications that try to do everything often struggle to do *anything* well. A successful application is a lean application: one that isolates a single problem and solves it brilliantly. As [37signals](#) would say, “Underdo your competition.”

The best way to get a lean streamlined application is by eliminating features. When deciding whether a feature is necessary, the default answer should be “No.” Make features really fight to be included. If it doesn’t help the majority of users accomplish their frequent tasks, then it should probably be left out. Don’t get me wrong, being the one who swings the axe is tough, but when customers wax lyrical later on about how easy the application is to use, you can take all the credit.

Hide Complexity

If you can’t kill a complex feature, the next best thing is to hide it. Too often, rarely used yet complex features take up more screen real estate than frequently used yet simple features. This shouldn’t be. A good user interface should make the most common tasks the most prominent and then hide rare tasks so that they don’t get in the way.

When we were redesigning our content management system, we weren’t able to kill one complex feature in particular: the bulk edit function. In the previous version of the CMS, an entire column of a main screen was devoted to the bulk edit basket. The feature consumed about 30% of the screen real estate, but we found that very few people actually used it. Our solution was to introduce a single icon with a number beside it indicating how many items were in the basket. Clicking on the icon or number revealed the list of items in the basket through a modal dialog. This saved a huge amount of space and prevented the majority of users from being distracted from their primary task.



In the words of design guru John Maeda, “When a small, unassuming object exceeds our expectations, we are not only surprised but pleased.”

Minimize Visual Noise

So far, we’ve discussed minimizing complexity by killing or hiding features. But minimizing “perceived” complexity is also important. UX designer [Brandon Walkin](#) says, “The amount of visual noise in an interface has a great deal of impact on the perceived complexity of the interface.” Keeping visual noise to a bare minimum will make an interface seem easier to use. The two primary tools for reducing visual noise are white space and contrast.

White space, as defined by [Mark Boulton](#), “is the space between elements in a composition.” White space should be your default layout tool. One rule of thumb is never to introduce a design element if you could accomplish the same goal with white space. You will be surprised by just how much you can accomplish with white space alone.

While white space should be used in abundance, contrast should be used as little as possible. Design theorist [Edward Tufte](#), who is responsible for the notion of the “least effective difference,” urges designers to use the smallest visual variation required in order to effectively communicate an idea. Practically, this means emphasizing what’s important and dialing back everything else.

To illustrate both of these points, take a look at this pricing chart from [TypeKit](#) and then look at the modified version:

	Trial Try it for free	Personal Great for your blog	Portfolio Everything you need	Corporate Business-class service
Cost	Free!	\$7-month \$24.99 /year	\$17-month \$49.99 /year	\$49.99 /month
Monthly bandwidth ?	5 GB	10 GB	20 GB	50 GB
Font library ?	<u>Trial Library</u>	<u>Personal Library</u>	<u>Full Library</u>	<u>Full Library</u>

[Original version.](#)

	Trial Try it for free	Personal Great for your blog	Portfolio Everything you need	Corporate Business-class service
Cost	Free!	\$7-month \$24.99 /year	\$17-month \$49.99 /year	\$49.99 /month
Monthly bandwidth ?	5 GB	10 GB	20 GB	50 GB
Font library ?	<u>Trial Library</u>	<u>Personal Library</u>	<u>Full Library</u>	<u>Full Library</u>

Modified version.

You'll notice that the modified version contains a dark heavy border, not unlike the default border around HTML tables. Which version does a better job of minimizing visual noise? (The answer, of course, is the original.) Plenty of white space and minimal contrast help reduce visual noise and will make your application feel simpler.

Reduce, Reuse, Recycle

As an application begins to take shape, problems often recur in various parts of the application. And recurring problems characteristically tend to have similar solutions. Look for ways to reuse components of the interface. Reusing elements for different purposes has two advantages: less development time is needed, and a more consistent experience is given to the person using the application. If a user learns how to achieve one task, they can apply that same knowledge to accomplishing a second task as long as it is implemented consistently.

In the process of designing a CMS, we spent a lot of time perfecting form validation. We outlined invalid fields in red and even added red validation bubbles to indicate how many errors were in each section of the form.

Later in the process, a version comparison feature was added to the requirements list. Instead of creating an entirely new mechanism, we piggybacked on the validation framework. Form fields that had changed were outlined in blue, and we used blue bubbles for section totals. It was quick for us to deploy, and the interface was already familiar to users.

The screenshot shows the 'The Four Seasons' form with a dark header. The header includes the title 'The Four Seasons' with a '2 errors' badge and a 'Compare' button. Below the header are tabs for 'Core Info' (active, with a '2' badge), 'Item Links', and 'Comments'. The 'Core Info' section contains several form fields: 'Name' (empty, outlined in red), 'Description' (empty, outlined in red), 'ID Number' (containing '78666'), and 'Ratings' (two sliders, one for 'Internal Rating' at 3 and one for 'Official Rating' at 5). Red error messages 'This field is required.' are shown next to the 'Name' and 'Description' fields.

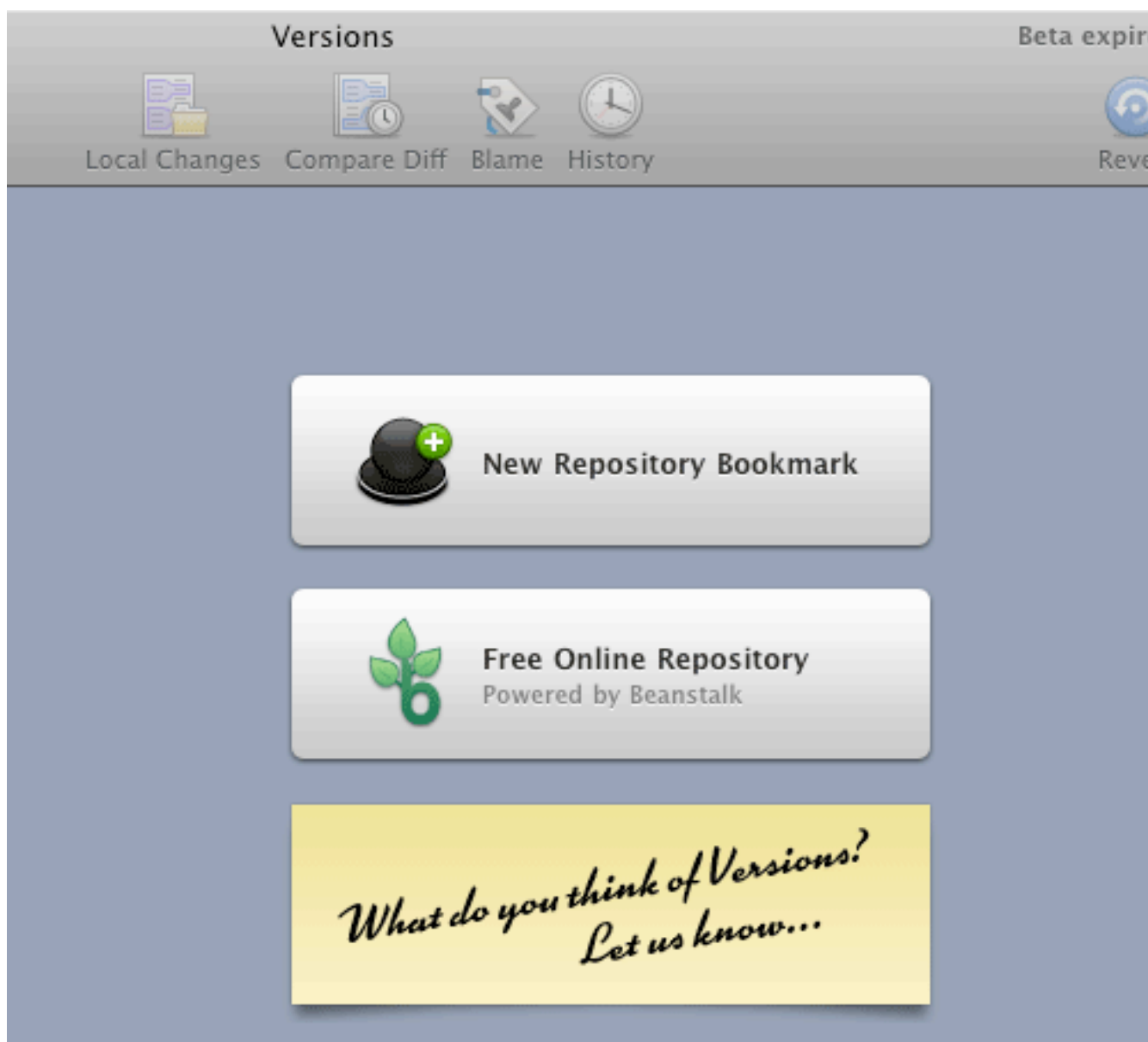
The screenshot shows the 'The Four Seasons' form with a dark header. The header includes the title 'The Four Seasons' with a '4 changes' badge and a 'Compare' button. Below the header are tabs for 'Core Info' (active, with a '4' badge), 'Item Links', and 'Comments'. The 'Core Info' section contains several form fields: 'Name' (containing 'The Four Seasons', outlined in blue), 'Description' (containing 'An outstanding hotelier with international prestige.', outlined in blue), 'ID Number' (containing '78666'), and 'Ratings' (two sliders, one for 'Internal Rating' at 3 and one for 'Official Rating' at 5). A blue bubble '4' is shown next to the 'Core Info' tab. A tooltip titled 'Modified by admin' is displayed over the 'Description' field, showing the text 'Four Seasons An outstanding hotelier with international prestige.' with 'Four Seasons' in red and 'An outstanding hotelier with international prestige.' in green.

Reusing interface components is another way to reduce perceived complexity because users will then quickly become familiar with the application and know what to expect from it.

The Blank State Should Not Be Blank

The blank state is how an interface appears when no data is available, such as when someone uses an application for the first time. As designers, we spend most of our time figuring out how best to present content, and too often we forget that content simply won't always be there.

It is vital to build in sensible defaults. The blank state is often the first impression someone has of a product and can be a deciding factor in whether or not they even get started using the application. A good blank state acts as a wayfinding device, helping the user take their first few steps.



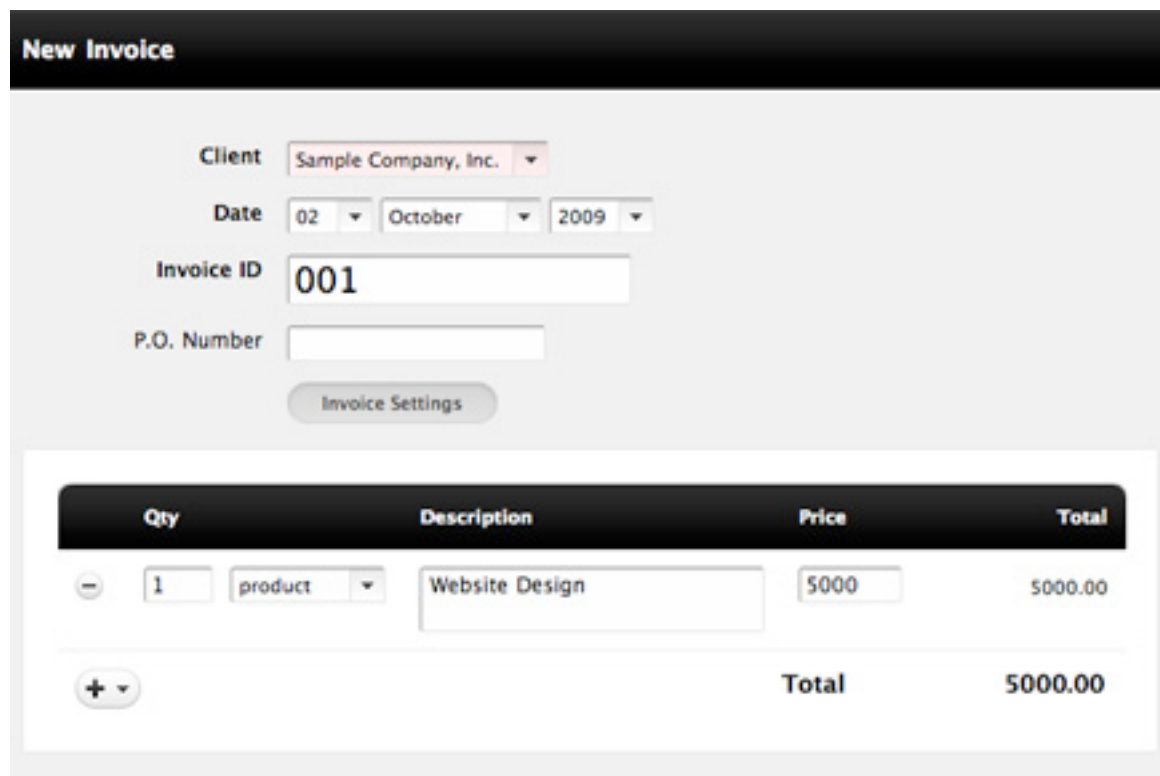
The people behind [Versions](#) (a subversion client for Mac) obviously put a lot of thought into the blank state of the application. The application prominently highlights the top two actions a user might want to take when using it for the first time.

It bears repeating: don't forget the blank state!

Showcase

We've all seen plenty of cluttered, complex software. Applications that effectively minimize complexity, however, are a bit harder to come by. Here are few that pull it off with flying colors.

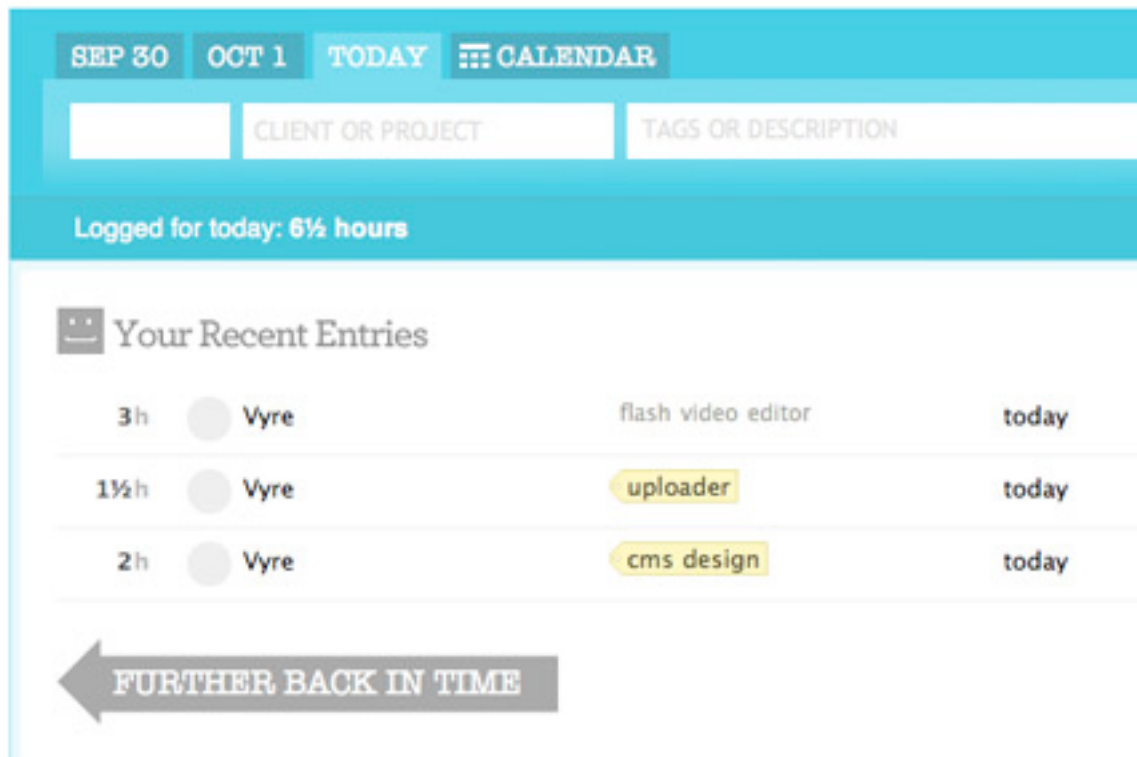
[Invoice Machine](#) is user interface minimalism at its best. The application exemplifies economy of expression and pays extreme attention to detail.



