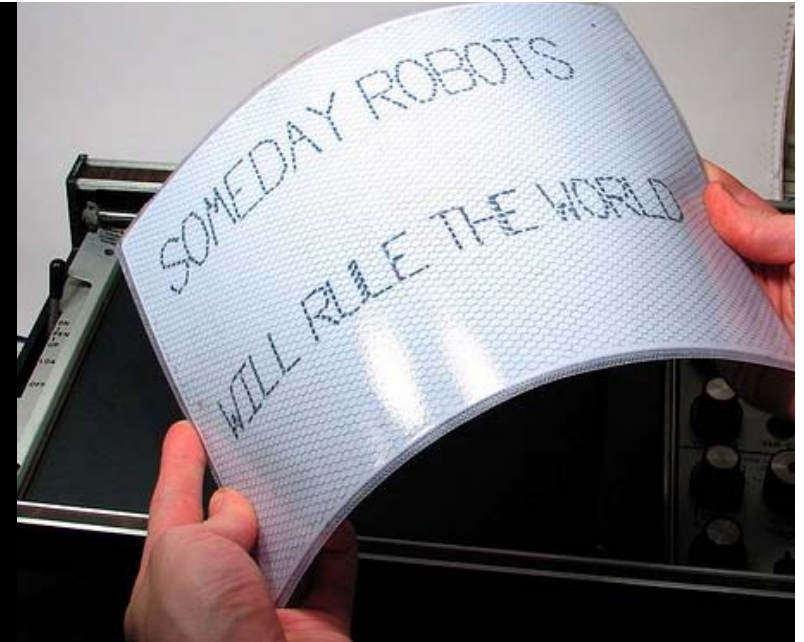


**Progressive
Enhancement
With JavaScript
and Ajax**

Highland Fling 2007

Christian Heilmann

- **As web developers we are in a very lucky position.**
- **Unlike real world products, web sites do not have a fixed state.**
- **Web documents were right from the start meant to be adaptive to their environment.**
- **By keeping things flexible and adaptive we can please all consumers.**

