

Introduction

So for the next couple of days, I'm at [An Event Apart in Chicago 2007](#), and *this is the first day!* I'm very excited about seeing all the guys that taught me web development:

- [Jeremy Keith](#)
- [Jeffrey Zeldman](#)
- [Eric Meyer](#)

The cool part about events like this is, it really gives you a big picture. Every day, you're inside designing and coding, and sometimes you need excitement!

Coming to a big conference with tons of people just like you is a sure way to get fired up about what you're doing. **An Event Apart** is definitely something that will send me back excited about working on the web.

Arriving and Settling In

A big part of any conference is networking. I love meeting new people, and *to be successful you have to be a people person.*

Within ten minutes of getting here I met up with no less than three people, all of whom adore **Jeremy Keith**. It's a gathering of geeks and we all have our personal gods.

We also got some sweet *free* stuff, including a **512mb flash drive**. Apparently I'm supposed to already have one, but the iPod seems to work well enough for file transfers!

The conference is going to be starting soon; keep your eye on the blog, it's going to keep updating as the day goes on!

Kicking it off!

Eric Meyer and **Jeffrey Zeldman** just took the stage, with some humor. Apparently all iphones and phasers should be turned off! Eric Meyer is going to be the first speaker.

Jeffrey Zeldman thinks he's very amusing, and **Eric Meyer's Dad** is here so we're all encouraged to laugh loud!

Eric Meyer - Secrets of the CSS Jedi

This demo challenges what we think we know about browsers, structure, and presentation. Sometimes we get stuck in our method of working and talks like this move us outside the box.

For instance, rather than using separate images for rollovers, why not use **transparent PNG's** and show through the background color on the text?

Think about things you haven't done with CSS before. He talks about how CSS doesn't care how an element is supposed to behave, it allows you define how it will behave.

To use CSS though, must be a certain document structure and type, so structure and presentation can't be entirely separate. What about text documents?

Breaking Tables

The first thing he runs through is how we can do whatever we want with a table layout. For instance, adding `display: block` to a table completely changes how a table renders. This is intense.

What I like about this is that it takes you out of the restrictions of certain tags, certain layouts. Even though a website might be stuck in tables, you can absolutely position all of the elements within the table.

[Interesting things he does with this table data is:](#)

- Position the main `<th>` outside the table.
- Convert each `<tr>` into a full-height column.
- Edit each `<td>` to look like a bar graph with various heights.
- To make it accessible, add **scopes** to each `<tr>` so the blind CEO can listen to the web page.

Reset the Styles

Like everything else in the world, browsers are going to do what they want to do unless you guide it correctly. In fact, each browser has default CSS files in their program directory that sets the margins and padding for headers, random blank lines, colors, etc.

[Check out his reset styles so you have full control!](#) *Create your own reset styles to fit in with your work style.*

One additional point is the CSS *counter* feature, which allows you to add styles based on where you are in the structure. For instance, if there are four `<p>` tags, add a style to the second `<p>` tag.

Jeffrey Zeldman - Writing the User Interface

This talk is about how to lead people through websites with simplicity and by getting the right message across.

Since content drives the web, we need appropriate, clear, simple copy that sends people in the right direction. Also, it needs to be updated regularly. Think about sites like YouTube and Facebook which are the most popular; their content is updated every second of the day!

Good user interfaces

Blogger

To get an understanding of the user interface, set up a new account on [Blogger](#). You encounter simple steps, rudimentary colors, and an arrow to point you in the right direction.

Basecamp

With [Basecamp](#), it takes a complex task like project management and makes it simple and a little bit fun. Zeldman has some fun by saying that their login prompt is like getting a towel slap in the locker room after a good game!

You need brand appropriate copy - if you're preaching simplicity, don't make your text complex. If you preach easy to use, make your website easy to use. It's simple! Don't send mixed messages and don't use wall-to-wall text that's impossible to get through.

