

## Why We Prototype

[Web Smart Newsletter](#) by Eric Holter, January 2007

### Why We Prototype

The year 2000 was a pivotal one for Newfangled. We began work on what has become the [NewfangledCMS](#), and it was also the year we first started [grayscreen prototyping](#). While our CMS is the most appreciated by our clients, grayscreen prototyping is really the heart of how Newfangled works. In fact, I would go so far to say that without grayscreen prototyping we would have gone out of business well before our CMS ever had the chance to mature.

This month's Web Smart newsletter digs into the reasons why we prototype. And why we couldn't get along without it.

Last week I was perusing my RSS feeds and saw a interesting posting from A List Apart, a web design and development blog. The posting was titled "[Paper Prototyping](#)." The article described a low tech means of website definition using paper. These paper prototypes utilize manila file folders. Main site sections are hand written on the tabs. Text clippings are glued to pages to describe content. Handwritten notes detail features and functionality. Buttons are cut out of paper and glued to the folders. Retro.

Another feed I subscribe to recently posted an article called "[Don't make the Demo look Done](#)" which observes that mock-ups should not be too detailed. Rather, pencil sketches can be a better means of communication than refined Photoshop layouts.

Each of these postings pointed out that counterintuitive methods for communicating about technology, using explicitly non-technical means, can work much better than technically detailed methods.

Here are some specific observations from these two articles...

1. "...many people are intimidated by a formal, highly technical design process..." [Paper Prototyping].
2. "The prototyping stage is the right time to catch design flaws and change directions, and the flexibility and disposability of paper encourages experimentation and speedy iteration" [Paper Prototyping].
3. "Show them something polished and pretty, and you'll get feedback on font sizes. The reviewers make incremental tweaks, blinded by what's in front of them. But show a napkin sketch, and they don't just see what's there, they see what's possible" [Don't Make the Demo Look Done].

I completely agree with these insights and I commend the efforts these developers have made to do whatever it takes to find ways to effectively communicate about the web with their clients. But ironically, I have found that paper based documents, by virtue of their being paper, create other occasions for misunderstanding and don't go nearly far enough to communicate the subtleties of a website's content, structure, and functionality. Paper is part of the problem in web communication. The fact that paper cannot be clicked, linked, and viewed in a browser is a fundamental limitation as a means of website documentation. There is just something about the act of clicking that facilitates the process of understanding how a website will work.

Nevertheless, their observations about the problems with technical specification are real. We made many of the same observations back in 2000, which lead us to experiment with new ways to communicate about our website projects. The approach we discovered, however, was very different.

## Why Paper Documentation Doesn't Work

I actually have a grudge against the use of paper for planning, describing and defining a website. Paper, while helpful as far as it goes, ultimately fails because it leaves too many gaps in understanding the subtle dynamics of a website. Whenever I talk about our process I always quote George Bernard Shaw who said "The biggest problem in communication is the illusion that it has taken place." Any incomplete communication process is subject to the biggest problem of all - the illusion that communication happened. We are in the most danger when we assume we've communicated. Paper prototypes give a false sense of completion. As a result we move forward with a degree of unfounded confidence, having used so much paper. But it's not until a project emerges from development, when the client can click around, that these gaps reveal themselves--usually too late for an easy fix.

No doubt a careful balance needs to be struck between being too technical and being incomplete or irrelevant. Our solution was to document the website using a website. We started using visually generic "grayscale" HTML prototypes. The problems of using too refined or polished a tool in the planning stage, problems highlighted in the two recent postings, were still a potential problem. But we found that as long as the prototype was intentionally gray and generic (even ugly) our clients knew that they were not dealing with a final product and that their ideas, questions, and changes were welcome.

I remember the moment when I first realized we were on to something with grayscale prototyping - *hey, why is everything getting blurry and wavy* - ah yes, it's the going back in time transition, nice...

## An Effective Website Development Methodology

It was Y2K and the computers of the world had not imploded. The web world was going strong. The dot com bubble burst was still in its nascent stages. We had two very large projects which both needed content management capabilities. We decided to invest extra time to build a CMS that would meet the needs of both sites and that we could reuse. This became the NewfangledCMS but that's another story. This story concerns the deep dread I felt at that time, going into these two large complex sites that would be using, by far, the most complex technology we'd attempted up to that time. If what I had experienced to date, using paper documentation, held true in these two complex sites, I would soon be in a world of hurt. Even though our paper prototypes were very detailed and thorough, and though our clients had approved them (many times, over many meetings, lasting many hours) I knew that it was the things they could not see in the paper that would come to light in the beta release. I was prepared, and had even planned for some pain in this regard, but how painful would it be? Sadly, past trends confirmed that the more complex a site the more questions/problems/changes/ideas we would get after the client could click through a programmed beta site.

