

Thomas Fuchs

Advanced Rails AJAX techniques



wo||ze||e
mo||ze||e

Thomas Fuchs

- **wolllzeile**
- Core team member of
Ruby on Rails
- *script.aculo.us*
- **mir.aculo.us**



wollzelle

mollzelle

- Branding/Design/Web
- fluxiom (yes, we'll get to see it)
- <http://www.wollzelle.com/>

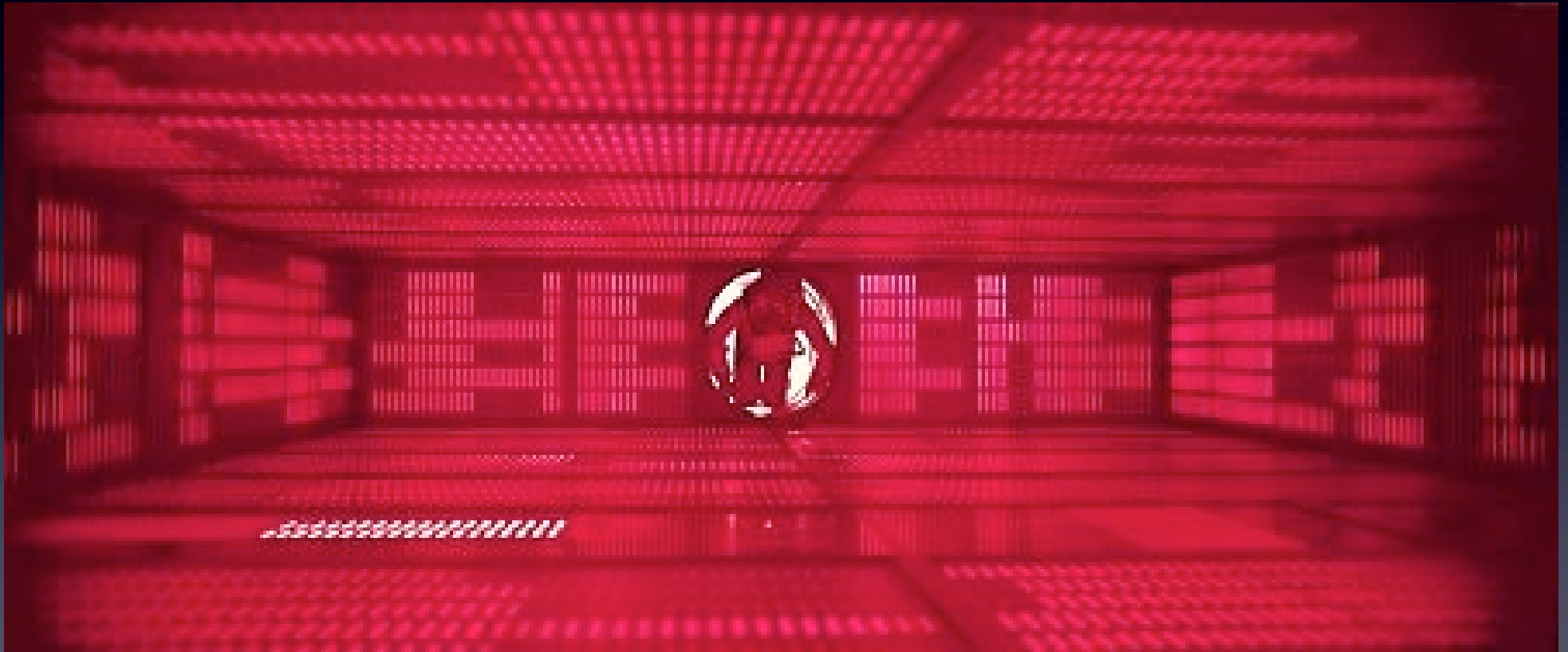