Language and reading

It's kind of difficult to read online after a while, so he gives some steps for making that process easier.

- Break text into chunks
- Use headers and lists
- Simplify wherever possible

The less content there is, and if there's semantic markup, then SEO is a breeze. Don't cram copy because people aren't going to read through the keyword stuffing.

Check out [Writing for the Web](#) to get a good idea of what's good and bad.

[Continue to the second An Event Apart post](#)

Continuation

The first post for An Event Apart - Chicago 2007 was getting long. One of the main teachings from Jeffrey Zeldman was to break your content up and not overwhelm people. Hence, the second of several posts.

Any time I feel that things are starting to get long, I'll move to another post. So without further adieu...

Jason Santa Maria - Design Your Way Out Of A Paper Bag

In every situation there's a message, process, etc, and it's our job to learn it. Anything we do without research is bound to fail, or bound to just be lines thrown on the screen.

In putting together your design process, he talks about putting together a set of resources to reference when you're down on ideas. Think sketchbook or collection of photos.

Have fun with design! Push boundaries, explore new ideas, and be willing to be funky sometimes.

This is all about being creative while still following a process, which is something I've struggled with. When it does work correctly, it's quite awesome.

A Site Redesign

- Think of what the site means to people and what they use it for. Don't take the site away from its purpose.
- Start with sketches of logos, text size, layouts, etc. With anything, you need to visualize before acting.
- When designing for yourself, think about your messages. Design should always be based around message.
- Realize that doing redesigns is a great idea. They become stepping stones to better designs.
- In redesigns, what are the old features that aren't desirable? You want to unstick people from unusable features.
- For areas of the site that are desirable or very useful, use the redesign to feature those areas. Site traffic will increase!
- Look for design features that are cluttered or confusing. Keep the interest high for the entire page.

Brand vs Logo

Logos should give a message about the quality of the thing it symbolizes. If you have great food and good service, your logo represents that.

What does a service or site allow people to do?

Wordpress

People can publish with [WordPress](#), they speak their minds, it's somewhere they've been and touched. How do you redesign their logo?

It focuses on typography, it's clean, and it doesn't disturb the image. Is this company still dependable?

Sometimes keeping brand equity is just a matter of updating the font, icon usage, and colors. Don't completely pull the carpet out, people don't like too much change.

Jason takes a lot of influence from [Paul Rand](#).

Setting up Layouts

Think grids. It's how cities are set up, and you know what to expect.

- How is it going to extend? By setting a standard "grid" for each page of the site, you never stray from the intention.
- Utilize white space as part of your message, it adds to how inviting your site is and if people want to stick around.
- Think about design using left to right, top to bottom, big and small. Other cultures interpret these things differently.

Sweat the small stuff

- Borders - solid or dotted? What stands out too much?
- Text color - black or gray? What's a good contrast?

To get into type

- [The Elements of Typographic Style](#) by Robert Bringhurst
- [Thinking with Type](#) by Ellen Lupton

For grids

- [Grid Systems in Graphic Design](#) by Josef Muller-Brockmann
- [Making and Breaking the Grid](#) by Timothy Samara

Question and Answer

Fluid layouts can be tricky - if there's a lot of content, it's better to go with fixed layouts.

Expectations and communication with clients is essential to getting a website complete. Otherwise, you might end up redoing the entire thing!

Lou Rosenfield - Search Analytics for Fun and Profit

I get the feeling that this one is about having the right data for your site when doing a re-design. Lou points out to use your search analytics in your log files to determine the right data.

If you do a good job on the common queries in your content, the people coming to your site adds up. You're taking steps in the right direction.

Also, gear your content towards the time of the day, week, month, or year depending on what the audience is known to look for.

To find out what they're looking for, check out:

- Search logs that captures data naturally
- Search keywords that you capture into your local database
- Keywords or phrases that your commercial search solution captures (like Google Appliance)

When people are navigating and searching, are you returning quality results? What is clicked most frequently? Maybe something needs to be focused on that you didn't even realize!