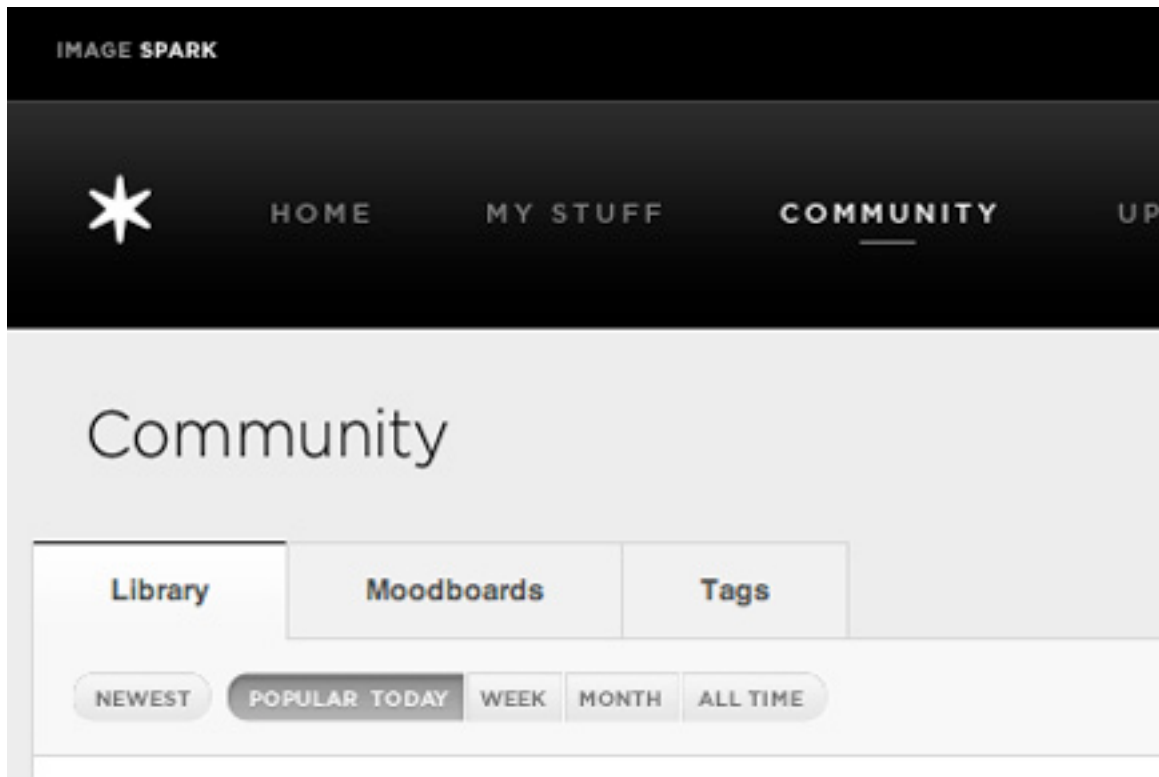
The screenshot displays the 'New Invoice' interface of the Invoice Machine application. The top section features a dark header with the text 'New Invoice'. Below this, the form includes fields for 'Client' (a dropdown menu showing 'Sample Company, Inc.'), 'Date' (three dropdown menus for day, month, and year, showing '02', 'October', and '2009'), 'Invoice ID' (a text input field containing '001'), and 'P.O. Number' (an empty text input field). A button labeled 'Invoice Settings' is positioned below these fields. The bottom section contains a table with four columns: 'Qty', 'Description', 'Price', and 'Total'. The table has a single row for 'Website Design' with a quantity of '1', a price of '5000', and a total of '5000.00'. A 'Total' row at the bottom shows the overall total of '5000.00'. The interface is clean and minimalist, with a light gray background and dark text.

Qty	Description	Price	Total
1	Website Design	5000	5000.00
Total			5000.00

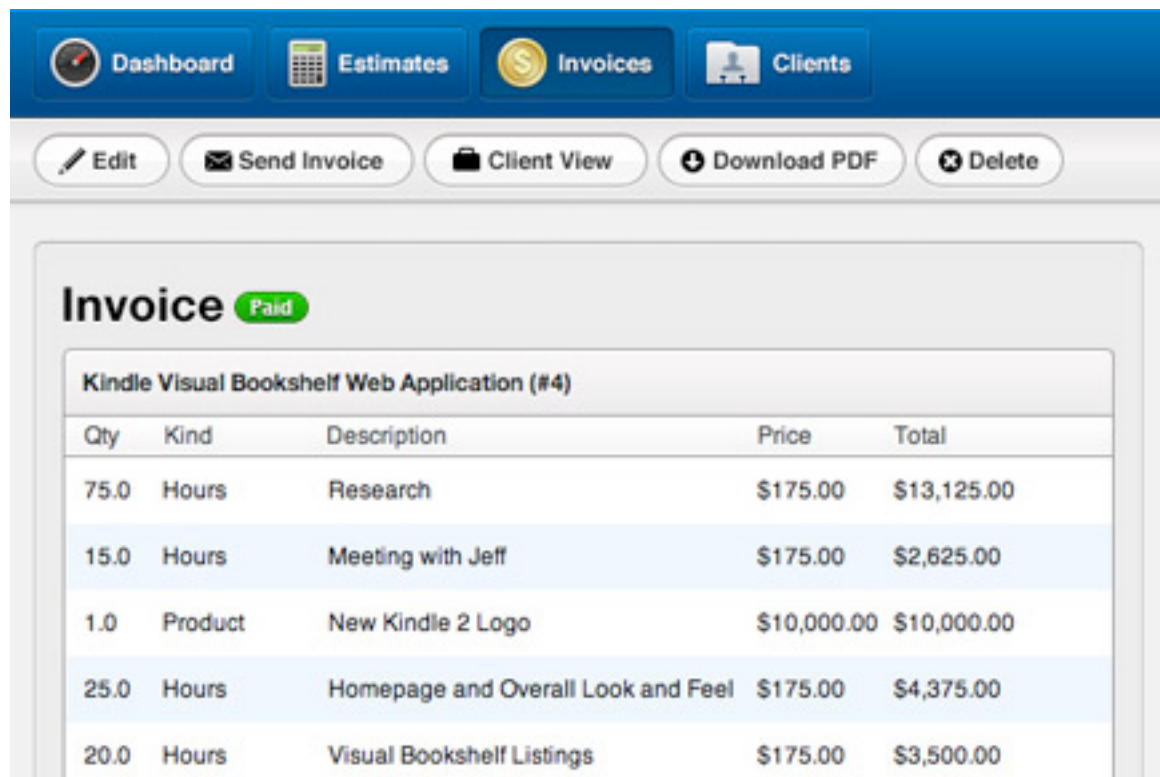
[Freckle](#) just makes you *want* to keep track of your time. The conveniently condensed interface shines with energetic color, turning a routine task into something fun.



[Image Spark](#) combines sleek black-and-white gradients with elegant typography. Its interface has few controls, making it simple to get around.



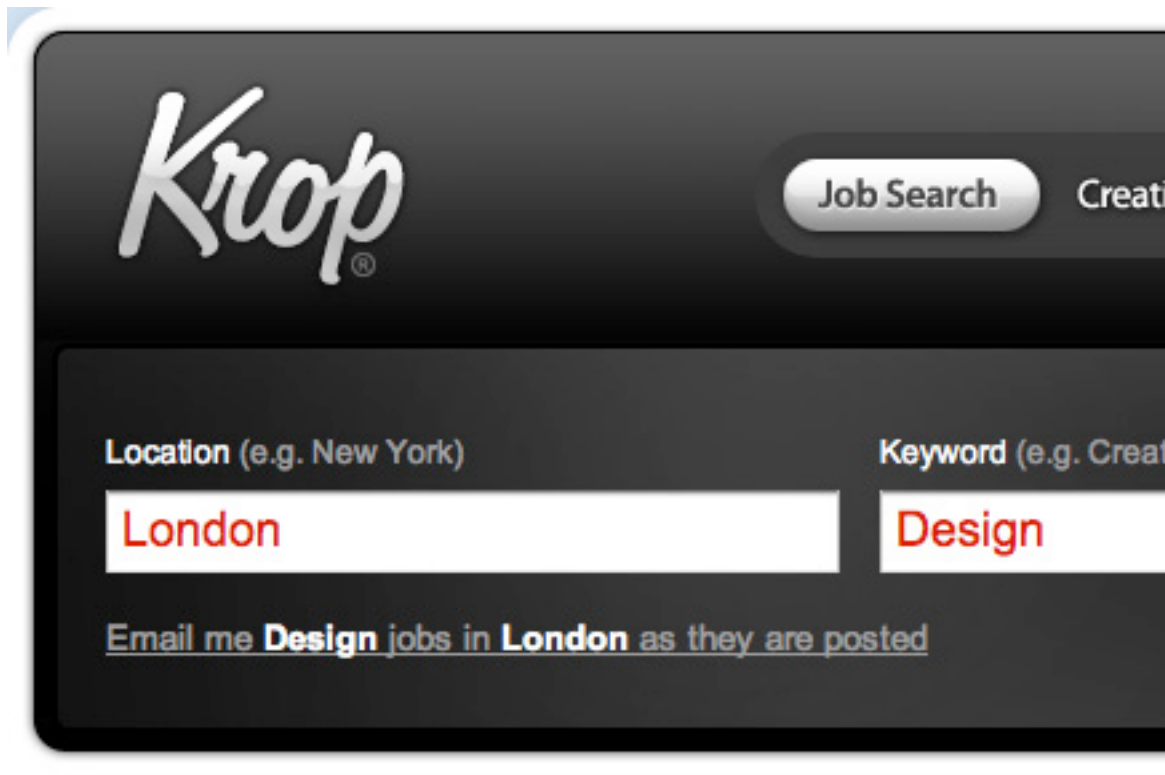
[Ballpark](#) has clear controls and an uncluttered interface, without being sparing in its use of color.



The screenshot displays the Ballpark software interface. At the top, there is a blue navigation bar with four buttons: 'Dashboard' (with a speedometer icon), 'Estimates' (with a calculator icon), 'Invoices' (with a gold coin icon), and 'Clients' (with a person icon). Below this bar is a light gray toolbar containing five buttons: 'Edit' (with a pencil icon), 'Send Invoice' (with an envelope icon), 'Client View' (with a briefcase icon), 'Download PDF' (with a download icon), and 'Delete' (with a trash can icon). The main content area features a section titled 'Invoice' with a green 'Paid' status badge. Below the title is a table for 'Kindle Visual Bookshelf Web Application (#4)'. The table has five columns: 'Qty', 'Kind', 'Description', 'Price', and 'Total'. It lists five items with their respective quantities, kinds, descriptions, prices, and totals.

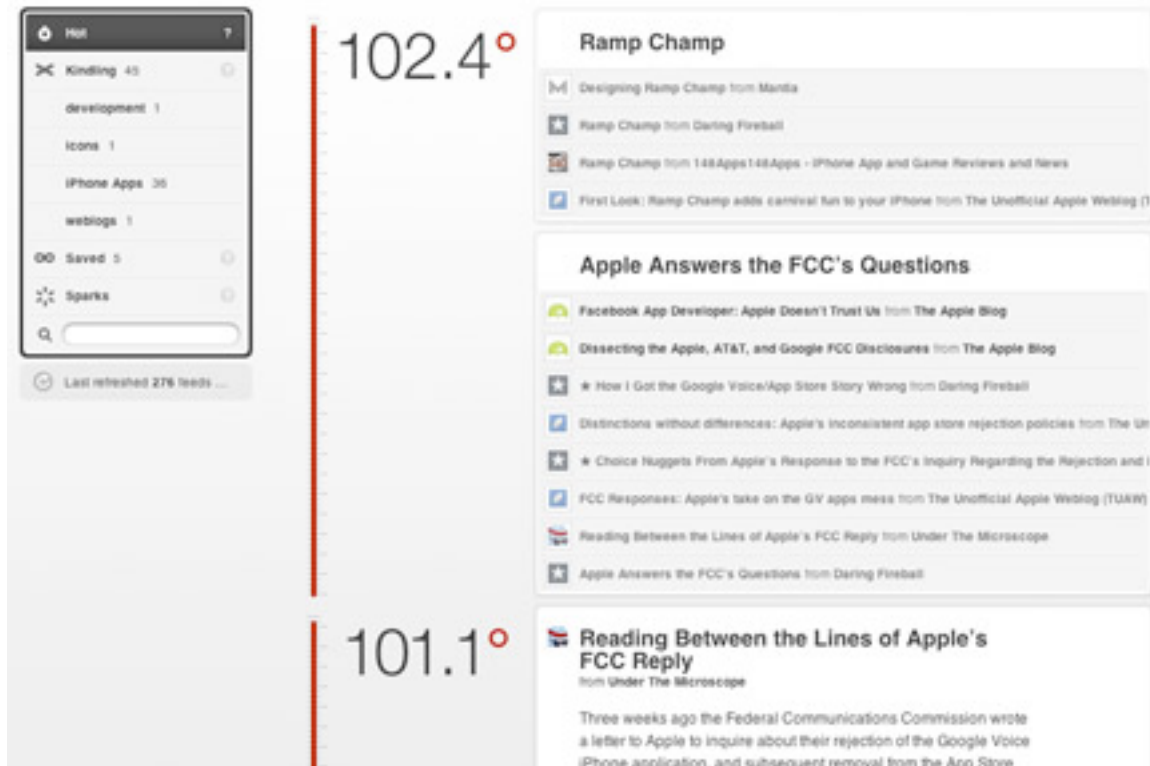
Qty	Kind	Description	Price	Total
75.0	Hours	Research	\$175.00	\$13,125.00
15.0	Hours	Meeting with Jeff	\$175.00	\$2,625.00
1.0	Product	New Kindle 2 Logo	\$10,000.00	\$10,000.00
25.0	Hours	Homepage and Overall Look and Feel	\$175.00	\$4,375.00
20.0	Hours	Visual Bookshelf Listings	\$175.00	\$3,500.00

[Krop](#), at its core, is just two form fields: location and keyword. It takes literally about five seconds to find relevant job postings.