The tools

JavaScript, as seen by the browser



Source: <http://www.pollux.franken.de/KNF/>

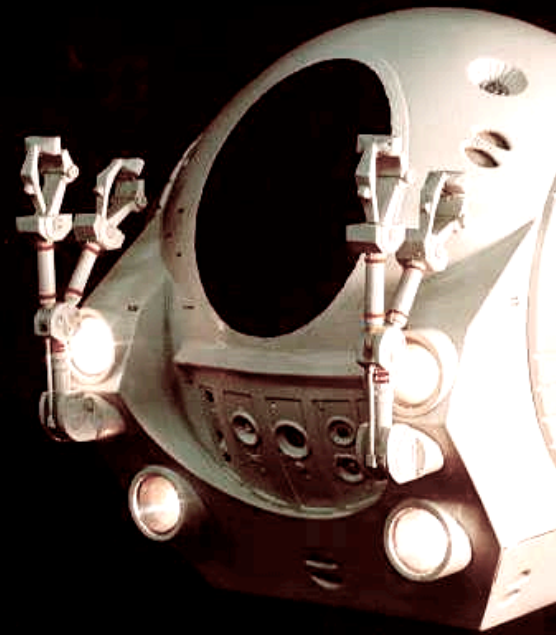
JavaScript, as seen by the developer

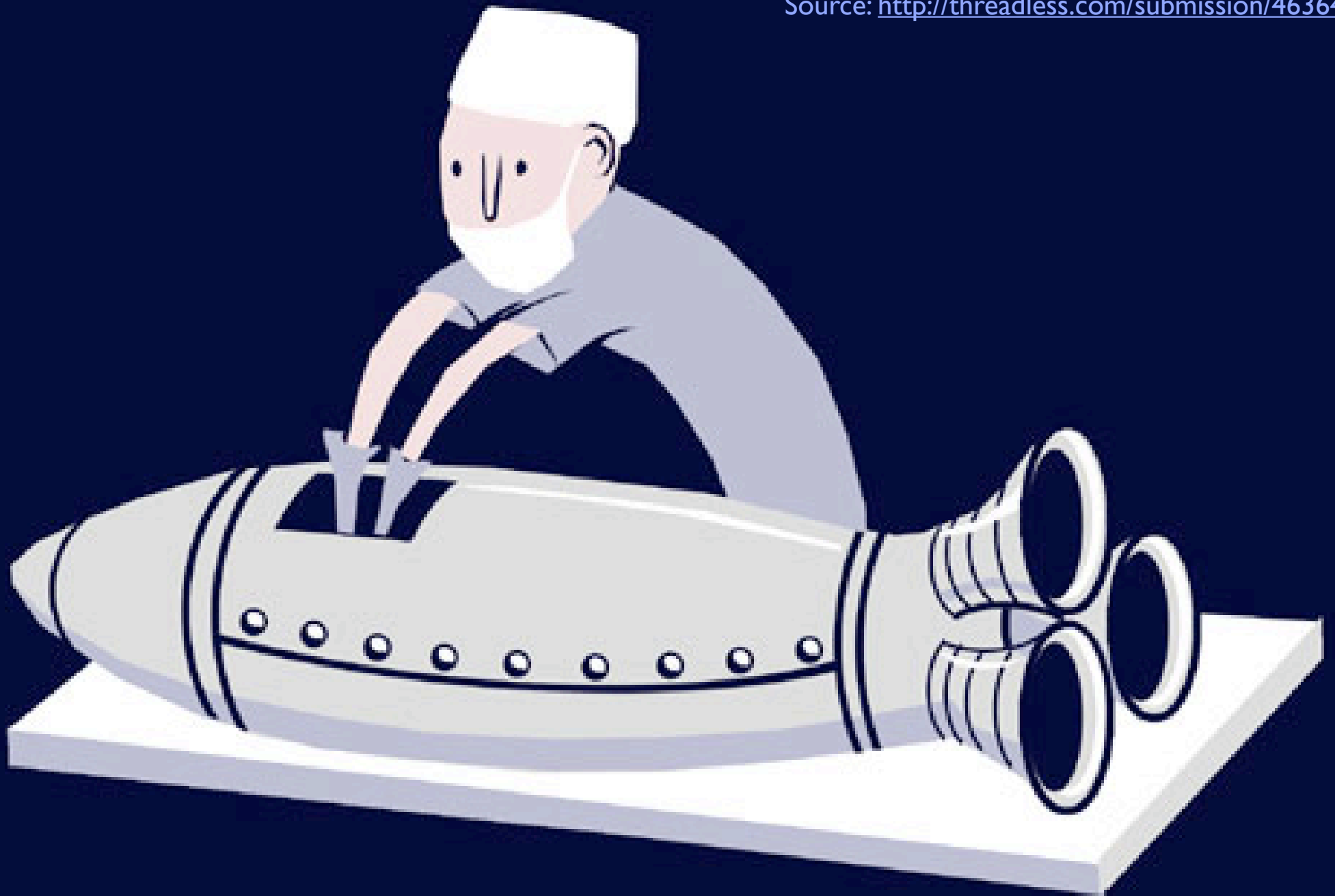


From 2001:A Space Odyssey

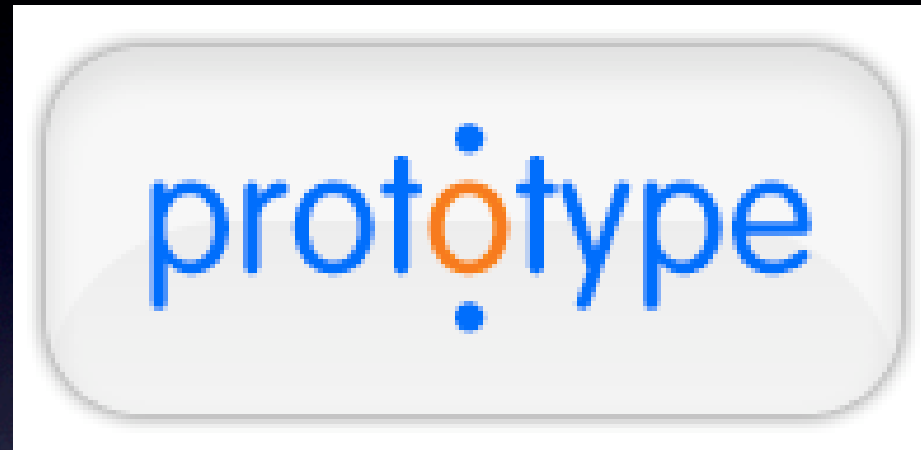
Concentrate on
your application...