At this point, stop and ask the big question - why are we building websites? What do people want from a particular site? Don't build sites without purpose or user orientation.

Even if a site is useful, is it confusing anyway? If it is, maybe redirect people to the right areas. BBC is researching how people are clicking through their website and develops personas for age group, regions, etc for different areas of the site.

Identify where people are getting stuck at and enhance search features to reduce the amount of dead-ends that occur on your site.

Sometimes, results are buried but those are the ones getting the most clicks. Ask people why they're clicking there and see if it needs to be featured.

This talk is a little slow, so I walked out into the lobby for a cookie and a chat! More to come...

[Continue to the next An Event Apart post](#)

Liz Danzico - The Seven Lies of Information Architecture

This talk is going to take us back to the basics of setting up a website, and what the basics are not! The basics are essential for anything to succeed, so this information will make people better information architects.

Lie 1: Navigation must always be consistent

The new Apple site

Apple does not have consistent elements because people don't need the macro site, they want the micro sites that get them what they're looking for.

Truth 1: Broad navigation is a positive for the user.

This makes sense. We're so stuck in the box on this topic that we could afford to switch things up. Maybe even go crazy once in a while!

Lie 2: There is a magic number seven (plus or minus two).

People only remember seven things from a site, which is incorrect. People have different memory capabilities depending on how they train themselves mentally.

Truth 2: There is a magic number, but it depends on the user.

Lie 3: Users must get to all parts of the site all the time.

Truth 3: Users must get to everywhere from everywhere.

This is a valid point and accounts for all the places that people might come to your site like Google or if a friend points them there.

Lie 4: People always need to know where they are

You can definitely get too minimalistic, and certain sites don't need to know. Think YouTube - you need to know where all the videos are?

This is smart, I'm prone to put breadcrumbs even on the smallest sites, and always track the breadcrumbs all the way back. Maybe I don't need to.

People don't need to use navigation or breadcrumbs, even just having a search feature or in-context links will get people where they need to go.

Lie 5: The user experience must be seamless

What does seamless even mean? The author of this blog has personally heard "This needs to be more ergonomic" as a comment on a website.

As long as the vision is clear, it doesn't have to be seamless.

Twitter

Although the way it functions is consistent, the interface guidelines are up for grabs. Widgets, through email, cell phones, whatever!

Truth 5: The designer must design beautiful seams.

Design things to be portable, have good structure, good presentation, and allow for interfaces to interact with your designs. Things will work out well.

Lie 6: Shorter is Better

When people are making decisions, longer is often better! I'm always prone to disagree with this, but it might just be a personal preference. Read a lot of content isn't ideal!

Truth 6: Short is better and sometimes long is too.

Lie 7: Information Architects do information architecture

We need to learn to do information architecture ourselves, and I agree with this. Designers and developers create frameworks for people, and the knowledge needs to be worldwide.

Truth 7: Information architects do information architecture and so should developers, designers, writers, clients, and users.

How do you take something that's a convention and break out of it?

Look at existing behaviors and use those behaviors to create new design techniques. You might see something there you didn't think about.

My wrist is starting to get carpal tunnel; seems like time for another break!

Dan Cedarholm - Interface Design Juggling

One of life's lessons is if you keep repeating the same tasks and expect different results, that's the definition of idiocy.

In keeping with that theme, I'm opting to take a minimalist approach on blogging for Dan Cedarholm's talk due to my wrist hurting. It might continue into tomorrow. Think bullet points.

Dan is funny, he combines juggling with web design and has a toupee website.

Color

Definitely subjective! I'm awful at putting together colors sometimes.

- [Color Generator](#) and [ColorJack](#) for color ideas.
- Grab the palette from a photo. Use the mosaic Photoshop filter to get more specific.
- Join color groups on Flickr for ideas.
- Use textures to make color more interesting.
- Reuse colors from the logo or icons. Utilize a core palette.
- The link color shouldn't be an afterthought.