In anticipation of this reality, after the clients had signed off on the paper prototypes, but before we started coding, I asked one of my developers to re-create the paper document using very generic unformatted web pages. The two documents, one paper, one HTML were basically identical and showed all the same information and all the same pages. We asked each client to review their web based HTML prototypes. And guess what. They had lots of questions/problems/changes/ideas that they did not have even having signed off on the paper. Happily, we had not started coding yet, and we could address these concerns before we began. I sure was glad we did that. What I didn't know was whether or not this would make any difference when it came to the final sites.

There's something you need to know about website project schedules. Every project starts off with an intended start date. Very few hit them, almost always due to content issues so that it's nearly impossible to predict exactly when a site will go live, especially large complex sites whose initial schedules stretch across a six month period. So how was I to predict that both of these large complex sites, using our brand new technology, employing a brand new process would both end up going live on the same week? And how was I to control that this would be the very week I was going on vacation to visit friends on the other side of the globe? I was out of the country on the day these two complex sites went live. That morning, in another country, I woke up with a feeling of dread. I desperately did not want to check my email. When I did, I was certain my vacation would be over. But it had to be done...

**Inbox (1)**, From: Newfangled, To: Eric. "Hi Eric. The client called and just wanted to say how happy they are. The site is doing everything they wanted and more. There were a few minor bugs that have already been fixed. Hope you're enjoying your vacation."

I had to read that again. This wasn't possible. Web projects, certainly not ones this complex worked this way. Maybe this was a joke and

the real email was still coming.

## The Secret to Effective Website Development

As it turns out both sites went live and were a great success. At first we were just using grayscreen prototypes for our most complex sites but we started doing it even for simple sites. To our amazement even the most basic sites benefit greatly from grayscreen prototyping. The more we did it the better our projects became, and the more our clients were satisfied. I started noticing other benefits, like when our clients reviewed visual site layouts after working through the grayscreens, they tended to approve designs with fewer rounds of changes, and sometimes even after one presentation. Wow. I guess you have to be a web designer to fully appreciate that, but believe me it's amazing.

We also began to recognize that certain intangible dynamics were changing. Sadly, web projects tend to disintegrate, sometimes badly. Many of our clients have been through one or more bad experiences before coming to us. We have great patience with new clients when they exhibit a cautious or even suspicious demeanor. It's entirely understandable that they would be uncertain about whether their experience with us will really be any different. When communication starts off on such a footing it can be hard to overcome. But throughout the prototyping process, during which all of a client's questions are answered, ideas explored, and their thoughts listened to, the relationship warms way up. This back and forth, iterative process, where they see and experience their website ideas in a clickable form, draws people in. Thus an upward spiral moves the project from uncertainty to confidence in a very short time.

I often ponder what our clients think about Newfangled and our work. We get lots of feedback about our CMS, they really like how easy it is to update and add content to their websites. Most of the testimonials we get focus in on the final site and the ability they have to maintain it. But I know that while the a great website is ultimately what we all want to produce, it's really this subtle communication technique that is the path to getting there. After all we're not magicians. We can't build a good site in a vacuum. We can only respond to what our clients want and need, and help them refine their ideas. The prototype just helps us draw that out. I mean, how beneficial is it really to be able to easily maintain a website you hate? If the site is bad, maintaining it is kind of moot.

**More information:** We recently updated our book [Client vs. Developer Wars](#) which documents in much more detail the discovery and details of grayscreen prototyping. The book can be read in its entirety on this site or download as a PDF. Also our video "Avoid the Illusion of Communication" provides an overview of our process (click on that image over on the right side bar).

But the fact that we can get down to it using grayscreen prototypes, that fact that our clients really get what we're talking about in the planning stages (because they can see it and click it), that fact that we're not being fooled by the illusion of communication means that the final site will be right. I credit our surviving the dot com bubble burst and the 9/11 website blackout to our grayscreen prototyping. While I really enjoying seeing our sites go live and looking at the final, fully designed result, I quietly appreciate and enjoy the fact that it really all happened because of something so subtle and simple as communicating about a website using a website.