...and not on
browser bugs.





IT'S NOT ROCKET SURGERY...



1.5

JavaScript,
minus the „WTF?“

AJAX

Enumerables

\$ and \$\$


```
$$('p.hint').each(function(e){  
    e.hide();  
});
```

Direct Element extensions

Prototype 1.4

```
if(Element.visible('blah'))  
    Element.hide('blah');
```

VS

Prototype 1.5

```
if($('blah').visible())  
    $('blah').hide();
```

Events

script.aculo.us

1.6

Visual effects

Drag and Drop

Ready-to-use controls

Plus some extras

JavaScript unit testing, DOM Builder

On-the-fly JavaScript: Rails JavaScript Templates

A sepia-toned historical photograph showing a large crowd of men, mostly in 19th-century attire including top hats and heavy coats, gathered around a railway track. The men are looking towards the right side of the frame. In the foreground, a single railway track runs diagonally from the bottom left towards the center. The background is slightly hazy, suggesting an outdoor setting with trees in the distance.

Rails Ajax: A history

Last Spike of the CPR - Craigellachie, British Columbia, Canada
Courtesy of the National Library and Archives of Canada

Rails Ajax: A History

Ideas to make the
web snappier

Late 20th century

Prototype 1.0,
Rails Helpers

Rails 0.11, March 22, 2005

Prototype 1.3,
script.aculo.us

July 6, 2005

Prototype 1.4,
script.aculo.us 1.5

Rails 1.0, December 13, 2005

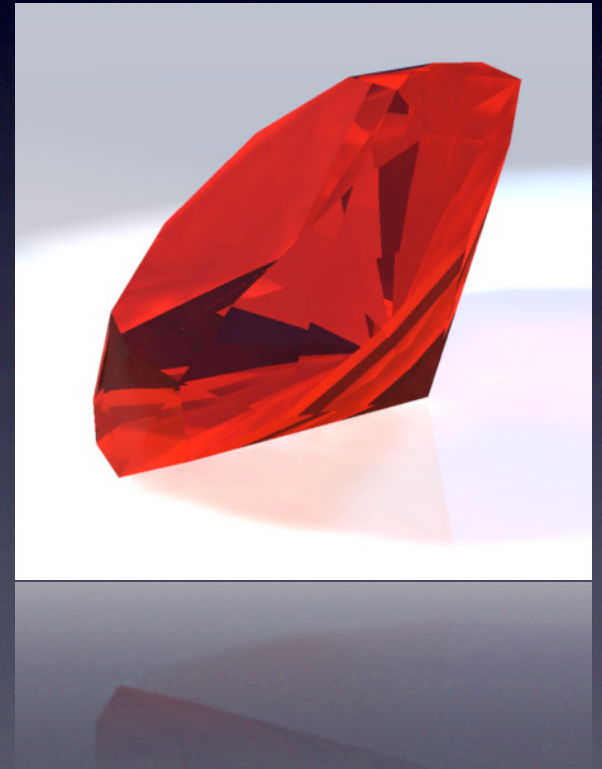
Rails JavaScript templates,
Prototype 1.5, script.aculo.us 1.6

Rails 1.1, March 28, 2006

In just one year



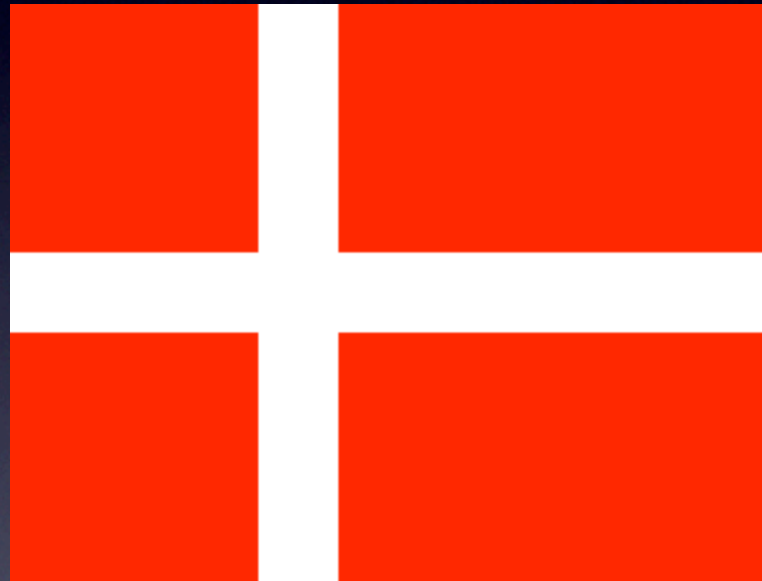
Ruby on Rails RJS Templates





No JavaScript
programming necessary

(might have something
to do with David)