Typography

Very interesting, although it's something I could stand to improve on. Typography and typefaces are two different things, it's the way you treat it.

That's good news!

- Use all the CSS type classes for different appearance.
- Vary between italic, bold, underlined, whatever!
- Define fonts that people might have with fallback options.
- [Webtypography.net](#) for online type resources.
- Harmonize type and graphics so there's cohesiveness.

Favicons

Something I never really consider, but they show up everywhere. Definitely worth using.

- Sometimes the logo can be used for a Favicon, sometimes not.
- Look for Flashy things you can do with rounded corners.

Recycling

It makes it easy to reuse themes, colors, and graphics, even if some elements aren't consistent. Template your designs.

I'm considering using Wordpress for a CMS for my next site redesign to keep things constant easily.

Microformats

[Standard sets of code](#) that people agree upon that facilitates interface juggling. Change the CSS, javascript, etc, and utilize the code within applications.

It appears like XML for the web or an API and the classes define what kind of data is present. This seems pretty amazing, very flexible, and easy to decode. This is taking us way outside the bubble and seems sort of abstract.

Conclusion

This conference is very much what I described in my first blog post of the day. It's all about breaking the traditions of what we know about web development, and taking us down new roads.

After this, I feel I won't get stuck as much while coding, and realize that I can move things around and make it work differently. Look at all the different options!

Anyway, that's it for today; closing down my computer so I can grab the train on time at 6 pm. See you all tomorrow!

A New Approach to Learning

Yesterday at An Event Apart was quite exciting, but today is going to be even more fun. I spent most of yesterday stressed out about everything under the sun and forgot to have fun!

In order to be successful in life, you have to have fun in what you're doing, both in work and in everything else. Picture a man running and jumping in your head.

That being said, the blog is going to be a lot more playful today. This is in hopes that we will both learn more, and become more successful web developers.

Day 2 - An Event Apart in Chicago, 2007

The room is humming with excitement as the chairs get filled up! There were wall to wall people at An Event Apart yesterday and again today.

Speaking of excitement, I got a chance to speak with Jeremy Keith on the way in. For those of you are curious, he wrote [DOM Scripting](#) and [Bulletproof Ajax](#) (two of the best books in the industry without a doubt!)

Jeremy is a great guy, very humble just like the rest of the speakers here. As any humble speaker, he was very nervous about messing up. You won't mess up Jeremy! And perhaps now I can send him an instant message without him thinking I'm a stalker!

In being in a much better mood today, people are being much more friendly. You have the power to set the atmosphere around you, whatever you want it to be.

Jeremy Keith - Be Pure, Be Vigilant, Behave

Jeremy comes up to talk, he's quite the revolutionary in the web industry. Having him first is like having Eric Meyer first yesterday. [He took a picture of everyone doing the "Eric Meyer Rock On."](#) You can see me in the third row with my mouth open. Quite fun!

His talk is something out of Austin Powers, based on the "Behave" in the title. It starts out being big on the separation of content and presentation, and how to use progressive enhancement.

Be Pure

Start with content. Add some XHTML structure. Mix in some CSS presentation. Sprinkle a bit of behavior with Javascript. Take out layers of the cake as needed. He's going to be comparing CSS and DOM. Nice!

The basics of CSS and Javascript are the same. You're getting elements and ids and classes, determined by your XHTML structure. This excites me and I'm not sure why!

Write in English what you want to do with DOM and then do it. Seems like this conference preaches a lot of writing and sketching before doing, which is a great idea. It actually happens sometimes when I'm not lazy.

Be Vigilant

Do not use "javascript:..." links or onclick="" links! It's like we're superheroes, fighting for web standards. It's funny because we're fighting ourselves sometimes. Lazy much?

He talks about using the correct semantic element. Don't base the element off what behaves the best only when Javascript is enabled. Don't think backwards, think forwards - it will get you further in life and in code.