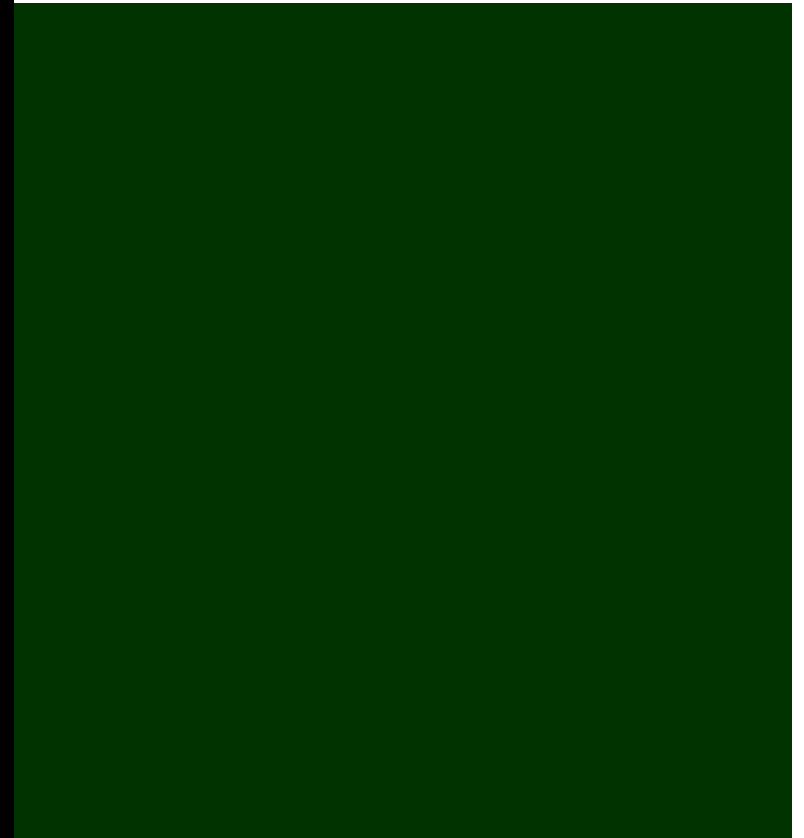


- **Real life enhancement**

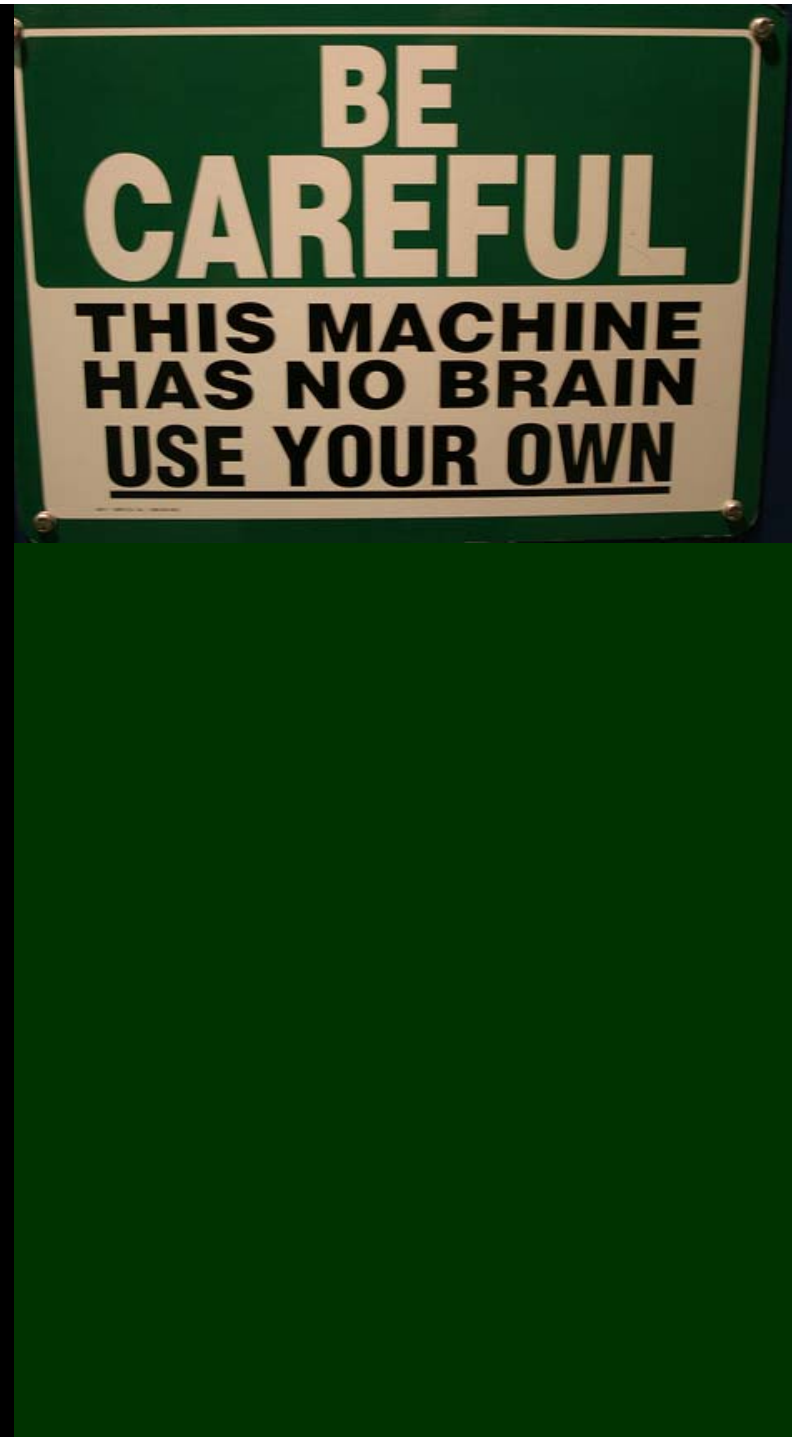


- Real life has some ideas of progressive enhancement, most of which are old chestnuts:

- Measure twice, cut once.
- Check the depth of the lake before you plunge in.
- Try before you buy
- ~~Use BitTorrent~~



- **The same logic applies to JavaScript and Ajax:**
 - **Test if JavaScript is available.**
 - **Test if the DOM is supported**
 - **Test for the HTML you want to use.**
 - **Test for any object you want to use.**
 - **Don't rely on the connection to be available.**



- **Testing for JavaScript is easy, simply don't make your product rely on JavaScript:**
 - Death to **href="javascript:foo()"**
 - Death to **onclick="foo()"**
 - Who the hell ever told you to use **onclick="javascript:foo()"** ?
- **Don't mix HTML and JavaScript, use the DOM to access HTML.**
- **The DOM is your API to the document.**