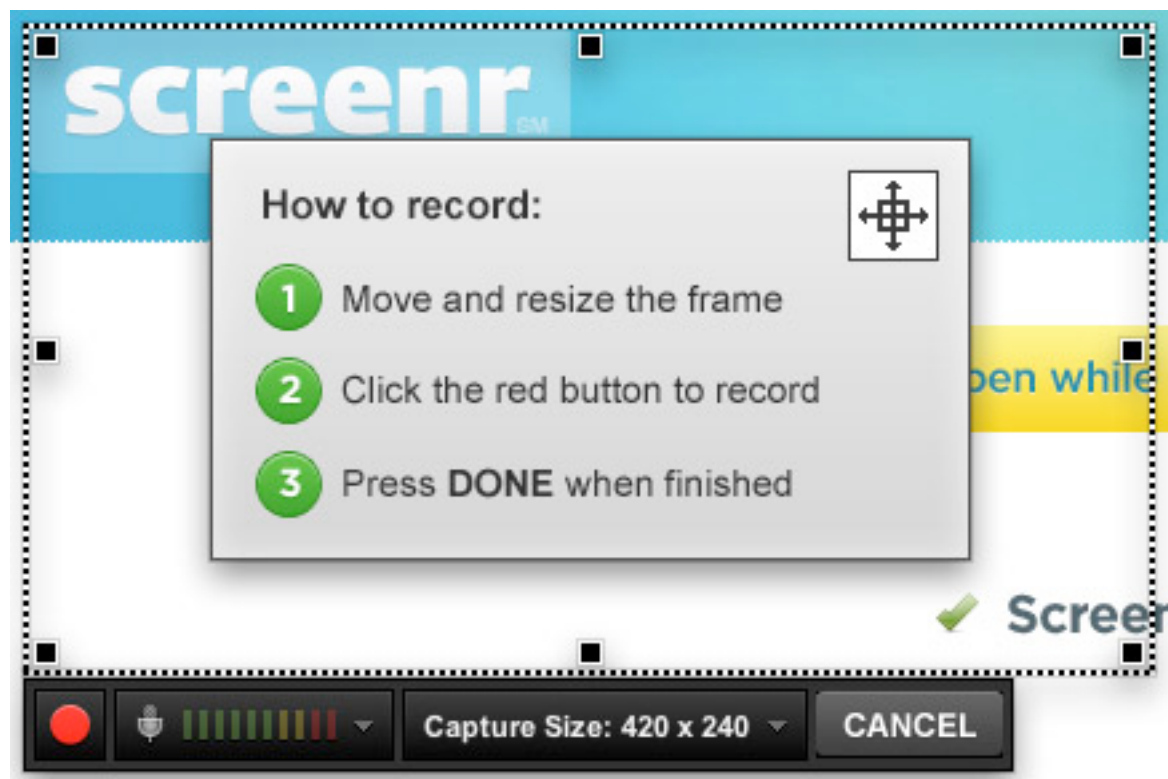


The image shows a dark-themed web interface for Krop. At the top left is the 'Krop' logo in a white, stylized script font. To its right is a 'Job Search' button with a white border and a slight shadow. Further right is a partially visible 'Create' button. Below these, there are two input fields. The first is labeled 'Location (e.g. New York)' and contains the text 'London' in red. The second is labeled 'Keyword (e.g. Creati' and contains the text 'Design' in red. Below the input fields, there is a line of text that reads 'Email me **Design** jobs in **London** as they are posted', where 'Design' and 'London' are bolded and underlined.

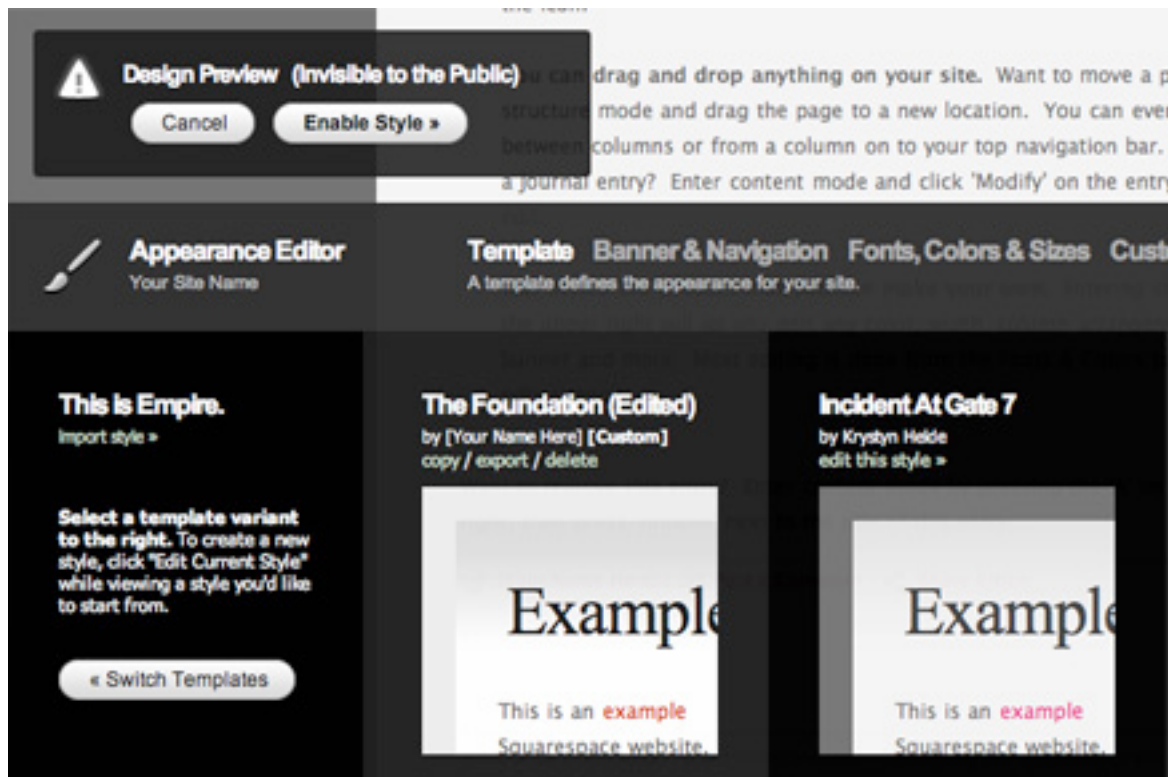
[Fever](#)'s entire purpose is to reduce the barrier between you and blog posts you're interested in. It accomplishes this through a simple, unique interface.



[Screenr](#) is an amazingly simple tool for recording screencasts and posting them to Twitter. It is notable not for all of the features it has, but for all of the features it *doesn't* have.



[Squarespace](#) does a great job of hiding complexity. While it is a full-featured Web publishing platform, it does an impressive job of hiding its power behind a simple user interface.



The Final Word

First, reduce actual complexity by eliminating unnecessary features and then hiding what you can't eliminate. Secondly, reduce perceived complexity by minimizing visual noise and reusing elements. And finally, use the blank state to help orient users.

Minimizing complexity in the user interface will help people learn your application more quickly, use it more effectively and be happier all the while. As jazz musician Charles Mingus said, "Making the simple complicated is commonplace; making the complicated simple, awesomely simple, that's creativity."

A Guide To Heuristic Website Reviews

Leigh Howells

In this article, we'll explore a scoring system for rating and comparing websites, we'll visualize those ratings using info graphics, and we'll see what data and structure this method provides for reviewing websites.

HOW TO TELL WHETHER A WEBSITE IS JUNK

We are all reviewers. We review many websites every day without even realizing it. In fact, many of us are experts at it. We don't realize it because the whole process occurs in moments.

That's how it is. We use websites; we judge websites. Even if we don't know we're doing it, we make judgements about trustworthiness, credibility, competency, reliability, design and style within seconds of arriving on a Web page. After looking around, we also get a pretty good feel for the user experience and usability.

CONSULTANCY REVIEWS

For many years, the agency I work for has conducted detailed reviews of its clients' websites. As part of the consultancy process, we offer recommendations for any redesign or redevelopment work that is necessary.

Snap judgments may be useful and unavoidable, but when it comes to reviewing websites professionally, we need to be more organized and thorough, and we do this by using a review methodology. It also pays in

both time and effort to be formulaic and consistent in our approach, because there are so many things to look at when considering a website.

To make this easier, we use a set of heuristics to score websites, along with a simple method to quickly visualize any weaknesses. I use a set of heuristics that I have worked with and edited and updated to suit the type of projects I work with, based on original work created by User Focus.

Heuristics



(Image: [Rick S.](#))

A heuristic is just a fancy word for a measurement of something that can't readily be quantified (i.e. when there are no actual numbers to judge whether item A is better or worse than item B). In a 100 meter sprint, the winner is easily identified by concrete data. In ice dancing, the contestants are judged based on a set of technical and artistic criteria, giving them a set of scores.

ALL THAT GLITTERS IS NOT GOLD

We might be swayed by something that looks good, but we all know that beauty is only skin deep. As with everything that glitters, the job of the reviewer is to poke about and see if they really have struck gold.

Conversely, some websites that are judged harshly for their graphic design are successful beyond measure — I'm looking at you, Amazon, eBay, Craigslist and even Google. These websites aren't much to look at, but functionally speaking, they do their job well and have evolved over the years to precisely meet their customers' needs.

As designers, we're asked to redesign websites that generally are getting to look better and better. It's getting to the point that we find ourselves questioning the need for a redesign at all. But usually the problems are not immediately obvious in the visuals, layout or code. Sometimes a website is just wrong for the client's brand; or the experience of performing tasks on it is unpleasant. Sometimes, a website just doesn't work.

You can't tell by looking. You need to dig deeper by really using the website, setting yourself tasks and trying things out. Only then will you experience what is really going on. Realizing just how much rethinking, redesigning and redeveloping a website needs often takes a while.

METRICS FOR SUCCESS

The success of most websites can be measured by some metric, be it the number of sales, uploads, downloads, clicks, comments or sign-ups. But a website can be successful in sales and still have problems; for example, it might be successful because of excellent marketing, because of its offline reputation (as in the case of high-street brands) or from having cornered the market. That does not mean it is without problems. But many more websites have no quantifiable metrics by which we can determine how good or bad they actually are. Judging these websites is more difficult and requires a bit more leg work.

A Many-Layered Cake



(Image: [Scheinwerfermann](#))

When reviewing a website in detail, we have to explore many layers, both on the surface and below, including the following:

- Task orientation and website functionality,
- Navigation and information architecture,
- Forms and data entry,
- Trust and credibility,
- Quality of writing and content,
- Search,
- Help, feedback and error tolerance,
- Page layout and visual/aesthetic design,
- Accessibility and technical design.

Taking these broad categories, we can devise a list of questions to explore each and get to the heart of the website. This formalizes the process and ensures that the same thought process can be repeated the next time. It also serves as a checklist, ensuring that nothing is forgotten. For example, when looking at the layout and visual design of a website, our questions could include these:

- Are standard elements (such as page titles, website navigation, page navigation and privacy policy) easy to locate?
- Is there a good balance between information density and white space?
- Does the website have a consistent and clearly recognizable look and feel that will engage users?

For accessibility, we could formulate questions such as these:

- Is the color contrast across the website enough to make all of the content accessible?
- Does the website work comfortably at lower resolutions (e.g. 1024 × 768 pixels)?
- Does the CSS validate with the W3C's validation services?

Regarding the written copy, our questions could include:

- Are the pages simple to scan on screen? Are they broken up by headings and subheadings? Are the paragraphs short?
- Are acronyms and abbreviations defined when first used?
- Does the website favor maps, diagrams, graphs, flow charts and other visuals over long blocks of text?

DEPTH

Although relatively easy to conduct, a heuristic review is not quick to perform. However, we can decide just how much depth to go into and how many questions to ask in order to get a feel for the website. The more heuristic measures we use, the longer the process will take; the fewer we use, the less informative the results will be. It's a matter of striking a balance between the time available and the quality of returns. Selecting heuristics that get to the heart of each category can significantly reduce the amount of effort you need to put in.

Devising A Scoring System

To get a yardstick score for each heuristic, a simple score can be given. For example, 0 points if it falls short of a metric, 1 point if it's halfway there, and 2 points if it does the job. So, if acronyms or abbreviations are defined in some sections but not in others, then the heuristic would score only 1 point. If the website worked comfortably at 1024×768 pixels, then it would receive 2 points.

These points can be totaled across each category to give a quantifiable sense of what's going on across the website, as shown here:

Heuristic Design and Usability Review

Summary of results

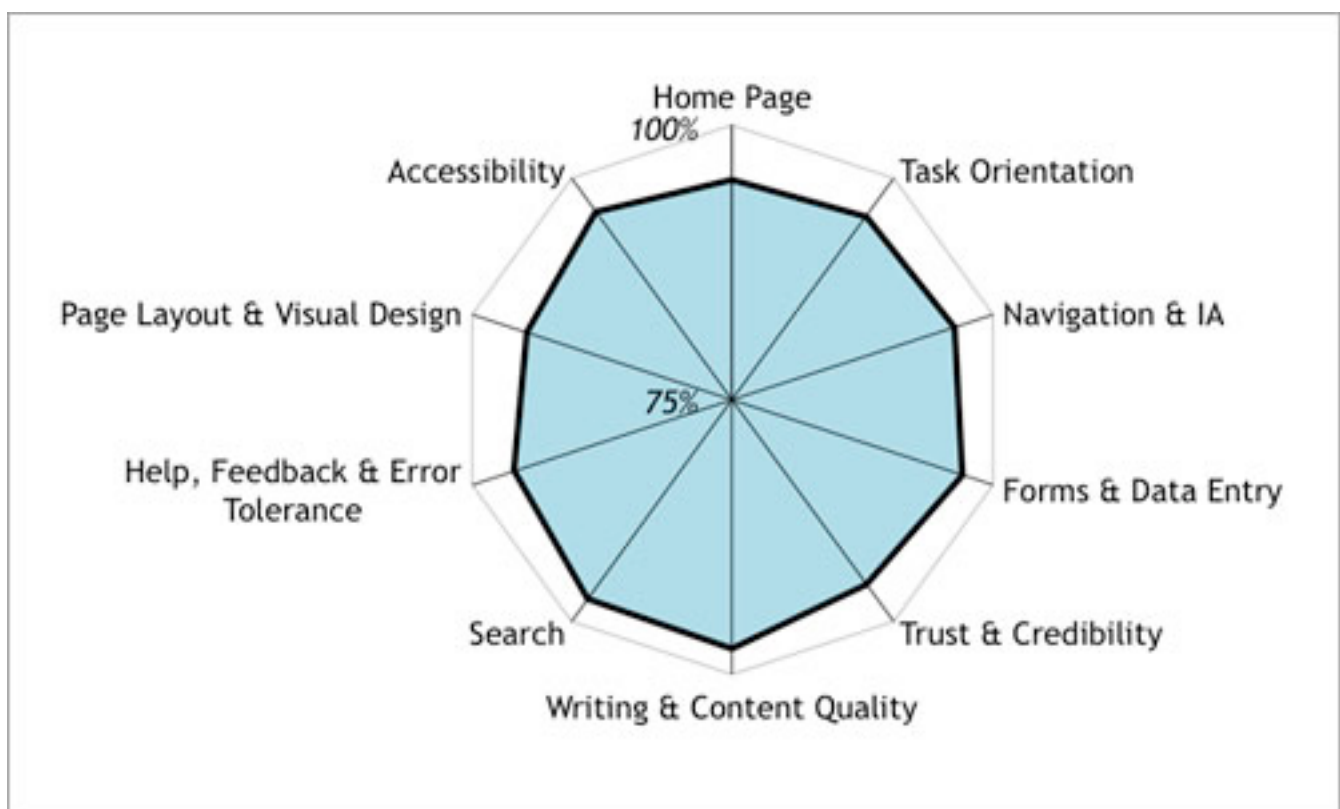
	Questions	Possible Score	Actual Score	Result
Home Page	20	40	38	95%
Task Orientation	35	70	67	96%
Navigation & IA	27	54	52	96%
Forms & Data Entry	17	34	33	97%
Trust & Credibility	12	24	23	96%
Writing & Content Quality	22	44	43	98%
Search	20	40	39	98%
Help, Feedback & Error Tolerance	25	50	48	96%
Page Layout & Visual Design	38	76	72	95%
Accessibility	26	52	50	96%
Overall score	242	484	465	96%

Totals of heuristic data across categories.