Javascript is used because we don't like the behavior, we want to switch around the elements. For instance, the way a dropdown <select> box behaves and utilizes the <optgroup> label which I've never seen before. Always learning something new here!

Jeremy uses a lot of loops to narrow down the elements. It's the way to go.

Behave

He defines Ajax as "communicating with the server without refreshing the whole page." The implications are huge, and it's very useful as an additional layer. When you do this, it's called **Hijax**.

Hijax means that everything is working before you add the Ajax, you're just adding efficiency to the process. The server does all the hard work for you! Why not do something that doesn't leave people in the dark when they don't have Javascript?

Ajax takes the browser out of the equation, so you become the browser. You are the one! It breaks things like bookmarking, the back button, etc. If I think I should add these features, then I'm probably doing too much with Ajax. When is it ok?

Ultimately, this is the same thing Eric Meyer spoke about yesterday. DOM Scripting takes away the rules of the browser, you can do whatever you want to the document. Sweet.

Circle Surround

We surrounded Jeremy outside after his talk to ask him a few questions. Interesting points include:

- Javascript libraries can be practical, but only if you understand Javascript. Develop your own libraries to fit your needs
- Him and Eric Meyer are of the same thought process, that anything is possible with CSS and Javascript. They're the entrepreneurs of the Web Standards field
- People were questioning the Hijax method - why not just build it with Javascript to begin with? He says it's more work to do it reverse (like Gmail and Gmail with Plain HTML), so do it the right way

Luke Wroblewski - Best Practices for Form Design

I missed the beginning of this talk when speaking with Jeremy, so I'll try to get as much as I can.

He started talking about the way that forms interact with users. Generally, they are one-way experiences with the form grilling the user. When he asked, no one actually enjoyed filling in forms. As an interesting note, a company I used to work for enjoyed deploying 10-page forms with the reward being a discount. Problem was, the discount wasn't actually advertised anywhere!

How do form layouts perform?

Customers don't think of themselves in terms of a demographic, I definitely don't - I want what I want. In terms of setting up forms, people want it easy and they want clear associations:

- Left-aligned labels have the lowest completion and it doesn't connect with the form field. It might be useful in forms where you want them to make sure all the data is correct so they'll take their time
- Right-aligned labels fall right in the middle - it's harder to read than top-aligned labels, and you can't see the required fields very well. In my opinion, it's also harder to style forms this way, and they're more likely to break.

- Top-aligned labels (label, then form on the next line) are the best way to design a form. People read from top to bottom and don't have to read top to bottom AND left to right, so it's a lot quicker. More customers, more money, more data.

Form indicators

Required fields should be consistent in placement. Also, if all the fields on a form are required, why even have the standard red star? This sort of gave me a shock since I hadn't thought about it this way before. Always disconnect yourself and look for alternate views.

Watch the length of the form field. Make things standard to what people are used to. If it isn't clear, chose a fixed size. [Don't make people think!](#)

Make sure that your **Submit** button and all your other options are different. **Submit** should be the primary button, and everything else should be a different color, regular links, whatever. Don't make them accidentally clear the form.

He showed us an eBay form that has **Cancel** as the *first* option! What are people thinking? Just get rid of the **Cancel** option in the first place!

In testing, buttons of the same color performed better, but the interviews showed that people appreciated the button differential to make them stop, slow down, and considered everything.

Help & Tips

Don't explain things unless they need to be explained. Don't treat people like they're dumb! Overwhelming forms are funny to look at. It might be useful when filling out your taxes - and it is, I've filled out my taxes online. I recommend it!

Maybe even approach help tips as rollovers, so they're not taking attention away from the form. He says this is the best way to go for super-complex forms.