Danish flag

Say it in Ruby

```
page[:cart].update render(:partial => 'cart')

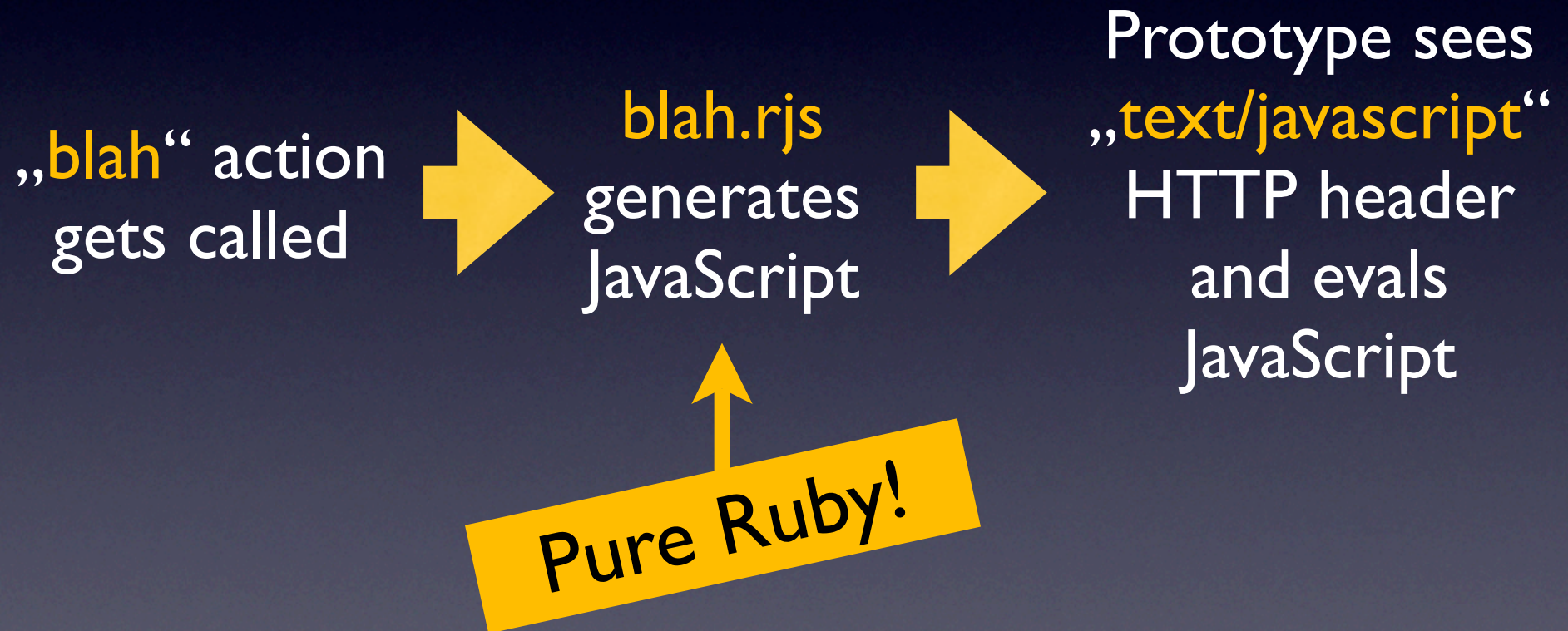
page.select('p.hint').each do |hint|
  hint.visualEffect :highlight
end
```

generates JavaScript:

```
$("#cart").update("...rendered partial...");

$$("p.hint").each(
  function(value, index){
    value.visualEffect("highlight");
  }
);
```


What it does



Of course,
just think of it as magic.

*Sam Stephenson,
author of Prototype*



replace

Replaces complete element

```
page.replace :list, render(:partial => 'list')
```



```
Element.replace("list", "<ul id=\"list\">\n...
```


replace_html

Replaces contents of an element

```
page.replace_html :list, render(:partial => 'list')
```

same as

```
page[:list].update render(:partial => 'list')
```


Staying DRY

```
page[:list].reload
```

Refreshes contents of element „list“ with
the partial of the same name

equivalent to

```
page.replace_html :list, render(:partial => 'list')
```

insert_html

Insert content at specific positions within an element

```
page.insert_html(  
  :bottom, :list, "<li id=\"i_2\">Item from RJS</li>\">  
page.sortable :list
```

page.sortable re-inits the sortable list so the newly added item is recognized

select

CSS selector-based loops

```
page.select('p.hint').each do |hint|  
  hint.visual_effect :highlight  
end
```



```
$$("p.hint").each(function(value, index) {  
  value.visualEffect("highlight");  
});
```


select

advanced Prototype enumerables

```
page.select('p.hint').pluck('t', 'offsetTop')
```

creates a JavaScript „t“ variable, containing the result

select

advanced Prototype enumerables

```
page.select('p.hint').find_all('h') do  
  page << "value.visible()"  
end
```

creates „h“ variable, containing an Array of elements
(that are visible)

select

using Prototype to inspect stuff

```
page.select('p.hint').find_all('h') do  
  page << "value.visible()"  
end  
  
page << "alert(h.inspect())"
```



http://localhost:3000

[[object HTMLParagraphElement], [object
HTMLParagraphElement]]