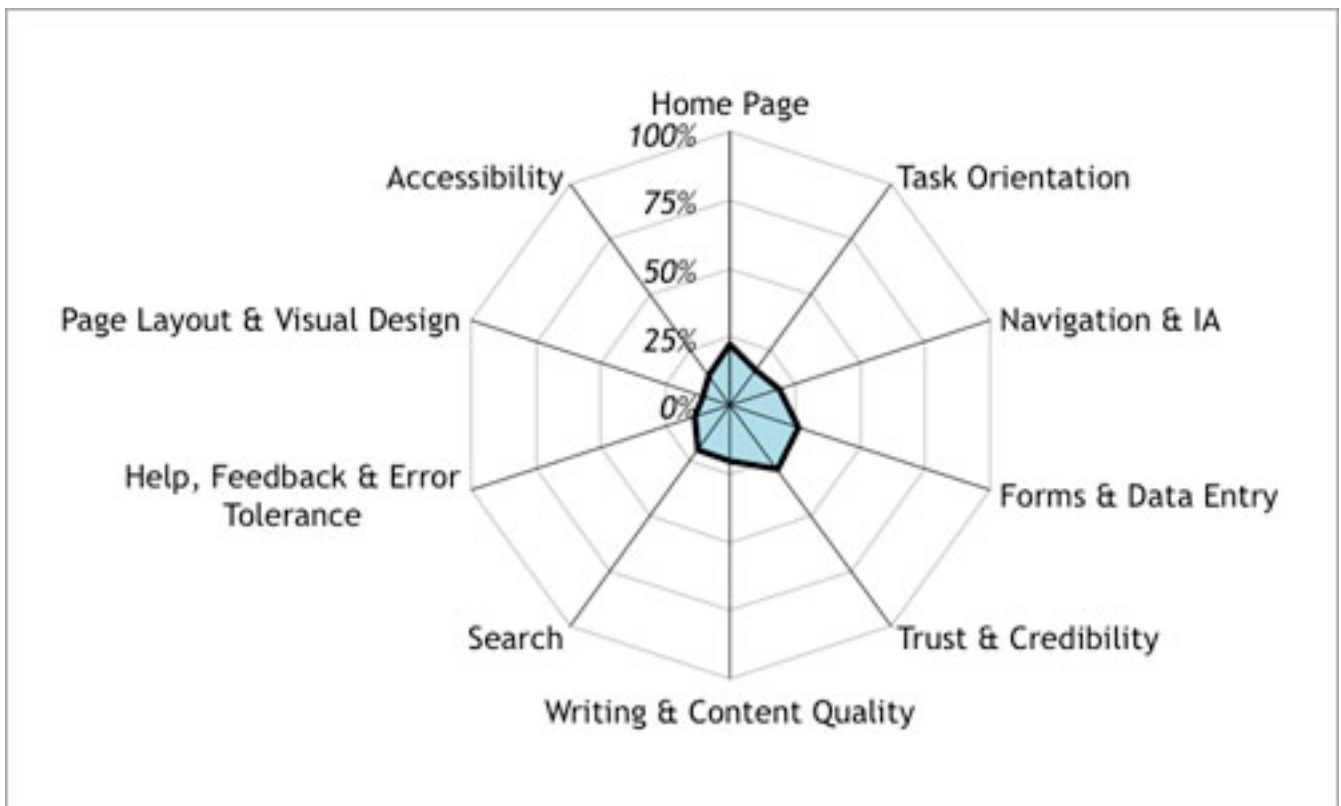
VISUALIZATION

Representing this data visually helps us quickly identify problem areas and makes it easier to compare websites.

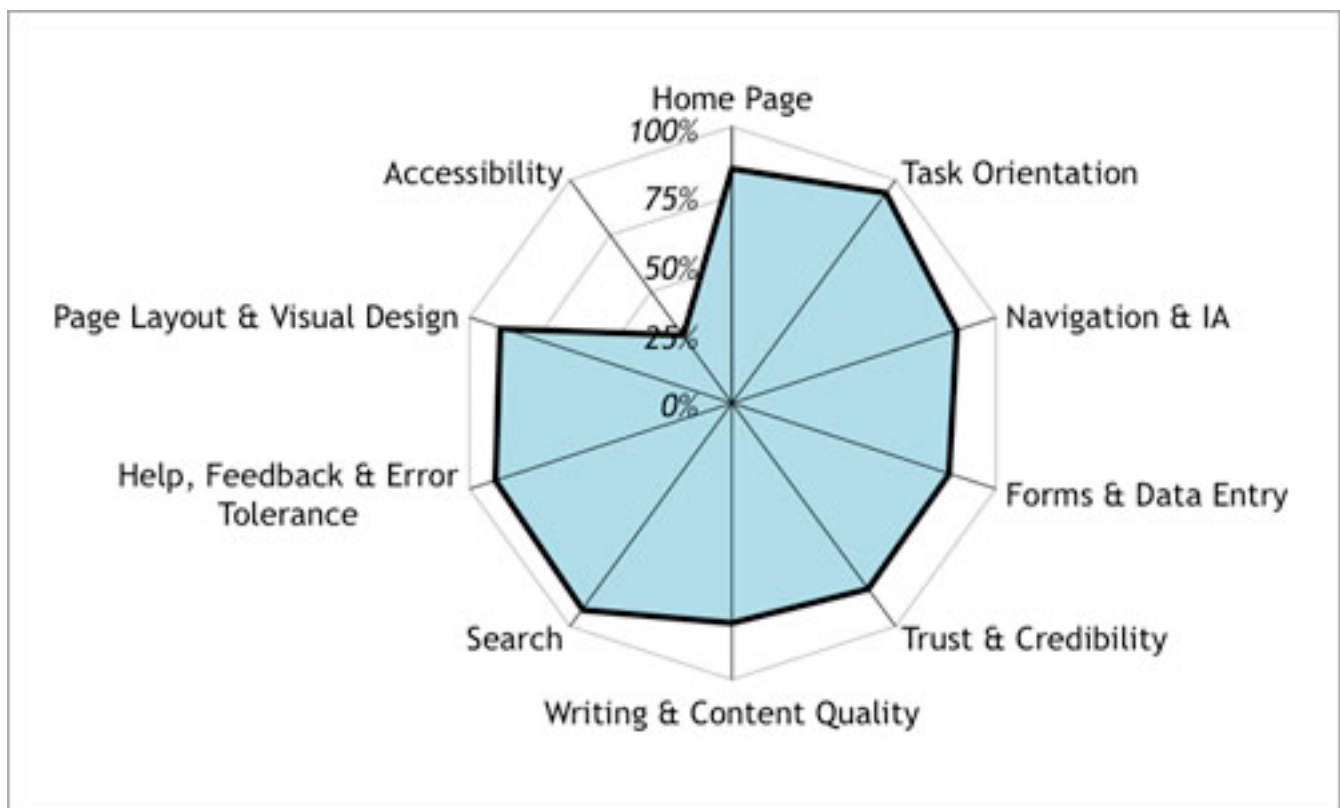
Radar diagrams are perfect for this kind of analysis, because they give a recognizable shape based on the score. The more circular the radar, the more balanced the score; the spikier the radar, the more variation in the score. The size of the radar plot on the axes indicates the score percentage itself, showing good and bad areas, as seen in the examples below:



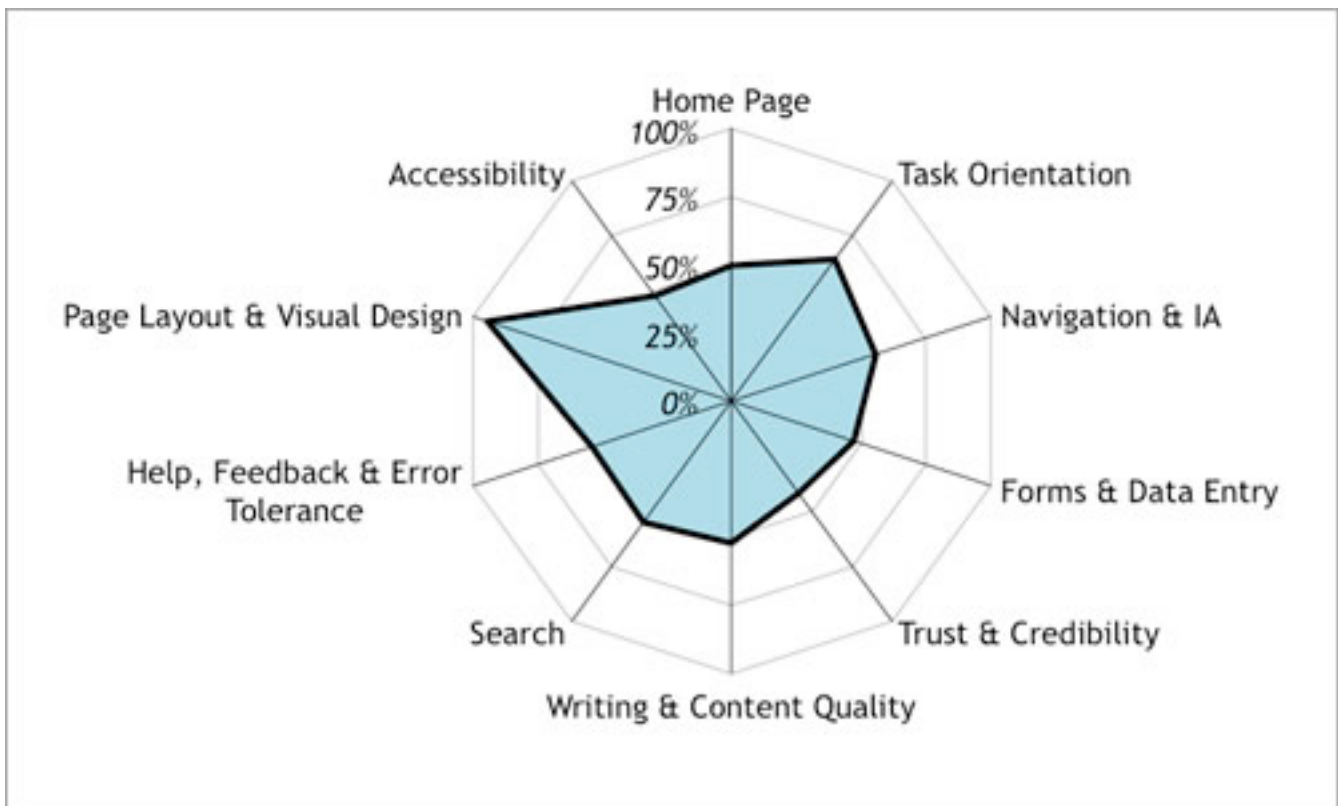
A radar plot showing a website that performs well across all heuristic categories.



A radar plot showing poor performance across all heuristic categories.



A radar plot showing a website that performs well in all areas but one.



A radar plot showing a website that performs poorly in all areas but one.

Competitor Reviews

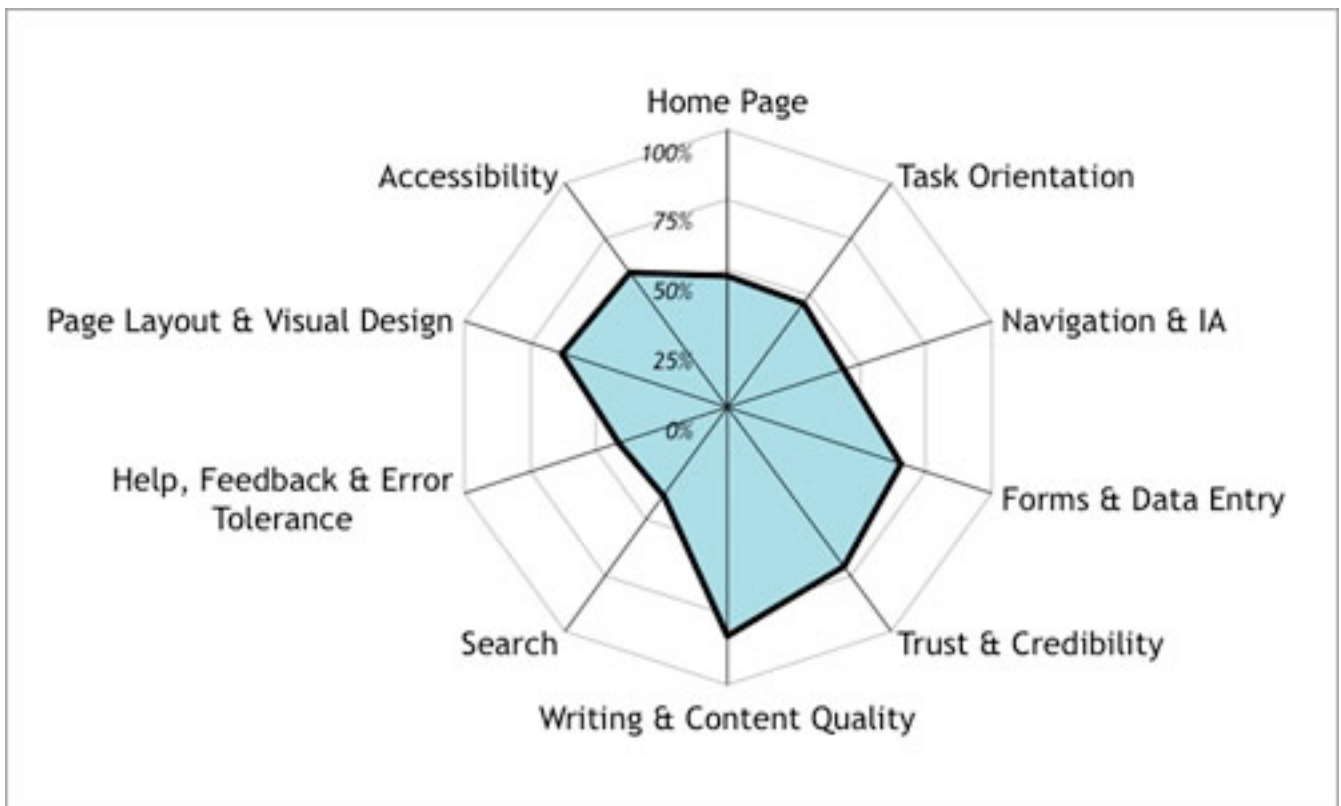
By combining the heuristic results of different websites, we can create a visual comparison of competing websites in a market segment. This is particularly good for getting a feel for which websites fail and which succeed in certain respects. Analyzing multiple websites can, of course, take a lot of work, so stripping your heuristics down to the essentials is a good idea.