Complete the form

- Don't put in unnecessary forms like **Annual Income**.
- Don't force people to enter data a certain way - accept all the ways!
- Set defaults to form fields when the answer is almost always the same.
- For the 10-page forms, offer a *Save* feature so people can come back later.
- Tailor your forms to be basic with options for advanced input.
- Validate data as people enter it, or use Ajax - don't make them reload the page.
- Give them limits and lead them through the limits (number of characters left)
- Think about blocking out the submit button when a lack of data exists

Learned about [Geni](#). The form is so simple, they created **5 million profiles in 5 months!** After playing with this site for a few minutes, I'm actually addicted to creating my family tree.

[Go to the next An Event Apart post to continue following!](#)

Today over yesterday

The sessions today have been great, and the upcoming session on accessibility is something I have been waiting for. This morning has been full of dynamic and/or legendary speakers, and this afternoon is **Meyer** and **Zeldman** - snooze free!

While all the information here has been valuable, the networking is invaluable, and the event has me thinking of the bigger picture and how to make better websites, day 2 beats day 1 so far!

Derek Featherstone - Accessibility: Lost in Translation

Featherstone starts off by saying that the only thing we really have to follow are the accessibility checklists. He doesn't like this method, and neither do I - I've never been able to sink my teeth into web accessibility.

I'd been hoping he'd present a new way of doing things, and he does - *accessibility as part of the user experience*. Make sure when we're building applications or web sites, we need to make sure:

- That the **user interface is intuitive**.
- On top of that, **functionality must echo what the traditional browser provides**
- Remember **mouse and keyboard interaction** and don't leave people out in the cold
- If people need to edit content, provide them with an **HTML editor in addition to any Javascript**

Live Updates

How do we indicate that the page has been updated? Does the screen reader pick that up?

In an example of *Amazon.com*, it looks like we're running through the code with a screen reader. It sounds a bit confusing, and he says they have a bunch of shortcuts.

Interesting screen reader errors include:

- When the Ajax box pops up, *it turns forms mode off for filling in data*.
- Since there wasn't good alt text, *the screen reader started reading the source attribute instead*.

- If there are no labels for form inputs, *then it starts reading the name of the input.*
- If you insert Javascript pseudo-tags, it tries to read the weird link formatting.

This is sort of revolutionary to me in terms of accessibility and has me taken aback. It's amazing that we don't pay more attention to this and consider all devices when building a website.

Structure of code

Alt text for Javascript widgets should resemble what the widget would actually be labeled as if it were regular HTML. This is probably one of the most important things I've heard at An Event Apart. To dumb it down, we should make everything with good structure, even if it doesn't naturally have good structure.

Google has an API for determining the strength of a password. However, there's no way to get at it with the screen reader.

Make sure that elements are ordered correctly on the page - that's why tables can be confusing. People navigate through forms, especially with screen readers, in a certain way.

Ultimately, how can we structure the code to create the same rhythm, flow, balance, and contrast that design communicates? It's a tough battle.

Does your design tell a different story with the styles off, with the javascript off? If things don't look the same, your users are going to get lost when they're not just there for the design. We fit things in and don't consider how sometimes.

This talk was really cool. Just like CSS and Javascript, you can do whatever you want with your structure to make it accessible. There are functionalities out there that allow you to make anything accessible.

He gives the example of designing an HTML crossword puzzle, which is apparently position. The code and presentation sort of confounds me, though :)

Eric Meyer - CSS in an IE7 World

Eric Meyer is a great presenter, quite funny and enjoys cracking really bad jokes. Life needs more bad jokes, people would enjoy everything more.

When you hear a horrible joke, don't you retell it and laugh at how bad it is? I know I do!

Anyway, off the point! He starts by talking about designing for your audience. *Don't design for IE7 if you're running Netscape 4.0 Plugins dot com!*

IE7 supports a bunch of new features - is it time to drop support for **IE6**? Again, depends on your audience!