- **Testing for DOM support is as easy:**
 - Check that **getElementById** and is available.
 - If you still care about Opera 6 also test for **createTextNode**

```
function init(){
    if( document.getElementById &&
        document.createTextNode )
        // we have DOM, wahey!
        checkDependencies();
    }
}
```

- **Testing for the HTML you need is also easy:**
 - **Check that all the elements are available.**

```
function checkDependencies(){
    var usedIds = ['nav','content'];
    var isOk = true;
    for(var i = 0; usedIds[i]!==undefined; i++) {
        if( !document.getElementById( usedIds[i] ) ){
            isOk = false;
        }
    }
    return isOk;
}
```

- **Testing for the HTML you need is also easy:**
 - **Check that all the elements are available.**
 - **Check that they also have the right child nodes**

```
function myNavStuff(){
    var n = document.getElementById('nav');
    if(n.getElementsByTagName('ul').length > 0){
        // I got your code right here!
    }
}
```

- **Testing for object availability**
 - **Simply test for any object before you access it**

```
var request;  
try{  
    request = new XMLHttpRequest();  
}catch(error){  
    try{  
        request = new ActiveXObject("Microsoft.XMLHTTP");  
    }catch(error){  
        return true;  
    }  
}
```

- **Prepare for unreliable and flaky connections.**
 - **Set a timer to time out and abort the connection after a certain period has passed.**
 - **Define a success and failure case for any connection.**
 - **Make sure you avoid caching issues by setting an expired HTTP header on your Ajax calls**



- This is the “how” of progressive enhancement with JavaScript and Ajax.
- The question however is **why use JavaScript at all?**
- After all it can be turned off.



- Instead we can use the amazing new trick called “CSS only solutions”.
- Why bother with writing complex scripts when you can do the same with some easy lines of CSS?
- Showing and hiding menus?

```
#nav li ul { display:none }
```

```
#nav li:hover ul {
```

```
    display:block
```

```
}
```

the Suckerfish

Overview	Species	Links
<p>Lorem ipsum dolor sit Suspendisse egestas risus. Praesent nonum pulvinar et, volutpat vel tempus. Nulla ac dui a ullamcorper sodales ri Phasellus ut mauris ne dapibus sapien. In ma tellus a vestibulum pha purus ipsum eu velit. P</p> <p>Lorem ipsum dolor sit consectetuer adipiscing Suspendisse egestas pede. Phasellus suscip blandit risus. Praesent nonummy. In erat. Duis nibh pede, accumsan eu, pulvinar et, volutpat vel, elit. Curabitur</p>	<p>Ceylonese remora Remora remora Sharksucker Slender remora Spearfish remora Whitefin sharksucker</p> 	<p>ipiscing elit. us suscipit blandit pede, accumsan eu, i sed nunc congue my. Maecenas dolor. Proin eu turpis. Nunc porttitor etus. Nulla egestas, tor lacus, ut semper cumsan turpis at erat.</p>  <p>The Remora</p>

- Using CSS for behaviour is based on one trick, which is pseudo selectors like **:hover**, **:active** and **:focus**
- These have several problems:
 - Some browsers only support them on interactive elements like links and buttons.
 - They have a strict on or off state.
 - and they fail for a certain user group...



- **Keyboard access support in HTML and browsers is bad.**
- **Support for `:active` and `:focus` is even worse.**
- **Which means that CSS only solutions with keyboard support turn out to be HTML soup nightmares (nesting tables inside links...)**

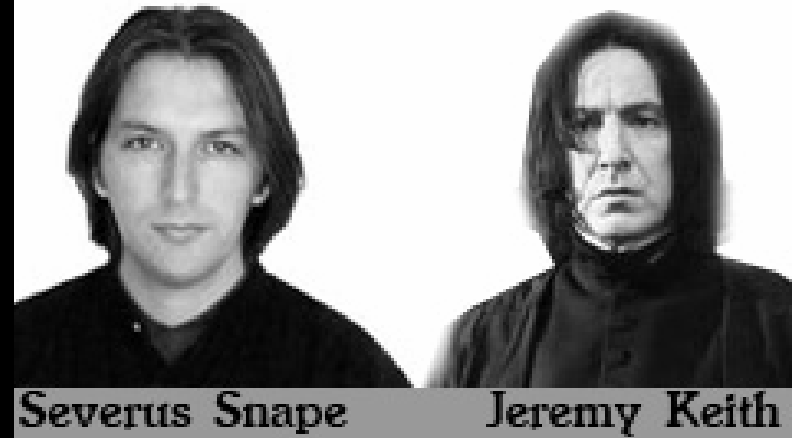


A revised and simplified flyout menu with THREE sub levels

28th July 2006

Item 1			
Item 2			
Item 3 »	Item 3a		
Item 4	Item 3b »	Item 3bi	
Item 5	Item 3c	Item 3bii	
Item 6		Item 3biii	Iter
			Iter