A DIRECT COMPARISON

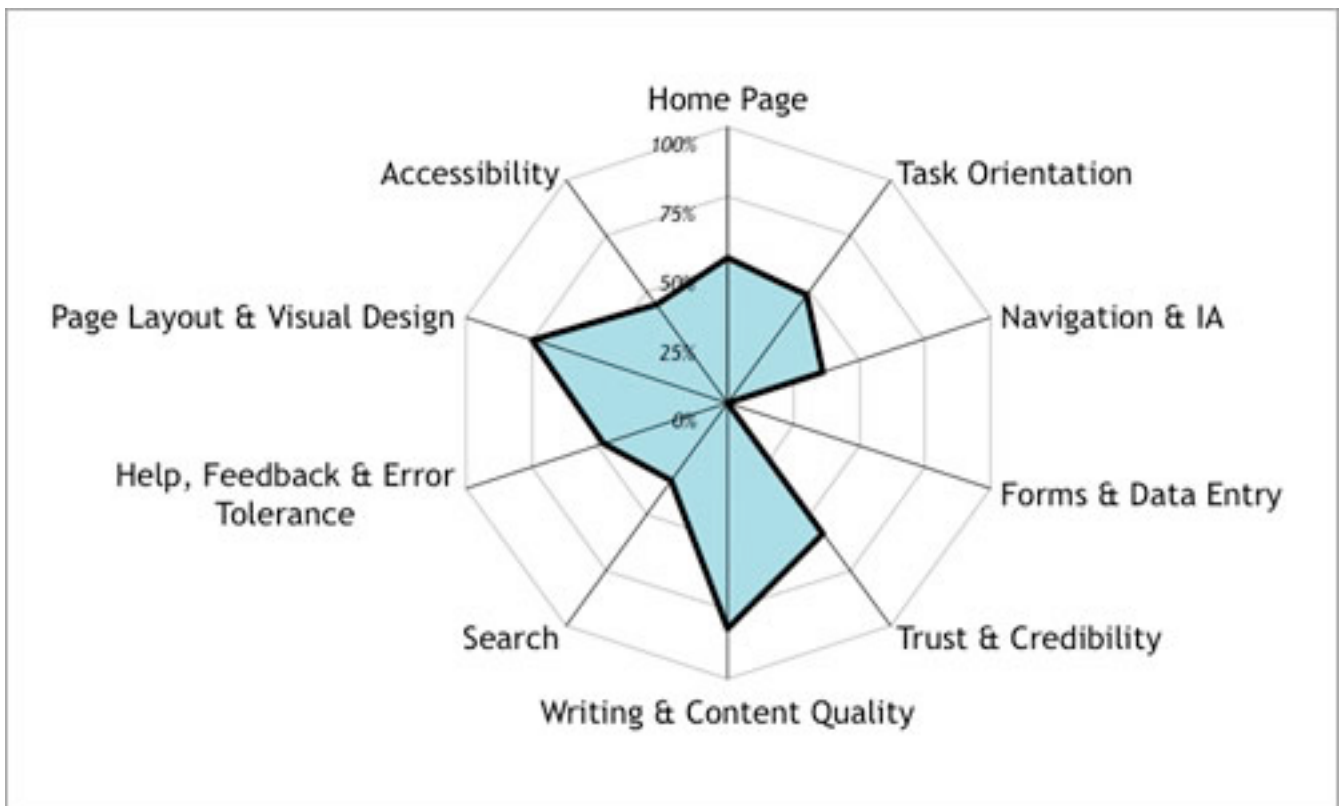
As a real-world example, below is a comparison of two similar websites: Smashing Magazine and Web designer Depot. We can see that both lack a little in most of the categories, apart from quality of writing and content, which is what we would expect from content-rich blogs. (Please note that I work for neither website and stand as an impartial bystander!)

Both websites score a little higher in page layout and visual design, but they have rather weak home pages, being in the format of a traditional, basic blog. Their calls to action score quite poorly (other than the advertising!). Smashing Magazine scores marginally better in navigation because it has the tabs on top to distinguish major content areas, whereas Web designer Depot almost loses the navigation below the advertising in the right-hand column. Smashing Magazine scores slightly higher in accessibility for a number of minor heuristics, such as the clarity of the text, spacing and contrast.

Web designer Depot falls behind a little on trust and credibility because of details such as the basic link to an email address in the footer (compared to the well-considered contact form on Smashing Magazine), and also for the very brief copy in the “About us” section. However, Web designer Depot picks up slightly more points in visual design for its colorful style. Of course, like the presentation scores in ice dancing, any process used to score aesthetics or design will always be subjective, so having a wide range of criteria for various aspects of design is a good idea.

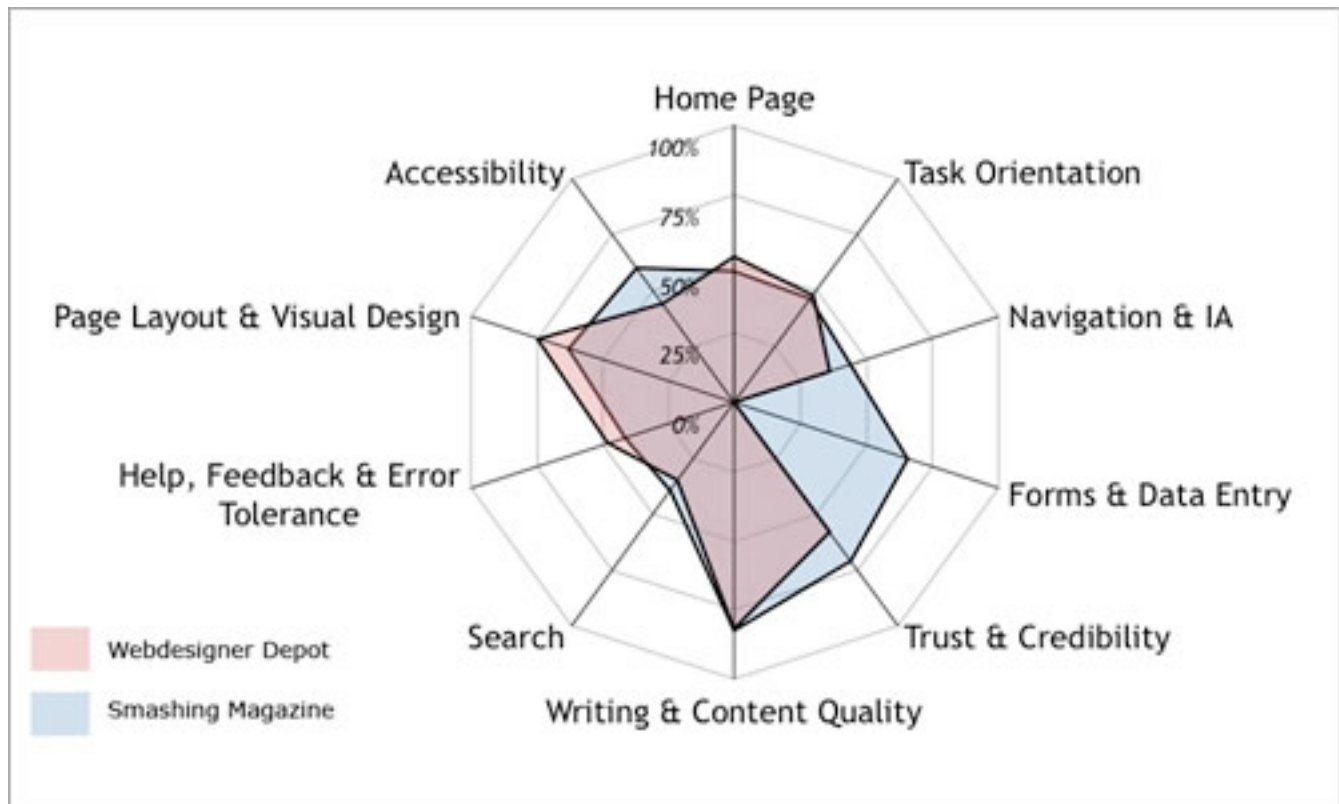


A heuristic analysis of Smashing Magazine.



A heuristic analysis of Web designer Depot. Note that Web designer Depot does not really have or require form inputs, so it scores 0 by default in the “Forms and data entry” category; this score can be either ignored or removed altogether if so wished.

To emphasize the differences in the heuristic measurements, we can overlay one radar plot on the other:



Overlaying one radar diagram on the other to enhance visualization.

Conclusion

When reviewing a website, subjective snap judgements are unwise. We can do justice to a website only with a detailed test drive. We need to perform tasks and look in detail at various components on and below the surface. Heuristic scoring is a useful process for quantifying and visualizing a website's quality when other measures are not appropriate or available. This formal process reveals problem areas, while focusing the discussion at the start of a redevelopment phase.

Stop Designing Pages And Start Designing Flows

Chuck Loganecker

For designers, it's easy to jump right into the design phase of a website before giving the user experience the consideration it deserves. Too often, we prematurely turn our focus to page design and information architecture, when we should focus on the user flows that need to be supported by our designs. It's time to make the user flows a bigger priority in our design process.

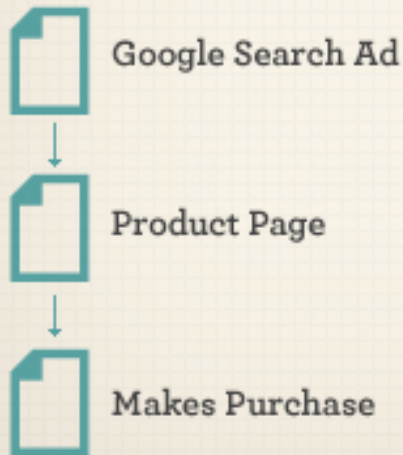
Design flows that are tied to clear objectives allow us to create a positive user experience and a valuable one for the business we're working for. In this article, we'll show you how spending more time up front designing user flows leads to better results for both the user and business. Then we'll look in depth at a common flow for e-commerce websites (the customer acquisition funnel), as well as provide tips on optimizing it to create a complete customer experience.

Start With The User

When starting a new Web design project, we're often handed a design brief, branding standards, high-level project goals, as well as feature and functionality requirements. Unfortunately, these documents typically amount to little more than the technical specifications of the project, with almost no thought given to how exactly the website will fulfill the multiple user objectives that lead to successful interactions.

USER FLOWS

eCommerce



Membership / Subscription



Two examples of popular user flows for e-commerce and subscription websites.

If you start with a detailed look at the objectives of the user and the business, you would be able to sketch out the various flows that need to be designed in order to achieve both parties' goals. User objectives could range from finding a fact to replacing a product to learning a new skill to buying a gift for someone. Business objectives could be getting a lead, a like, a subscriber, a buyer, a download or a phone call. Identifying each user and business objective is the first step to creating design flows that meet all of them.

Map User Flows Into Conversion Funnels

Not all website visitors are created equal. Users come from different sources, with varying levels of knowledge and engagement, and with different goals. It's up to you as a user experience designer to map those inbound user flows to conversion funnels that provide value to the user as well as the business. You should prioritize the flows and focus your effort on the few that will impact the most users and have the greatest gain. Custom flows allow you to architect experiences according to traffic source or visitor type and enable you to set the experiential pace, build user confidence and get buy-in on the way to the ultimate conversion.

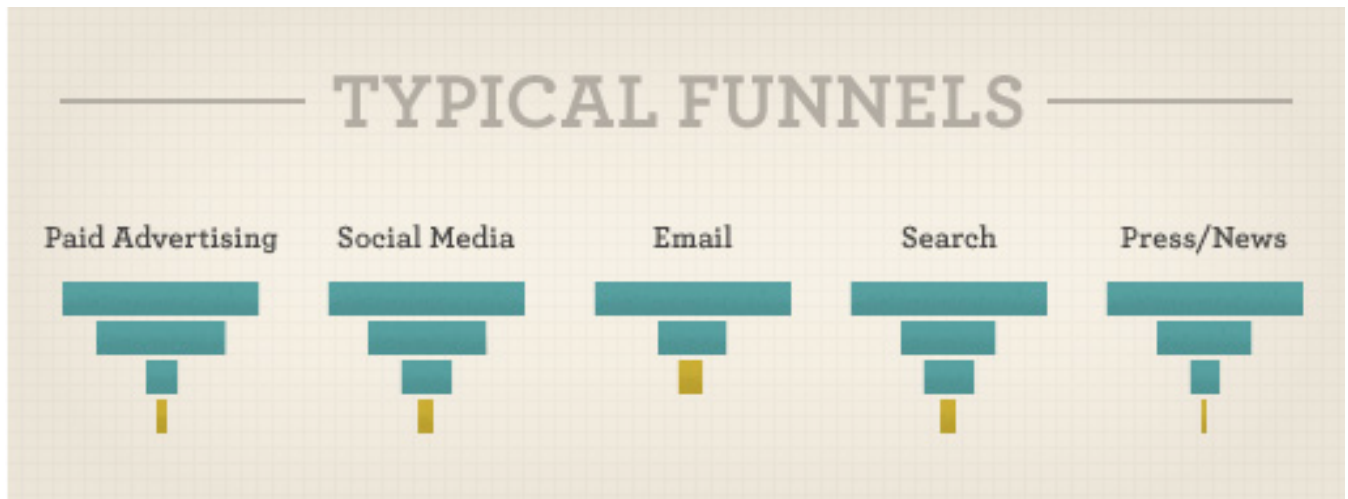
TYPICAL USER FLOWS

Some typical user flows are:

- **Paid advertising**
A user coming from a banner or Google AdWord ad.
- **Social media**
A user coming from a friend's post on a social network.
- **Email**
A user coming from an email newsletter or referral invitation.
- **Organic search**
A user coming from a deep link that was surfaced by a search.
- **Press or news item**
A user coming from a mention in the news or a blog post.