OK

<<

Reuse plain JavaScript

```
page << "(3).times( function(){ alert('Ho!') }));"
```


delay

```
page.delay(5.seconds) do
  page[:blah].visualEffect :puff
end
```

```
page.delay(15.seconds) do
  page.select('p.hint').each do |element|
    element.visualEffect :fade
  end
end
```

Prototype/script.aculo.us

Element extensions

```
page[:blah].set_style :backgroundColor => '#f00'  
page[:blah].set_opacity 0.5
```



```
$("#blah").setStyle({"backgroundColor": "#f00"});  
$("#blah").setOpacity(0.5);
```


Chains

Hide the first element that is a „P“ and has a class „hint“:

```
page.select('p.hint').first.hide
```

script.aculo.us

Visual Effects

```
page[:blah].visual_effect(  
  :pulsate, :duration => 4.seconds)
```

:appear, :fade, :slide_up, :slide_down,
:blind_up, :blind_down, :pulsate, :switch_off,
:puff and more

(or DIY)

script.aculo.us

Drag & Drop

```
page.draggable :blah
```

```
page.drop_receiving :blah,  
  :url => { :action => 'do_something' }
```

```
page.sortable :list,  
  :url => { :action => 'do_something' }
```

Call your own

RJS

```
page.abra_cadabra.simsalabim
```



JavaScript

```
AbraCadabra.simsalabim();
```


render :update (controller-based)

```
def update
  render(:update) do |page|
    page.hide :cart
  end
end
```

RJS helpers

Helper:

```
module SomeHelper
  def ajaxy_stuff
    page.hide :blah
  end
end
```

Call in .rjs:

```
page.ajaxy_stuff
```


redirect_to

```
def redirect
  render(:update) do |page|
    page.redirect_to 'http://www.rubyonrails.org/'
  end
end
```

(takes url_for syntax)

respond_to

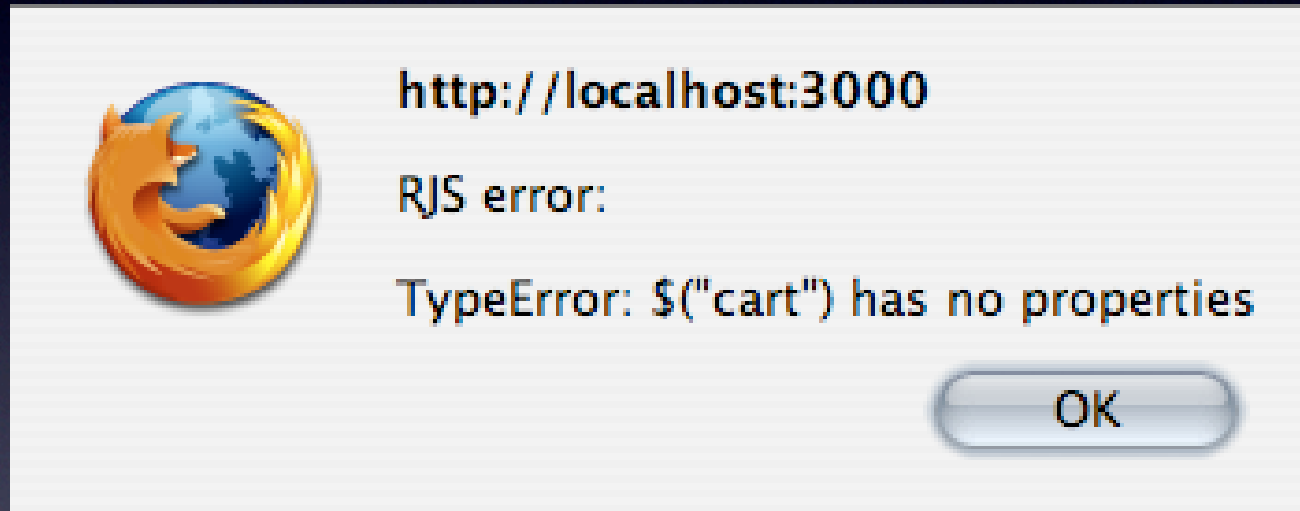
```
def redirect
  respond_to do |type|
    type.html do
      redirect_to "http://www.rubyonrails.org/"
    end
    type.js do
      render(:update) do |page|
        page.redirect_to "http://www.rubyonrails.org/"
      end
    end
  end
end
```



one action for Ajax...

...and non-Ajax calls

Helpful hints, built-in



config/environments/development.rb:

```
config.action_view.debug_rjs = true
```

API Documentation

ActionView::Helpers::PrototypeHelper

and

ActionView::Helpers::PrototypeHelper::
JavaScriptGenerator::GeneratorMethods

**What about
debugging?**

No rocket surgery,
either.