- In essence, CSS only solutions are a bit of a shifty character and you don't know if you can trust them.
- It is also debateable if CSS only solutions are in the end easier than JS solutions.



Search Results: dropdown

Your search for "dropdown" returned 861 matches. Displaying results 1 to 40.

Subject

[\[css-d\] does anyone know ...](#)

[Re: \[css-d\] IE7 dropdown submenu shows behind menu](#)

[Re: \[css-d\] CSS IE6 Dropdown Issues](#)

[Re: \[css-d\] CSS IE6 Dropdown Issues](#)

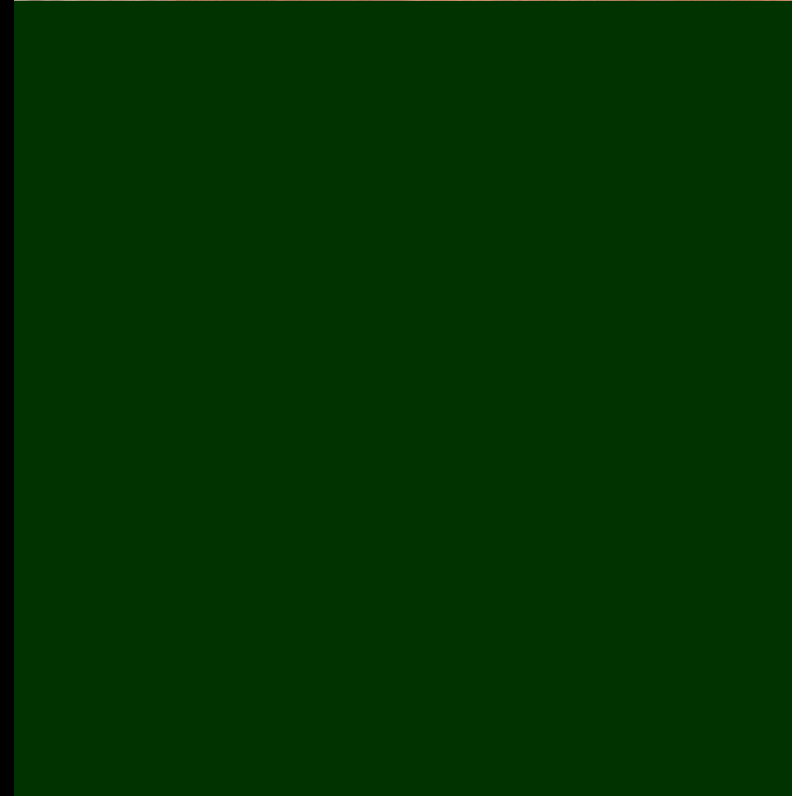
[\[css-d\] CSS IE6 Dropdown Issues](#)

[Re: \[css-d\] CSS Development General Question](#)

[Re: \[css-d\] CSS Development General Question](#)

- **JavaScript has so many more options for you:**

- **Real testing for support of objects**
- **Real event handling including simulation of application keyboard handling (cursors instead of tabbing)**
- **timeout() to allow for user errors**
- **Creation of necessary markup on demand.**



- **JavaScript has so many more options for you:**
 - **Test of available screen estate**
 - **Loading of extra content on demand**



Example: Multi-tiered Menu From Markup (YUI Library) - Mozilla ...

File Edit View Go Bookmarks Tools Accessibility Help

http://

Getting Started Latest Headlines post to del.icio.us Show Meta Info

Navigation Text Equivalents Scripting Style Validators Tools

Contacts Events Reviews Tags

Disable Cookies CSS Forms Images Information Miscellaneous

Google Mail ... WordPress ... Summary (...) Wait till I co... Example: ...

Example: Multi-tiered Menu From Markup (YUI Library) [\[Examples Home\]](#)

This example demonstrates how to create a multi-tiered menu using existing markup on the page. The Menu API also supports building a menu like this [using nothing but JavaScript](#).