Each of these visitors has their own needs, expectations and level of knowledge, and they need to be treated accordingly.

Diving Into Funnels: A Closer Look At Customer Acquisition



E-commerce websites typically have many different conversion funnels.

Let's look at a critical flow for many websites — paid online customer acquisition — and break down its various elements. For this example, we'll examine the experience flow from new visitor to email subscriber to purchaser.

Consider a company that uses display advertising to generate new customers for its business.

DISPLAY MEDIA

With display advertising, it all starts with the banner. The design of the banner needs to achieve one precious goal: get a click from the right person. Here are some key questions to answer when designing the ads that represent the very front of your user flow:

- What type of user am I targeting?
- Are they actively seeking a solution to a problem, or are they casually browsing?
- What problem are they trying to solve?
- How can I best capture the user's attention?
- How do I relate to the user?
- Is there a message that will resonate with the user?
- Is there a pain point that my product or website alleviates for the user?
- How can I articulate this solution clearly and quickly?
- What compelling calls to action will get our target user to click?

Your ads should address these main motivations and provide a compelling hook to get that all-important click. Up-front research and real-world testing will help to optimize the experience. Using this model, ReTargeter improved its banner click-through rate by four times. Its [blog post](#) lays out exactly how it achieved this success.

THE LANDING PAGE

The point when the user hits the landing page is when the user flow work really begins. Because these users are coming from a low-information source (such as a banner, as opposed to an in-depth blog post), you must

design a flow that fills in the gaps of information by providing the user with the data that they need to be converted.

In our example, the user will hopefully be converted to an email subscriber; but depending on the business, the conversion could be to create an account, download a white paper or make a purchase. Whatever the conversion goal for the business, the key is to give the user a reason to keep moving through the flow, down the funnel.

Use the following methods to keep the user moving down the funnel:

- Build user confidence by clearly articulating key benefits, backed by easy-to-digest proof points.
- Streamline the content and design to focus on a clear call to action (in this example, to sign up for an email newsletter).
- Remove friction at every step. Ask for the minimum amount of information, and reduce the number of fields, extra clicks and page-loading time.
- Create an enticing hook, an itch that can only be scratched by completing the registration step.

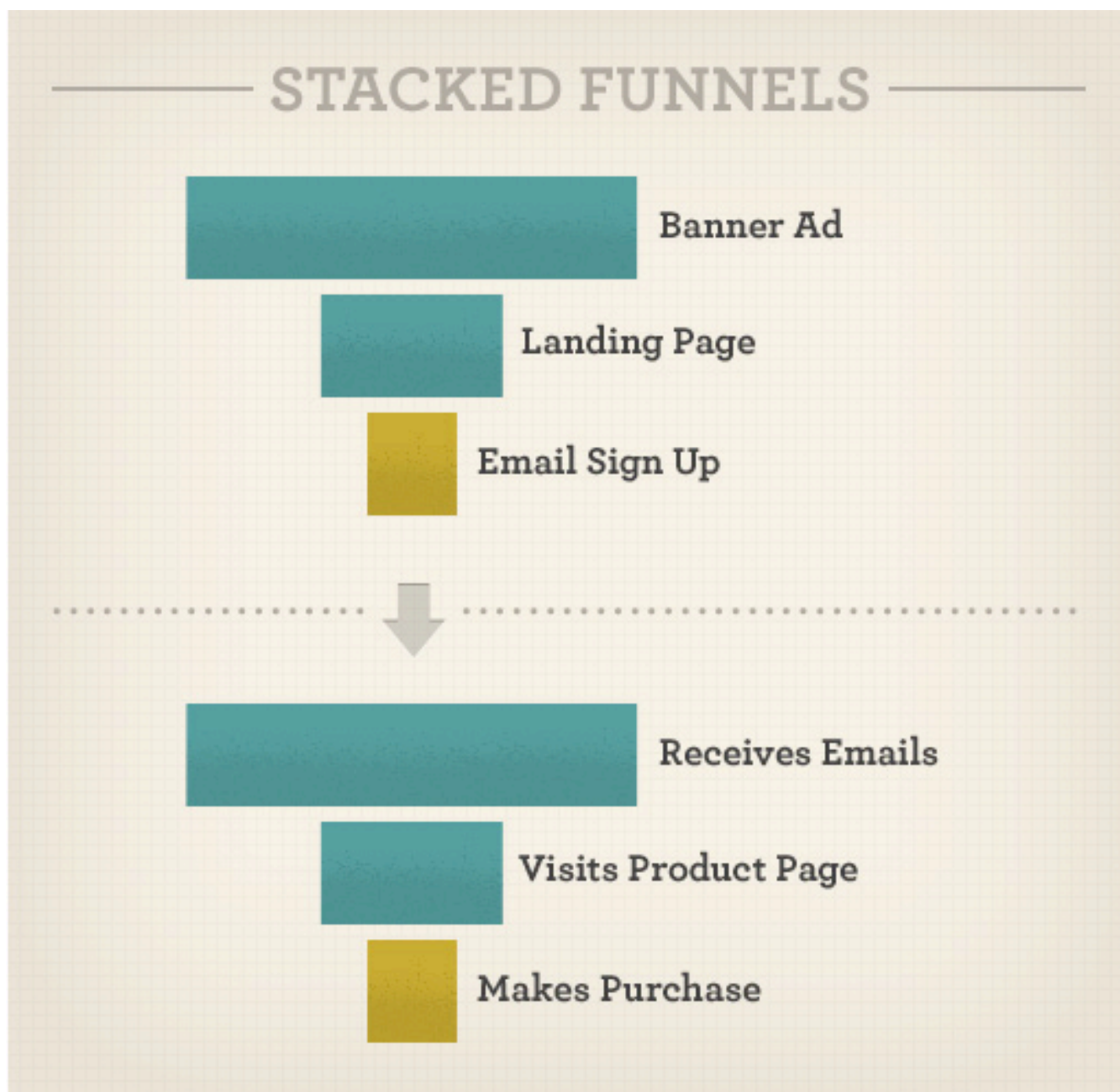
KISSmetrics' "[Anatomy of a Perfect Landing Page](#)" details the design, UI and copy elements that can help you meet your users' need and drive conversions for your business.

Stacking Flows For A Complete User Experience Life Cycle

While viewing a funnel as something like *Click on banner ad → Land on Web page → Register email* is easy, designing and building stacked flows that drive the business' ultimate objectives takes a bit more thought. In our example, we've successfully gained an email subscriber from the banner ad campaign, but the real business objective is to generate revenue through new purchases.

Treating the email subscriber flow and the e-commerce purchasing flow as two separate conversion funnels is an easy trap. In reality, the experiences are connected, and by looking at them as stacked flows, we can create a more cohesive interaction, one that drives optimal results for the business.

In our example, this stack is made up of the customer acquisition funnel and the customer relationship management (CRM) flow.



Stacking funnels creates a cohesive user experience life cycle.

When designing this flow, you need to consider what the biggest levers are for converting the subscriber into a buyer. Many of the earlier principles apply, but this time you have more [touch points](#) to consider and leverage.

In this flow, you need to look at all elements of your CRM strategy and the purchase flow of your website, including:

- Email communication back to the subscriber,
- Pages that the subscriber lands on when returning to the website,
- The flow from internal content pages to check-out.

Here are a few key considerations when designing the flow from subscriber to purchaser:

- Tell a visual story that the subscriber can identify with and wants to be a part of.
- Ensure that your emails reinforce the story, and give proof points to remind users why they subscribed.
- Include compelling calls to action to give the subscriber an opportunity to relate to and be a part of the story.
- Include prominent calls to action and easy, direct paths to the check-out process from the website's internal content pages and blog posts. These validate the user's hope about their role in the story.
- Make the check-out process as frictionless as possible, and reinforce confidence along the way to help the buyer commit to being a part of it.

By considering how the two flows interact, you can create a seamless experience that builds confidence and deepens the user's connection to your website, leading to the ultimate purchase conversion. Equally important, this flow also increases customer satisfaction because the stacked funnels keep the user experience smooth and on track to meeting their needs, with little confusion or ambiguity.

Putting Flow Design To Work For You

Whether you're mapping out a brand new website or looking to optimize an existing user experience, flow design will keep you out of the trap of designing individual pages and interactions and instead focus you on fulfilling users' needs. By prioritizing your user flows and focusing on the ones that drive the most value to the most users and to the business, you can make the greatest impact with your initial flow design.

When considering user flows, think past the first conversion, and design for the ultimate conversion, which might lie a few steps behind. This is particularly important with any type of commerce-driven business, for which the first conversion is often just a prelude to the primary revenue event. By stacking these complementary funnels, you create a more cohesive user experience that drives better results for both the user and your business.

So, the next time you're asked to create a new design, step back and ask yourself and your team what user flows you are trying to create through the website, and let that insight drive the design process.

The Data-Pixel Approach To Improving User Experience

Rian van der Merwe

There are many ways to skin a redesign (I *think* that's how the saying goes). On a philosophical level, I agree with those who advocate for realigning, not redesigning, but these are mere words when you're staring at a design problem in the face with no idea where to start. This article came out of my own questions about how to make the realignment philosophy practical and apply it to my day-to-day work — especially when what's needed is more than a few tweaks to the website here and there.

I propose an approach to redesign *through* realignment, by using a framework adapted from Edward Tufte's principles on the visual display of quantitative information.

But first, a little context.

Redesign Through Realignment

Let's recap the redesign versus realign argument. Here is Cameron Moll in "[Good Designers Redesign, Great Designers Realign](#)":

The desire to redesign is aesthetic-driven, while the desire to realign is purpose-driven. One approach seeks merely to refresh, the other aims to fully reposition and may or may not include a full refresh.

A realignment can include a full refresh, but the starting point is not the visual layer. The starting point is an understanding of the website's users and their objectives, of market trends and of brand strategy. These are the hard questions that guide a realignment — not a desire to try out some new fonts or see whether a +1 button would look good on the home page.

But surely a visual refresh can be beneficial, too? What's the danger in giving users something new to look at? In an essay that builds on Cameron's article, Francisco Inchauste [sums up the problem as follows](#):

Great designers adjust an existing work with little disruption of the foundational design for a goal or purpose. The end result is a modification to the design that improves the user experience. Good designers, on the other hand, recreate existing work focusing on the aesthetic, with a misunderstood notion that it will always improve it. However, they end up disrupting and/or damaging the user's experience, making no real impact with the effort.

The main problem with big redesigns, therefore, is that, even though objectively the UX might have been improved, users are often left confused about what has happened and are unable to find their way. In most cases, making “steady, relentless, incremental progress” on a website (to borrow [a phrase of John Gruber](#)) is much more desirable. With this approach, users are pulled gently into a better experience, as opposed to being thrown into the deep end and forced to sink or swim.

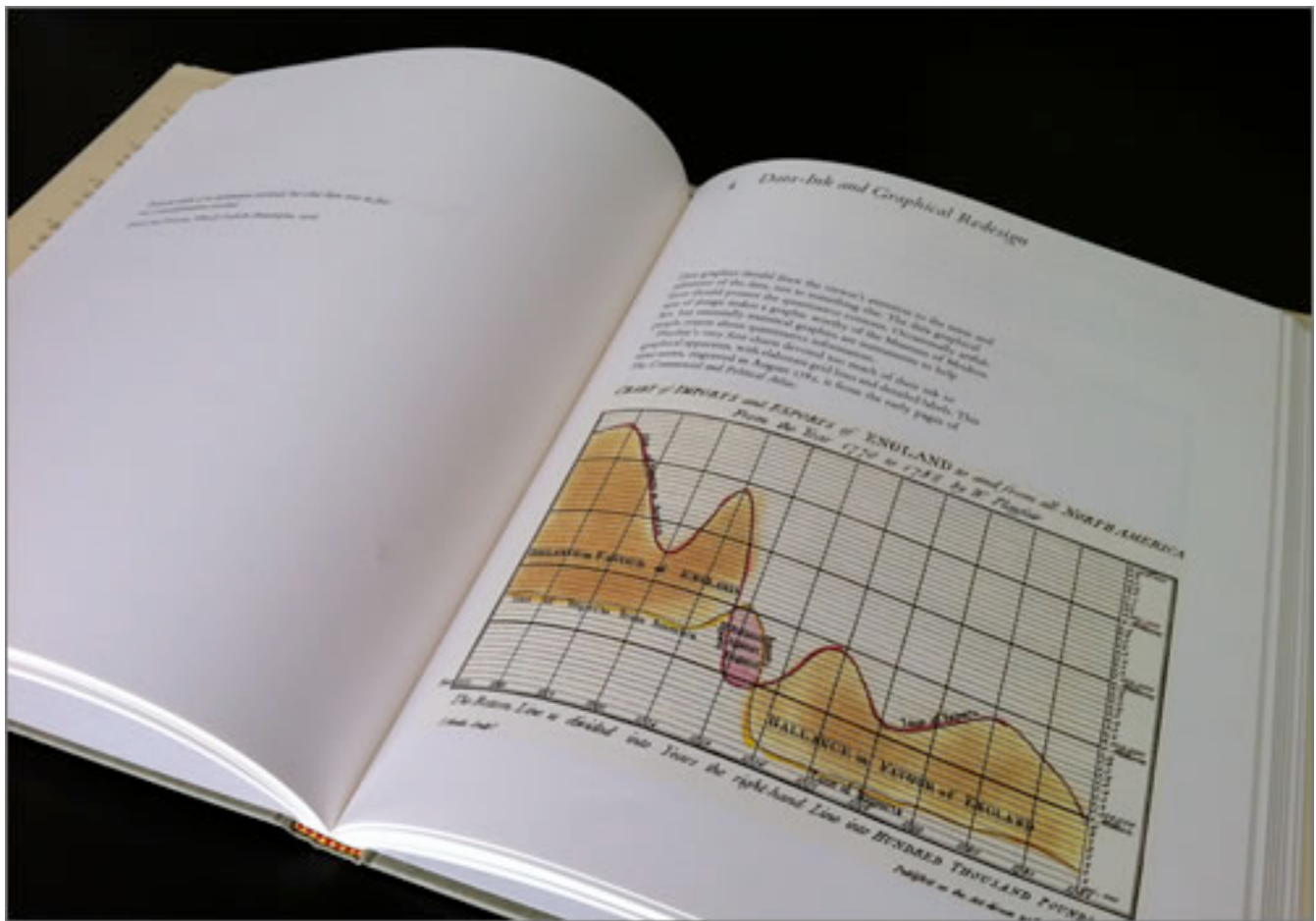
So, if we agree that a realignment is preferable to a redesign (and I'm sure we never will, but let's assume we do for the sake of this article), a big question remains unanswered: What happens when a realignment requires major changes to the website? What happens when small tweaks aren't

enough, when a website's UX is so far gone that you're tempted to scrap everything and start over?