New features include:

- **Fixed positioning** - a specific element will never move from a certain area of the browser. For instance, a footer or header.
- **Children of elements** - for instance, `body > div > div`, which is all the divs in a div in the body.
- **First children** - `#ericmeyer > *:first-child` is his first child. `#ericmeyer:first-child` picks all the first children.
- **Attribute selectors** - Css2 can select elements based on properties like language and link source. Css3 (which doesn't seem to be used widely) has a lot of sub-selectors like `https` or `.org`.
- **Form styling** - Sweet! Select form elements with `input[type="text"]` or `input[value="required"]`. It gets very specific and you don't need tons of class elements.
- **Visual Link Typing** - Links that lead to certain document types are styled differently. For instance, `a[href$=".pdf"]` or `a[title*="(PDF)"]`.
- Styling specific table elements for `<tr>` `<td>` `<th>` etc.
- Highlight Images by Type
- Sample Diagnostics by attributes

The list goes on, he covered this topic really well and sort of takes me out of the box. It made me sad to see that 25% of my visitors are on IE6, and Firefox accounts for 75%.

So what about IE6, then? Apparently, [Dean Edwards wrote an IE7 Javascript library to add these features automatically to IE6](#). That's amazing! I can't wait to implement some of these new features for easier use.

[Move on to the final post from An Event Apart, Chicago 2007](#)

Finishing out

The last couple talks of An Event Apart are sure to be powerful. You always need a strong finish; it's what people take home with them. It's what gets them emotional a year down the road.

Part of what tells me that the last speech in particular will be awesome is that there aren't slides; that either means that he's totally winging it, or it's something entirely different. Total audience participation, maybe? Let's get rolling!

Jeffrey Zeldman - Selling Design

This is a good talk for practical use in the business world. How do you get the people in your company to buy into your designs?

This is generally a problem with anything - how do the people around you accept your ideas, and how can you persuade them? Let's take a look!

It's all about the relationships

Great clients aren't found, they're made. Except if you're a hermit, in which case you're probably stuck inside all day designing in your underwear hoping for new clients!

Great clients are created through endless scopes, in theory - "but some of us would like to..sleep...it's nice to have a social life and family and all that other stuff." If you respect each other, then the results will be positive.

Just because someone doesn't know a lot about design or technology, they are not an idiot. We're just elite?

He talks about working for a bad manager and them having a bad impression of you. Sometimes, no matter what you do, things will not change. It means they knew in January they needed it in September, but they didn't get approval until August.

Even if you make a deal to have a several day turnaround, they will most likely ignore your emails and they'll still need it when they need it. No good!

For new clients, have people go through a pre-qualification process first to get in touch with you by phone. Otherwise, they're not worth your time.

It might be based on how flexible they are to what they need. It's definitely based on your personality and how you interact with theirs, and follow your instincts.

Clients that are panic stricken are possibly good clients up front, but that's just super stressful and honestly I wouldn't do it based on my own priorities.

Selling to Clients

The non-webby client

Happycog (Zeldman's web development company) gives the client a presentation about the web and the main features of it, so you can reference blogs, wikis, whatever. This might work well in organizations when you're presenting to the higher-ups.

Share your process with the client and share with them where in the process you are. That's smart, it shows the client that they're dealing with busy professionals.

In the business world, you need to translate what people are saying. If they ask a question, what's the sub-text? It applies to design, too.

Gear a new website to people's image, and the design and content and everything represents their image.

The last point I'd like to make from his talk is about taking criticism. For instance, someone might say the "button is too big." They might be right if you just opted to make it big, and you should change it. You might have made it big because users couldn't find it, in which case you kindly remind them.

Conclusion of An Event Apart

There is one more speaker but I've opted to not take notes and just have some fun listening. It's about pragmatic client projects vs personal crazy projects, some of which I've already personally delved into. When you're doing things in one, you use those things in the other.

But anyway, I've really enjoyed blogging An Event Apart, Chicago 2007 for you, and I hope you enjoyed following along.

Please leave any comments and feel free to contact me with any questions. Also feel free to email me if you want to invite me to your conference for free. I'll blog that, too :)