Note: By default clicking outside of a menu will hide it. Additionally, menu items without a submenu or a URL to navigate will hide their parent menu when clicked. Click the "Show Menu" button below to make the menu visible if it is hidden.

Show Menu

- Communication >
 - 360°
 - Alerts
 - Avatars
 - Groups
 - Internet Access
- Shopping >
 - Yahoo! Mail
 - Yahoo! Address Book
 - Yahoo! Calendar
 - PIM
 - Member Directory
 - Messenger
 - Mobile
 - Photos
 - Yahoo! Notepad
- Entertainment >
- Information >

http://calendar.yahoo.com/ FoxyProxy: Disabled

- Does this mean that only scripters should create interfaces and CSS is not needed?
- Not at all, as first of all CSS can help JavaScript a lot.



- **Task: Hide all list items in the first UL**
 - **A lot of code, that one...**

```
window.onload = function(){
  if(document.getElementById && document.createTextNode){
    var uls = document.getElementsByTagName('ul');
    if(uls.length > 0){
      var lis = uls[0].getElementsByTagName('li');
      for(var i = 0, j = lis.length; i < j; i++){
        lis[i].style.display = 'none';
      }
    }
  }
}
```

- **Task: Hide all list items in the first UL**
 - **How about leaving the main job to the CSS parser?**

```
window.onload = function(){
  if(document.getElementById && document.createTextNode){
    var b = document.body;
    b.className += b.className ? ' dynamic' : 'dynamic';
  }
}
<style type="text/css">
  body.dynamic ul li{
    display:none;
  }
</style>
```

- **This also allows you to keep the whole look and feel of your scripts in the hands of the CSS maintainers.**
- **All you need to do is make sure you communicate the class and ID names with them.**



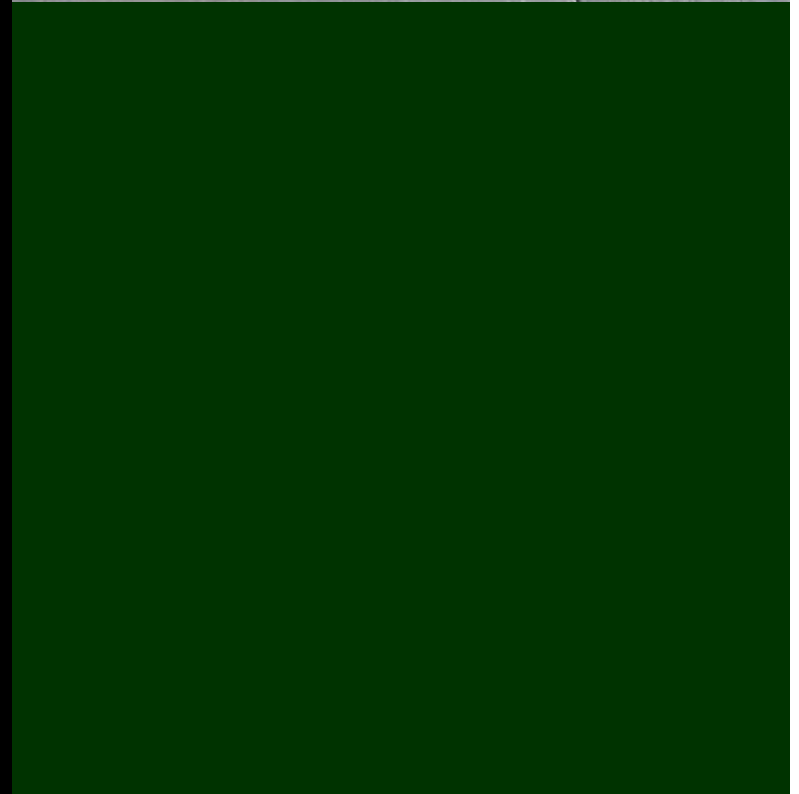
- You achieve highest flexibility by separating those and the text labels out into own includes:

`slideshow.js`

`slideshow-css.js`

`slideshow-labels.js`

```
slideshow.css = {  
  
  'navigationID': 'nav',  
  'contentID': 'main',  
  'hide class': 'hide'  
}
```



- **There is a danger in progressive enhancement with JavaScript though:**
- **Be aware that what you enhance is as important as how you enhance it.**
- **A solution that doesn't really make sense shouldn't be enhanced, but changed.**



- Useless progressive enhancement.