One way to go about it is to use a continual realignment process to redesign the website. Build a vision, and know where you're going in the long term; but get there incrementally, not with a big-bang release. Remaining rooted in the realignment approach ensures that the focus remains purpose-driven, even if the process results in major visual changes. "That's fine," you say, "but how do you do it? Where do you begin with such a project?" Let's now look at one possible approach to redesigning through realignment.

Edward Tufte And The Data-Ink Ratio

I've always been intrigued by Edward Tufte's principles for visualizing large quantities of data. [*The Visual Display of Quantitative Information*](#) is one of my favorite books of all time. Recently, I've been wondering whether its principles could be applied to Web design and, specifically, the realignment process. The deeper I got into it, the more I realized that within Tufte's principles lies a goldmine for people who make websites.



One of Tufte's central principles in *The Visual Display of Quantitative Information* is what he calls data-ink:

Data-Ink is the non-erasable core of a graphic, the non-redundant ink arranged in response to variation in the numbers represented.

Before we unpack what this means for Web design, it's important to note that Tufte's work applies specifically to information graphics and the display of quantitative data, not to the design of graphical user interfaces. However, when carefully interpreted and applied to the field of Web design, the principles are extremely useful.

With that in mind, I propose the concept of data-pixels for the design of user interfaces to mirror Tufte’s data-ink for information graphics. In the context of Web design, we can then think of data-pixels as the simplest and most desirable path that a user can take through a flow (the “non-erasable core of an interface”). It is what would remain in “[focus mode](#)” — if nothing else existed on the screen, just the design elements that enable users to get from one screen to the next.

For example, on a payment screen, data-pixels would be the credit-card fields, text labels and “Pay now” button. Nothing else. This is obviously not possible — you need headers, payment summaries, tooltips, trust seals, etc. But the “core data” are the elements of the page that we cannot remove without the user getting stuck with no ability to recover.

From this irreducible point of the design, you can start to add other elements as necessary, and this is what Tufte’s work is all about. Much of what Tufte espouses is finding the right data-ink ratio (or what we’ll call data-pixel ratio) for quantitative data, so that the core data can shine through. He lays out five principles for data-ink. Here is an overview and how these principles can be adapted to redesign through realignment.

PRINCIPLE 1: ABOVE ALL ELSE, SHOW THE DATA

Data graphics should draw the viewer’s attention to the sense and substance of the data, not to something else.

– Edward Tufte, The Visual Display of Quantitative Information

During a realignment, we should be guided by the principle that every page should be focused on the core data and the primary task that users need to

take in that particular flow. Anything else is noise and should be added only after very careful consideration.

Craig Mod has written a great article titled “[The Shape of Our Future Book](#).” In it, he describes the “quiet confidence” that a Kindle has when it is woken from its sleep state (compared to the iPad in particular), and then he addresses the data-pixel issue as follows:

I think the same concept of “quiet confidence” can be applied to data. Namely, in designing user experiences, we need to produce data that doesn’t draw attention to itself explicitly as data.

This doesn’t mean that design has to be boring or that aesthetics are not important. It means that we need to be mindful that any layer of design we add to the core data has to serve a specific function and cannot distract from the data itself.

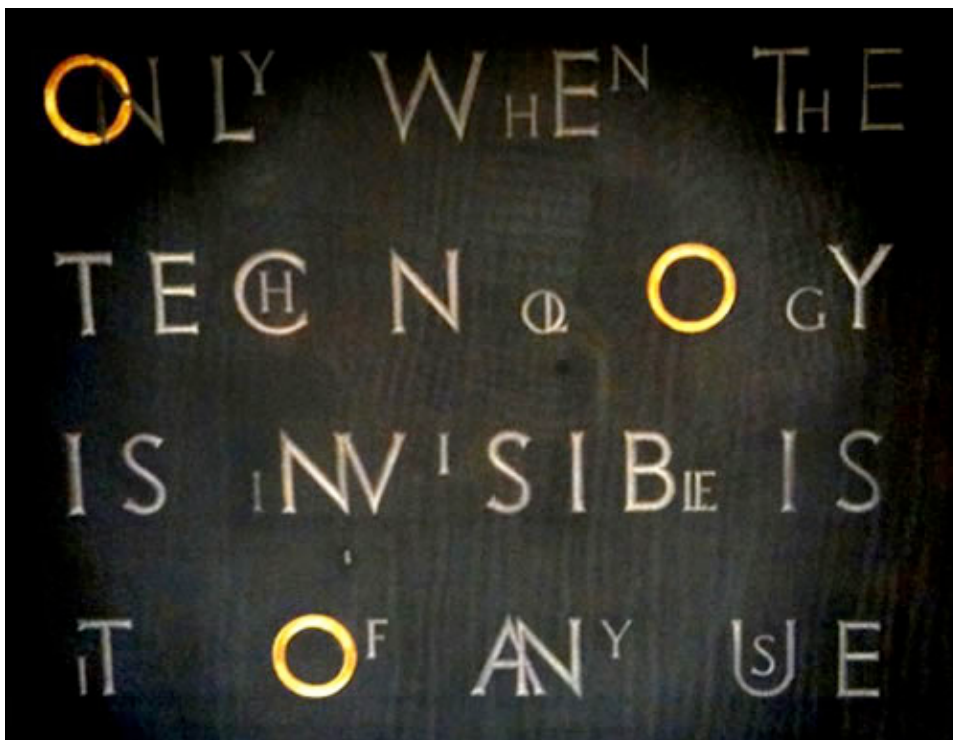


Image credit: [Sarah Blake](#)

PRINCIPLE 2: ERASE NON-DATA PIXELS, WITHIN REASON

While it is true that these boring [pixels] sometimes help set the stage for the data action, it is surprising [...] how often the [pixels] themselves can serve as their own stage.

– Edward Tufte, The Visual Display of Quantitative Information

Once the context for the realignment is set by the overarching principle of focusing on core data, it's time to evaluate the design and start improving the data-pixel ratio.

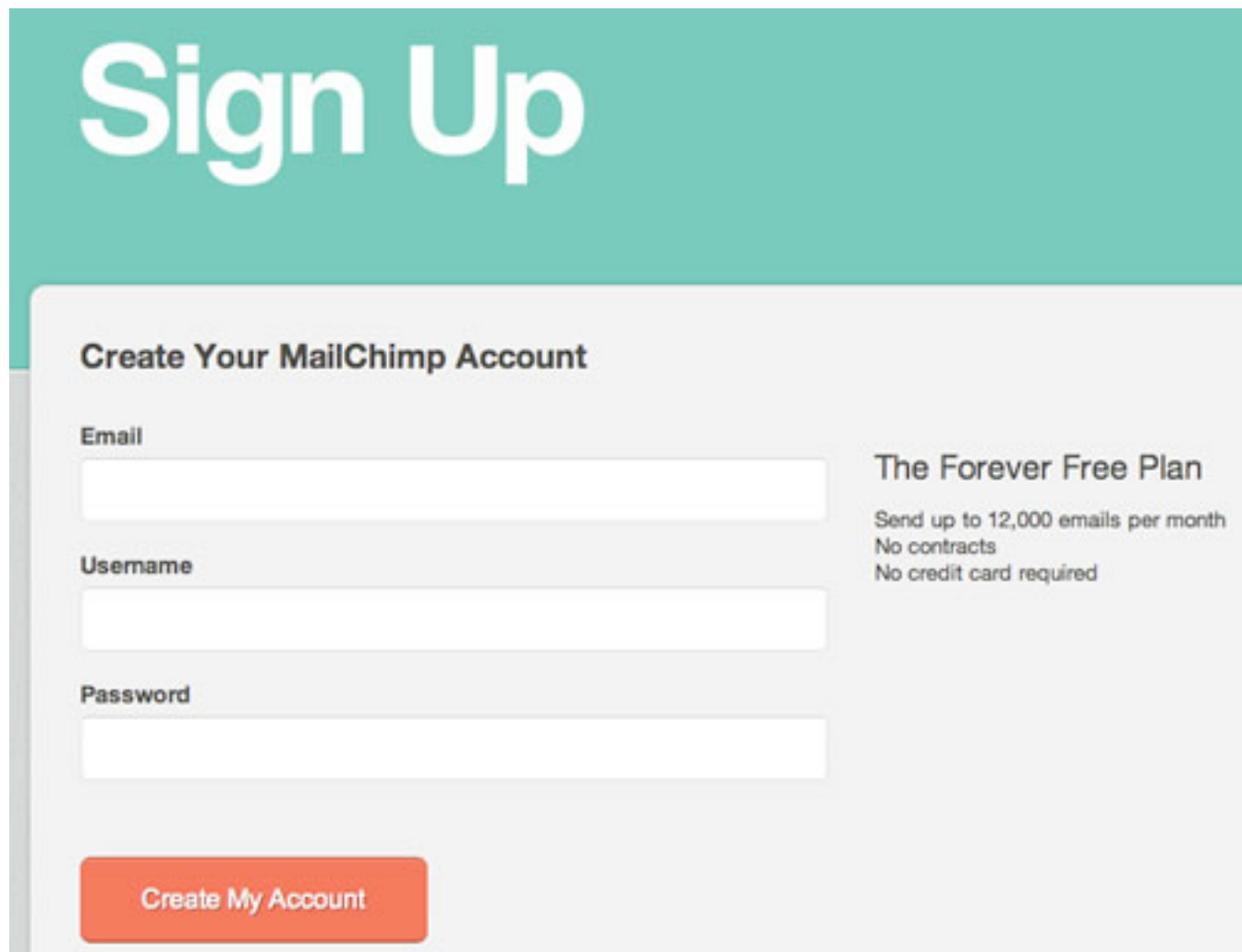
The first step is to look for ways to erase non-data pixels — the parts of the design that don't directly apply to the user's primary task. Look for elements that cannot be connected to guiding a user to the desired outcome, such as:

- Colors that don't support the visual hierarchy through contrast;
- Typefaces that draw attention to themselves for no good reason;
- Gratuitous imagery (stock photography in particular) that does nothing more than break up the page (consider using white space and proximity grouping to create natural breaks instead);
- Alignment, sizes and color contrast that draw unwanted attention (for example, indented paragraphs, large social-media icons, a bright ad that doesn't fit the website's visual style).

One website that could certainly benefit from the removal of non-data pixels is [doHome](#). Notice the gratuitous reflection of the navigation menu, the unnecessary stock photography, and the fact that so little content on the page tells you what the website is about:



[MailChimp](#) does a good job of limiting the non-data pixels on its sign-up page. Clear language indicates what to do; large fields with top-aligned labels ensure that you complete the form quickly; some information is included about the plan you're signing up for; and the button to sign up is big and high-contrast.

The image shows the MailChimp sign-up page. It features a teal header with the text "Sign Up" in large white font. Below this is a light gray form area. On the left, the heading "Create Your MailChimp Account" is followed by three input fields labeled "Email", "Username", and "Password". To the right of these fields, the text "The Forever Free Plan" is displayed, followed by the details: "Send up to 12,000 emails per month", "No contracts", and "No credit card required". At the bottom left of the form area is a large orange button with the text "Create My Account".

Sign Up

Create Your MailChimp Account

Email

Username

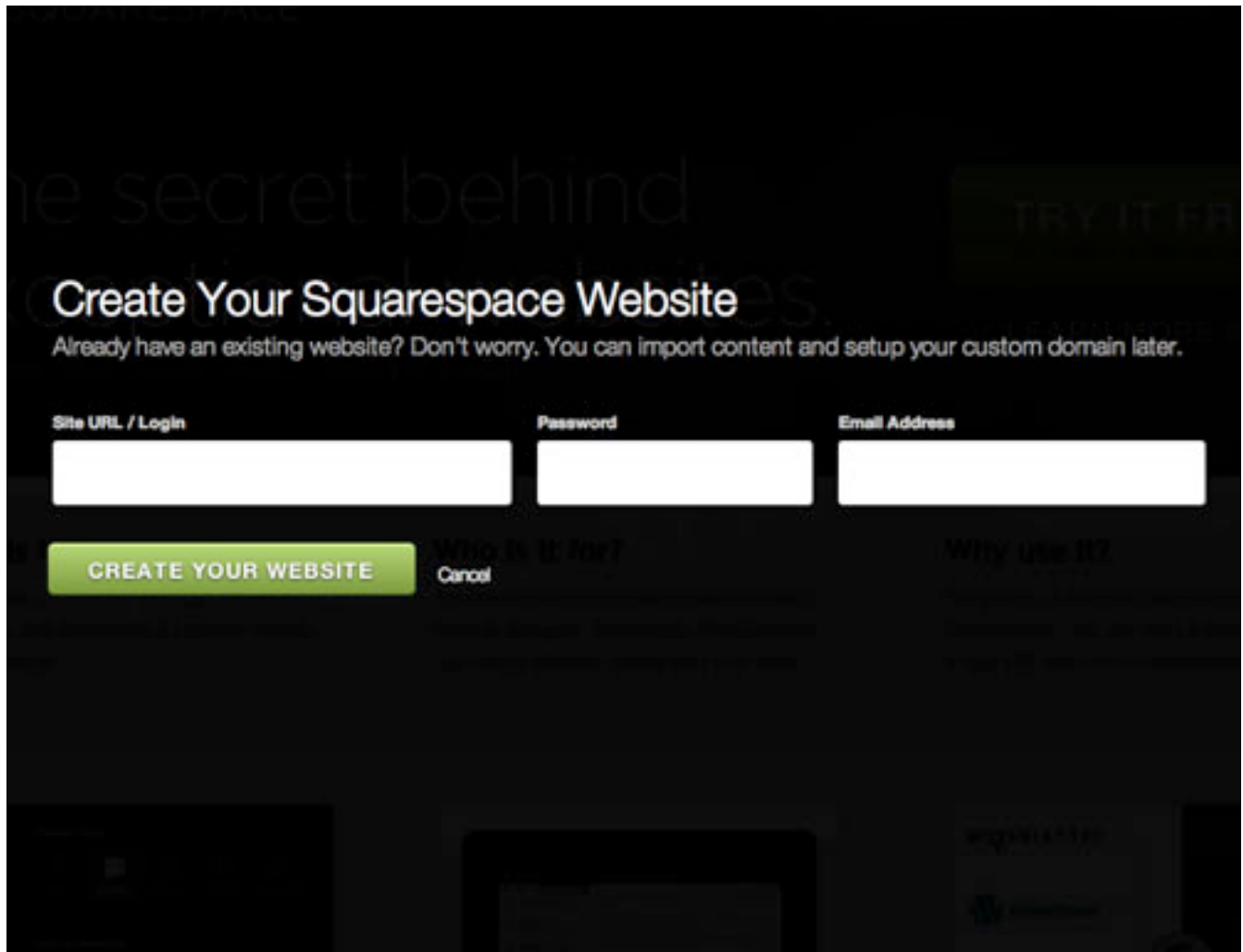
Password

The Forever Free Plan

Send up to 12,000 emails per month
No contracts
No credit card required

Create My Account

[Squarespace](#)'s sign-up page removes as much non-data pixels as possible to put all the focus on the task at hand. The background disappears as soon as you click the “Sign up” link, and you’re presented with the following zero-distraction form:



The screenshot shows a dark-themed sign-up form for Squarespace. The background is a dark, textured gray. The main heading is "Create Your Squarespace Website" in a large, white, sans-serif font. Below it, a smaller line of text reads: "Already have an existing website? Don't worry. You can import content and setup your custom domain later." The form consists of three white input fields arranged horizontally, labeled "Site URL / Login", "Password", and "Email Address". Below the "Site URL / Login" field is a green button with the text "CREATE YOUR WEBSITE" in white. To the right of the button, there is a link that says "Who is it for?" and a "Cancel" link. The bottom of the form features a row of four small, dark rectangular boxes, each containing a logo or icon.