The first
rule of
AJAX
debugging:
Use Firefox

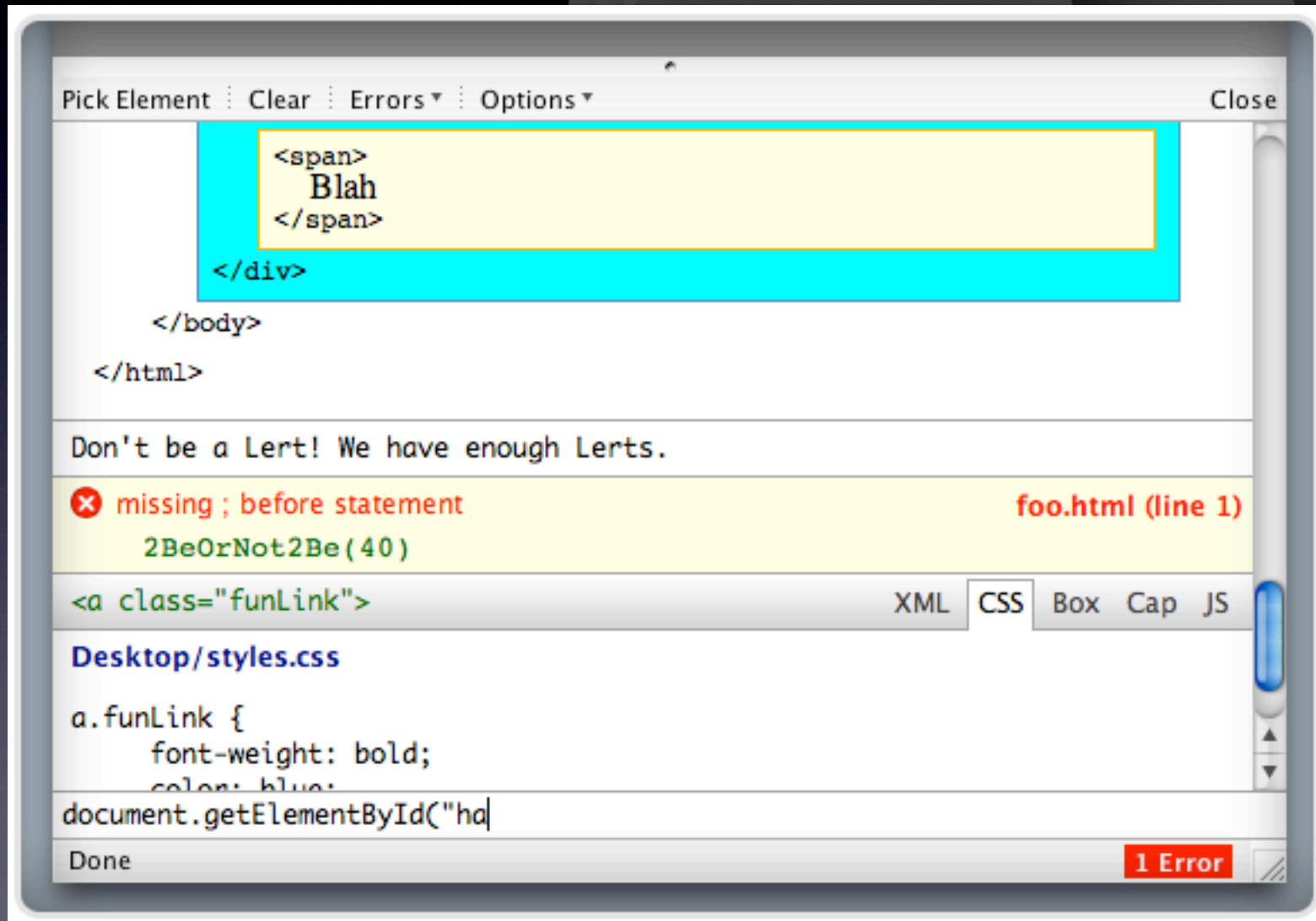


The second
rule of
AJAX
debugging:
Use Firefox

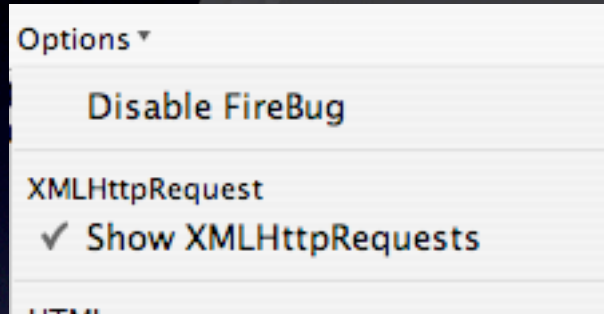
+Extensions



Firebug

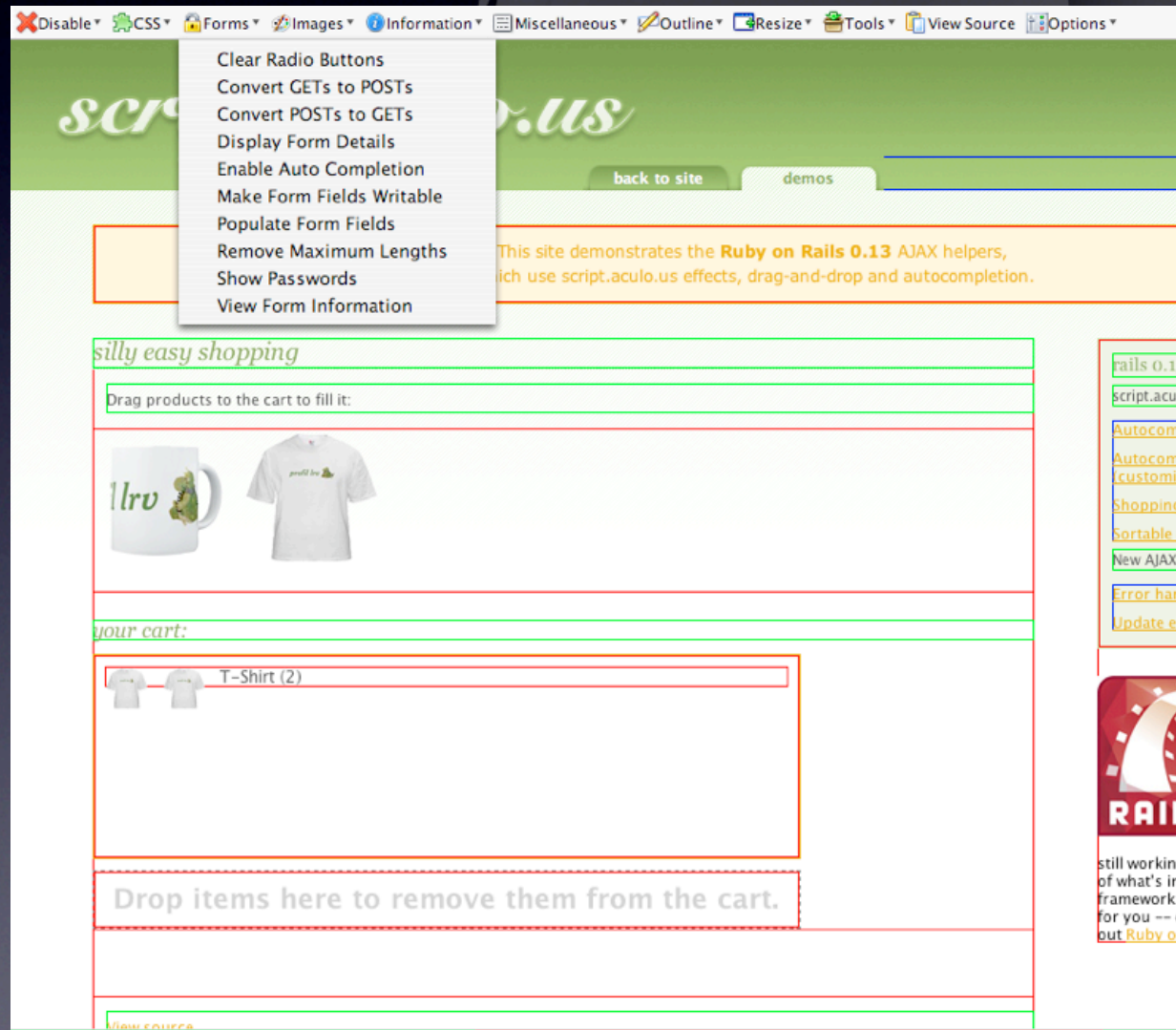


Firebug



post http://localhost:3000/logtest/update	Post	Response
<pre>try { \$("list").replace("<ul id=\"list\">\n <li id=\"i_1\">item 1\n \n "); \$\$("p.hint").each(function(value, index) { value.visualEffect("highlight"); }); var h = \$\$("p.hint").findAll(function(value, index) { return value.visible(); }); \$\$("p.hint").first().hide(); new Draggable("blah", {}); }</pre>		

Web Developer Extension



Venkman JavaScript Debugger

The screenshot shows the Venkman JavaScript Debugger interface overlaid on a web browser window. The browser window displays a page titled "script." with a section "silly easy shopping" and a "your cart:" section showing two T-Shirts. The debugger interface includes a toolbar with buttons for Stop, Continue, Step Over, Step Into, Step Out, Profile, and Pretty Print. The "Loaded Scripts" panel lists various functions and their line numbers. The "Local Variables" panel shows the current scope, event, style, and this object. The "Source Code" panel displays the JavaScript code for "dragdrop.js", with a breakpoint set at line 295. The "Breakpoints" panel shows the breakpoint at line 294 of dragdrop.js. The "Interactive Session" panel shows the current session context and a list of commands.

Loaded Scripts

Name	Line
[endeffect]	247
[destroy]	282
[currentLeft]	288
[currentTop]	291
[startDrag]	294
[finishDrag]	307
[keyPress]	333
[endDrag]	341

Local Variables

Name	Value
scope	{Call}
event	{Event}
style	void
this	{Object}

Source Code

```
288 currentLeft: function() {
289     return parseInt(this.element.style.left || '0')
290 },
291 currentTop: function() {
292     return parseInt(this.element.style.top || '0')
293 },
294 startDrag: function(event) {
295     if(Event.isLeftClick(event)) {
296         this.active = true;
297     }
298     var style = this.element.style;
299     this.originalY = this.element.offsetTop - th
300     this.originalX = this.element.offsetLeft - th
301     this.offsetY = event.clientY - this.original
```