Do you live with a parter? yes no

Do you live with a parter?

Do you live with a parter?

I live with a partner

I live with a partner



- **It is important to know your audience**
- **It is important to keep up with technology as a lot of truths of the past are not applicable any longer.**



- Research what can be enhanced and how.
- Example “connected select boxes”:

Form illustrating connected select boxes:

Top dropdown: Foo3

Second dropdown: Foo3_1

Submit Query button

Form illustrating a connected select box:

Top dropdown: Foo1

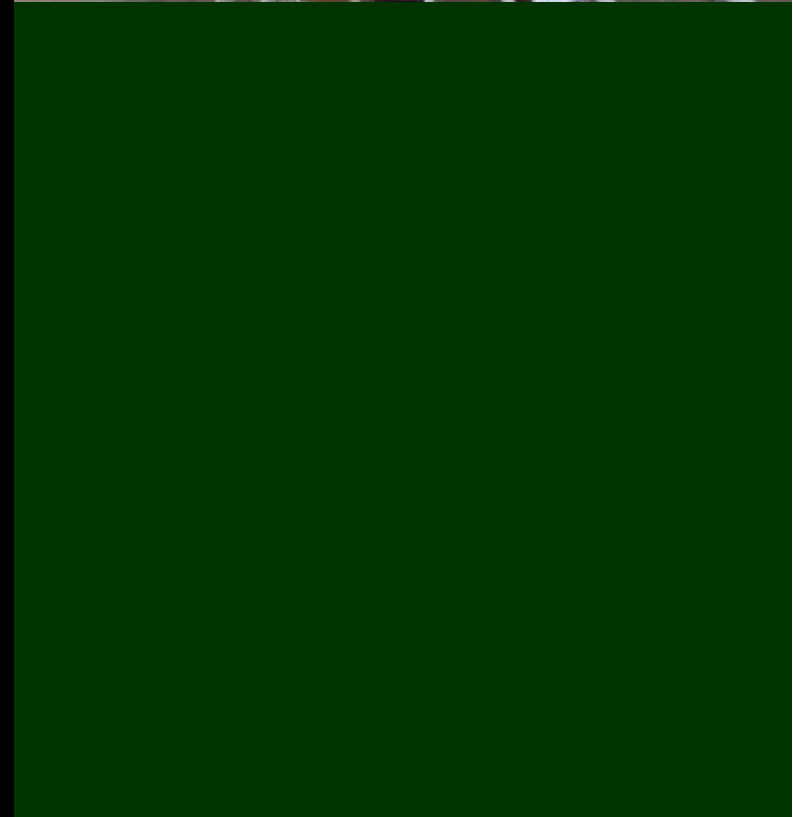
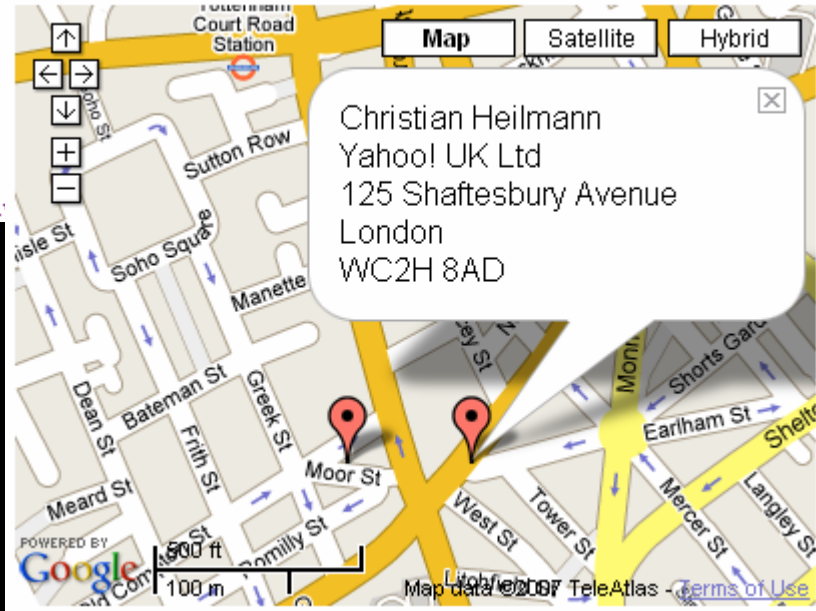
Dropdown list contents:

- Foo1**
- Foo1_1
- Foo1_2
- Foo1_3
- Foo1_4
- Foo1_5
- Foo1_6
- Foo2**
- Foo2
- Foo2_1
- Foo2_2
- Foo2_3
- Foo2_4
- Foo2_5
- Foo2_6
- Foo3**
- Foo3
- Foo3_1
- Foo3_2



- Research what can be enhanced and how.
- Example “heard to Googlemap”

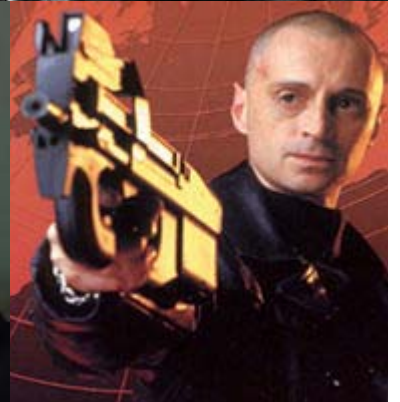
```
<div class="vcard">  
  <span class="given-name">Christian</span>  
  <span class="family-name fn">Heilmann</span>  
  <div class="organization-name">Yahoo! UK Ltd</div>  
  <div class="address">125 Shaftesbury Avenue</div>  
  <div class="city">London</div>  
  <div class="postal-code">WC2H 8AD</div>  
</div>
```



- **So why isn't progressive enhancement yet a common practice?**



- **Stockholm Syndrome**



- **Stockholm Syndrome**
- **Enhanced interfaces don't go down too well with the mainstream audience, we are used to waiting and things breaking.**
- **Technologies are perceived as given, no need to improve them.**



- **Monetary reasons**
- **Progressive Enhancement costs money**
- **As with any best practice there is no immediate tangible benefit**



- **Show desire**
- **You can help to change this**
- **Look at things, and see how you can improve them**
- **If need be, do it on the sly
(heard in Yahoo! Local)**



- **Benefit from experience of others**
- **A lot of tricks that helped us in the past can be brushed up and renewed now.**
- **Example “lowsrc”**



- **Keep your feelers out there.**
- **Read up on technologies and what is done with them.**
- **Hot:**
 - <http://www.w3.org/TR/aria-roadmap/>
 - <http://www.whatwg.org/>



- **Contact, subscribe, talk, share.**
- **If you know something, please tell others:**

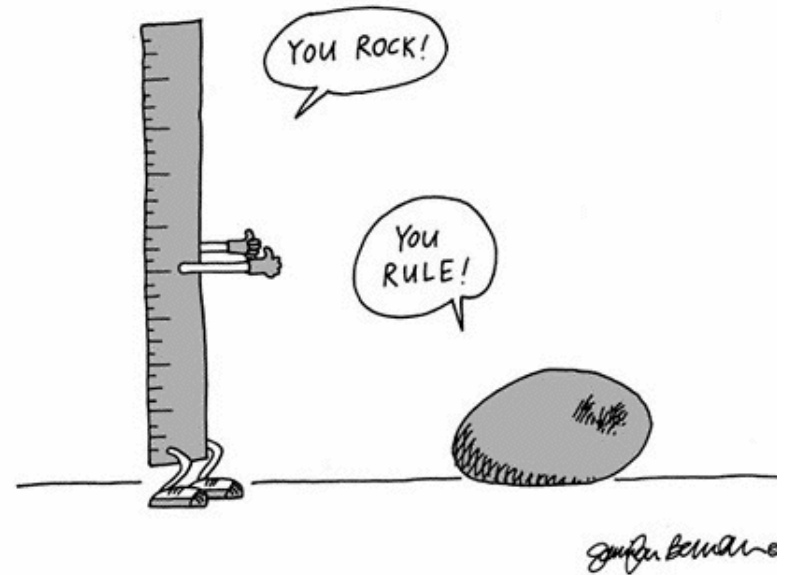
<http://makemeaspeaker.com>



- **Good progressive enhancement helps the user, and not the developer.**
 - **Automatic saving of drafts in email apps**
 - **Parallel upload of content while you can add meta data**
 - **Making it easy and fun to add content.**
 - **Preventing unnecessary steps.**



THANKS!



chris.heilmann@gmail.com

<http://wait-till-i.com>

<http://icant.co.uk>

Flickr/delicious: codepo8