Google+ is another example of a design that employs minimal non-data pixels. As [Oliver Reichenstein says](#):

It is extremely difficult to keep a complicated user interface so light, white and free of lines, boxes and ornaments. The content hierarchy is always clear, color definitions are consistent and clear without labeling them.

This list isn't exhaustive, but it illustrates the purpose of the principle: to critically evaluate the visual elements in order to strip out what isn't necessary.

PRINCIPLE 3: ERASE REDUNDANT DATA-PIXELS, WITHIN REASON

Gratuitous decoration and reinforcement of the data measures generate much redundant data-[pixels].

– Edward Tufte, The Visual Display of Quantitative Information

“Redundant data-pixels” refer to elements of the design that are repeated without good reason. Some examples:

- Rows of products in a table with an “Add to cart” button next to each one.
(Consider using check boxes, with one “Add to cart” button at the bottom.)
- Animation as a way to draw attention to an element.
(Consider using high-contrast color and size instead.)

- “Are you sure you want to do this?” dialogs for simple tasks such as adding a product to a cart.
(For potentially catastrophic actions, like deleting an account, this type of dialog is, of course, appropriate.)

Below is an example from the offline word, courtesy of [Allan Kent](#). The chart shows the price of parking per hour. Surely a simple sign that reads “R10 per hour” would suffice?



PARKING TARIFFS	
VISITORS PARKING	
1 hour	R10,00
2 hours	R20,00
3 hours	R30,00
4 hours	R40,00
5 hours	R50,00
6 hours	R60,00
7 hours	R70,00
8 hours	R80,00
9 hours	R90,00
10 hours	R100,00
LOST CARD	R100,00
<small>ENTRY INTO THESE PREMISES AND THE USE</small>	

Note the addition of “within reason” to each of the data-pixel principles. Tufte himself acknowledges that redundant pixels are sometimes necessary:

Redundancy, upon occasion, has its uses: giving a context and order to complexity, facilitating comparisons over various parts of the data, perhaps creating an aesthetic balance.

The exception, then, is when, in Tufte’s words again, “redundancy has a distinctly worthy purpose”. A “Pay now” button at the top and bottom of a check-out page could be an example of this. One of the buttons is redundant, yet it introduces efficiency so that users don’t have to scroll up or down to place their order.

The guiding principle here is to strive for a minimalist aesthetic, adding redundant pixels only when they serve a larger purpose (for example, when they’re essential to the brand’s promise or to user efficiency).

PRINCIPLE 4: MAXIMIZE THE DATA-PIXEL RATIO, WITHIN REASON

Every [pixel] on a graphic requires a reason. And nearly always that reason should be that the [pixel] presents new information.

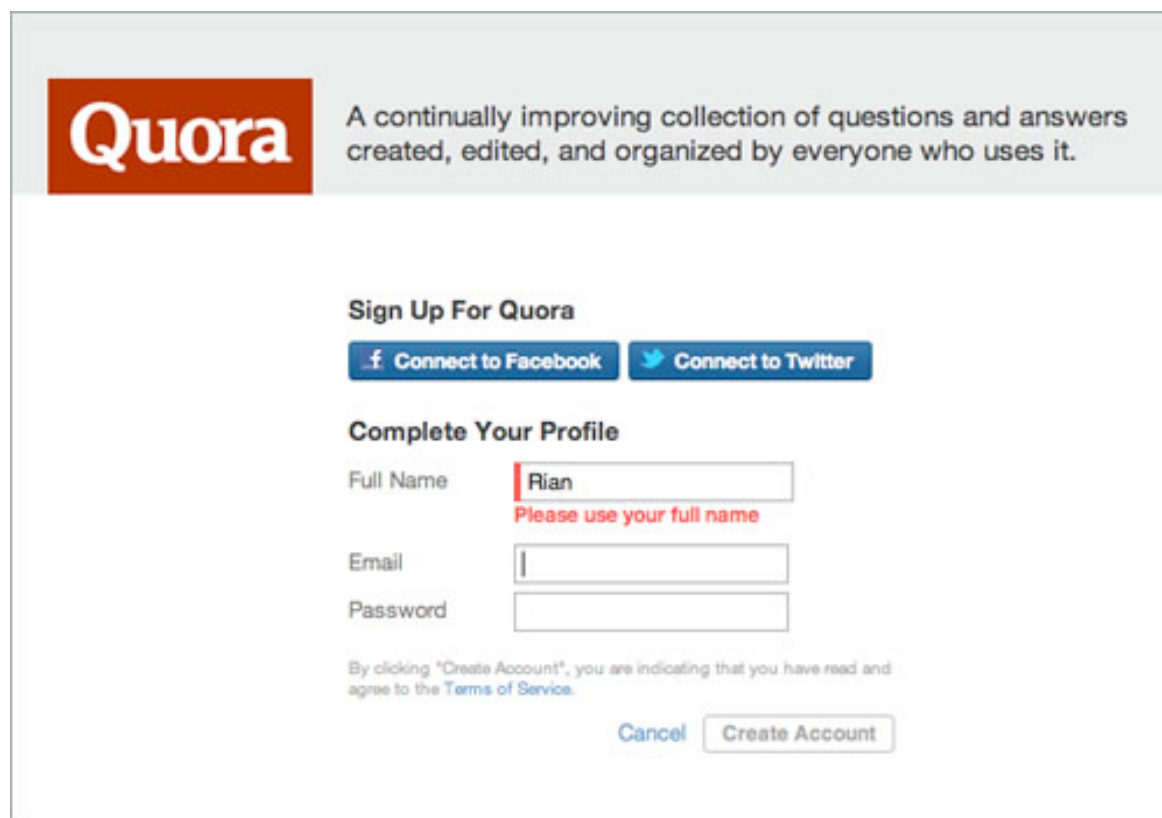
– Edward Tufte, The Visual Display of Quantitative Information

Once you’ve erased as many non-data pixels and redundant data pixels as possible, the next step is to figure out what (if anything) is missing from the design. The goal of this principle is to add more pixels to the design, if necessary.

There **should always be a reason** for adding elements to a design, and the reason will usually be that those elements provide information and/or functionality that increases usability. Some examples:

- Breadcrumbs that tell users where they are on the website and give them an easy way to get back to where they came from.
- An aesthetic layer of color, typography, layout and so on, to ensure consistency between brand perception and the website.
- Hover states or tooltips to provide appropriate guidance or contextual help to users.

An example of necessary pixels is relevant inline error messages. Consider the sign-up form on [Quora](#) below. Very few non-data pixels are on the page. The form's layout is simple, with no extraneous decoration. But if you try to enter only your first name, the page instantly reminds you that a full name is required:



The screenshot shows the Quora sign-up page. At the top is the Quora logo and a tagline. Below is the 'Sign Up For Quora' section with social media connection buttons. The 'Complete Your Profile' section contains input fields for 'Full Name', 'Email', and 'Password'. The 'Full Name' field contains the text 'Rian' and has a red error message 'Please use your full name' displayed below it. At the bottom are 'Cancel' and 'Create Account' buttons.

Quora

A continually improving collection of questions and answers created, edited, and organized by everyone who uses it.

Sign Up For Quora

Connect to Facebook Connect to Twitter

Complete Your Profile

Full Name
Please use your full name

Email

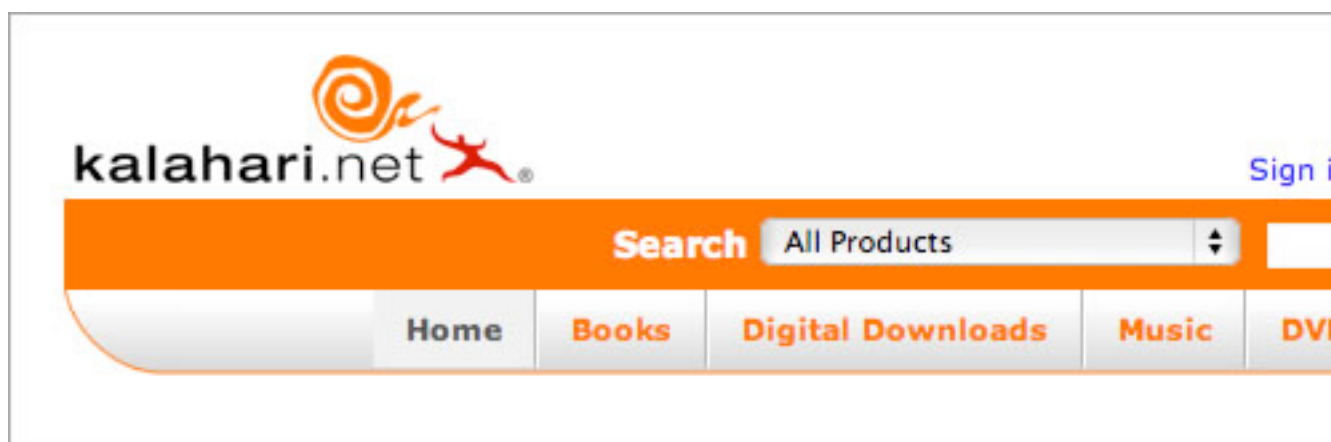
Password

By clicking "Create Account", you are indicating that you have read and agree to the [Terms of Service](#).

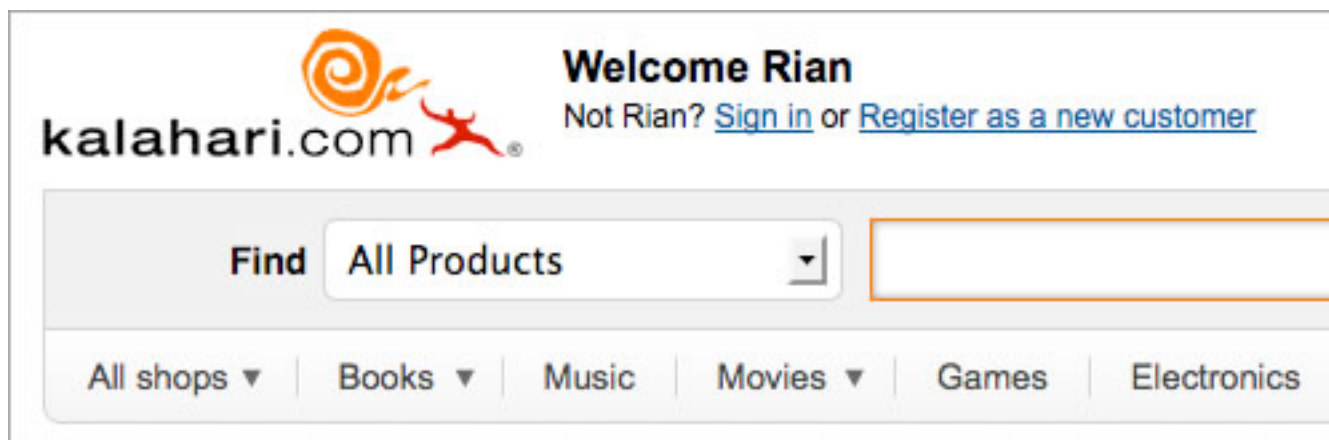
Cancel Create Account

One could argue that this isn't technically part of the "core data" of the design. Quora could have let this slide and either allowed accounts with first names only or sorted it out after users have signed up. But it has decided that data integrity is important from the start, so it has added this real-time check.

A little closer to home, this is what the header of kalahari.com (where I currently work) looked like when I started at the company:



Identifying the non-data pixels in this design is easy: very large radii on the rounded corners, color that grabs too much attention, too many unimportant links, etc. After maximizing the data-pixel ratio to put the focus on the core data (which is search), we ended up with this header:



PRINCIPLE 5: REVISE AND EDIT

Probably, indeed, the larger part of the labour of an author in composing his work is critical labour; the labour of sifting, combining, constructing, expunging, correcting, testing: this frightful toil is as much critical as creative.

– T.S. Eliot, *Selected Essays, 1917–1932*

Tufte quotes T.S. Eliot to describe the relentless effort of editing and revising in graphic design work, and it's certainly true for Web design.

Once you've completed a cycle through these principles, it's time to go back and start again. Every realignment cycle exposes new opportunities to “above all else, show the data”. UX is a never-ending cycle of improvement, and following a realignment process bakes this constant cycle into the strategy in a very natural way.

Summary

As I mentioned at the beginning of this article, there are many approaches to redesigning a website. The hardest part often is knowing how and where to start. As I've shown, Edward Tufte's timeless principles for the visual display of data can be adjusted and used as a framework to get over that initial hump and serve as a catalyst for a cycle of continual improvement through realignment.

What other models or approaches are there to frame a realignment project? How do you get started?

About The Authors

Dave Shepard

Dave Shepard is the Lead Developer and Architect at digital-telepathy specializing in application architecture, user interaction, interface development and technology. Dave is the lead developer of the SlideDeck WordPress plugin and Hello Bar the website notification bar.

Leigh Howells

Leigh is a designer with 15 years experience, now working in user experience. He's been helping websites look better, be more organised and work better since grey backgrounds were the norm. He's a jack of all trades, from video to music and still trying to master at least one of them. He survives on coffee and custard creams and blogs occasionally from his own planet; leighhowells.com.

Luke Wroblewski

Most recently [LukeW](#) was the Chief Product Officer (CPO) and co-founder of [Bagcheck](#) which was acquired by Twitter Inc. in 2011. Luke is also the author of the upcoming book [Mobile First](#) and was Chief Design Architect (VP) at Yahoo! Inc.

Chuck Loganecker

Chuck Longanecker is the CEO and co-founder of digital-telepathy, a [user-experience design company](#) passionate about creating products, like

[SlideDeck](#), [Hello Bar](#) and [Impress](#), that make the Web more intuitive and compelling.

Shell Greenier

Shell is a compulsive adventurer, gardener, sender of happies & seeker of *swoon*. She enjoys meeting new people on twitter so go meet her [here](#).

Rian van der Merwe

Rian is passionate about designing and building software that people love to use. After spending several years working in Silicon Valley, he is currently Head of Product & User Experience Design at South African online retailer [Kalahari](#). He also [blogs](#) and [tweets](#) regularly about User Experience and Product Management.

Tyler Tate

Tyler Tate is a user interface designer at Vyre in London. He is the creator of the 1KB CSS Grid, and his latest venture is TwigKit.