Breakpoints

Name	Line/PC
dragdrop.js	294

Interactive Session [context: dragdrop.js, scope: [startDrag]]



```
http://www.hacksrus.com/~glinda/venkman/faq/venkman-faq.html>. Visit the Venkman
homepage <http://www.mozilla.org/projects/venkman/> for more information.
- You are running Venkman version 0.9.84.
- Use "/help <command-name>" for help on specific commands.
- Visit <-x-jsd:help> for a searchable command reference.
- Commands start with a forward-slash ('/') character. Any text that DOES NOT start with a
forward-slash will be evaluated as JavaScript. For example, to execute the "step" command,
type "/step". To evaluate "1 + 1", just type "1 + 1".
- Recorded local startup 26, global 2458322.
* Stopped for breakpoint.
- #0: function anonymous(event=Event:0) in <http://script.aculo.
us/demos/scripts/dragdrop.js> line 295
: 293: },
: 294: startDrag: function(event) {
: 295:     if(Event.isLeftClick(event)) {
: 296:         this.active = true;
: 297:
```


Tamperdata


This site demonstrates the **Ruby on Rails 0.13** AJAX helpers, which use script.aculo.us effects, drag-and-drop and autocompletion.

silly easy shopping

Drag products to the cart to fill it:



your cart:

 T-Shirt (1)

Drop items here to

[View source](#)

Tamper Data - Ongoing requests

Start Tamper Stop Tamper Clear

Time	Duration	Size	Method	Status	Content Type	URL
14:42:...	196 ms	-1	POST	200	text/html	http://script.aculo.us/demos/shop... ..

Request Header Name	Request Header Value
Host	script.aculo.us
User-Agent	Mozilla/5.0 (Macintosh; U;...
Accept	application/x-shockwave...
Accept-Language	en-us,en;q=0.5
Accept-Encoding	gzip,deflate
Accept-Charset	ISO-8859-1,utf-8;q=0.7,*...
Keep-Alive	300
Connection	close
X-Requested-With	XMLHttpRequest
X-Prototype-Version	1.3.0
Content-Type	application/x-www-form-...
Content-Length	15
Cookie	_session_id=6878c1b45...
POSTDATA	ⓧid=product_2&_= ⓧ

Response Header Name	Response Header Value
Status	OK - 200
Date	Mon, 07 Nov 2005 14:38:...
Server	Apache/2.0.52 (Gentoo/L...
Set-Cookie	_session_id=6878c1b45...
Cache-Control	no-cache
Connection	close
Transfer-Encoding	chunked
Content-Type	text/html

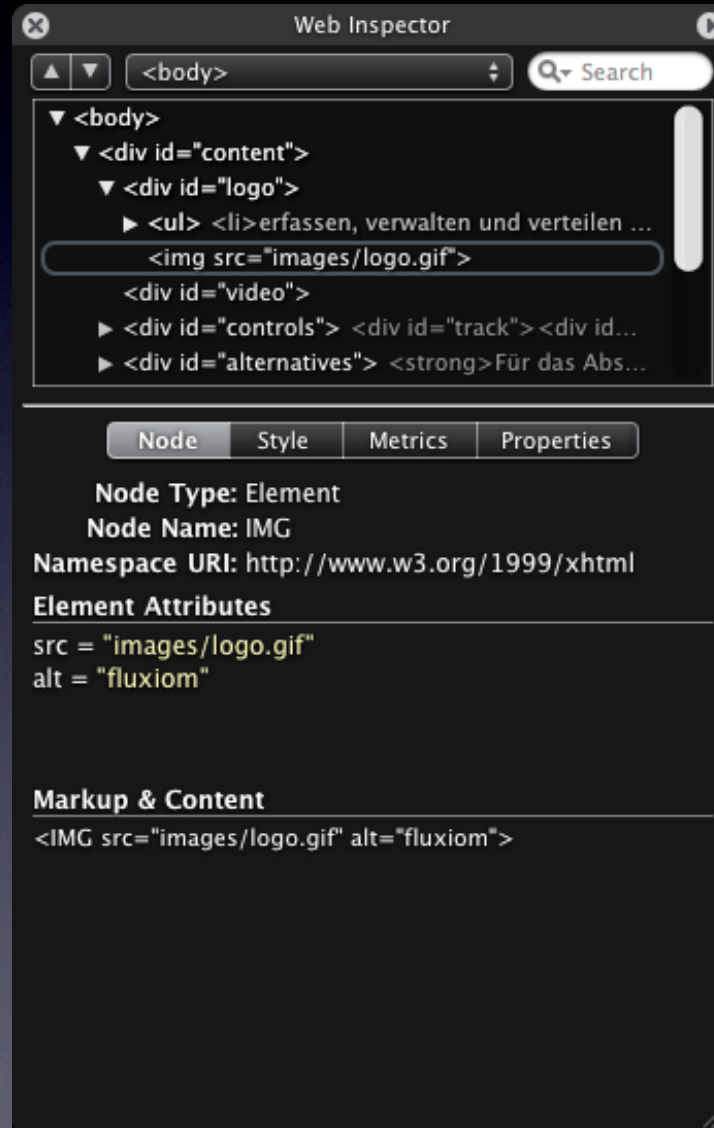
The third
rule:

Test with all
browsers
you want to
support





Safari Web Inspector





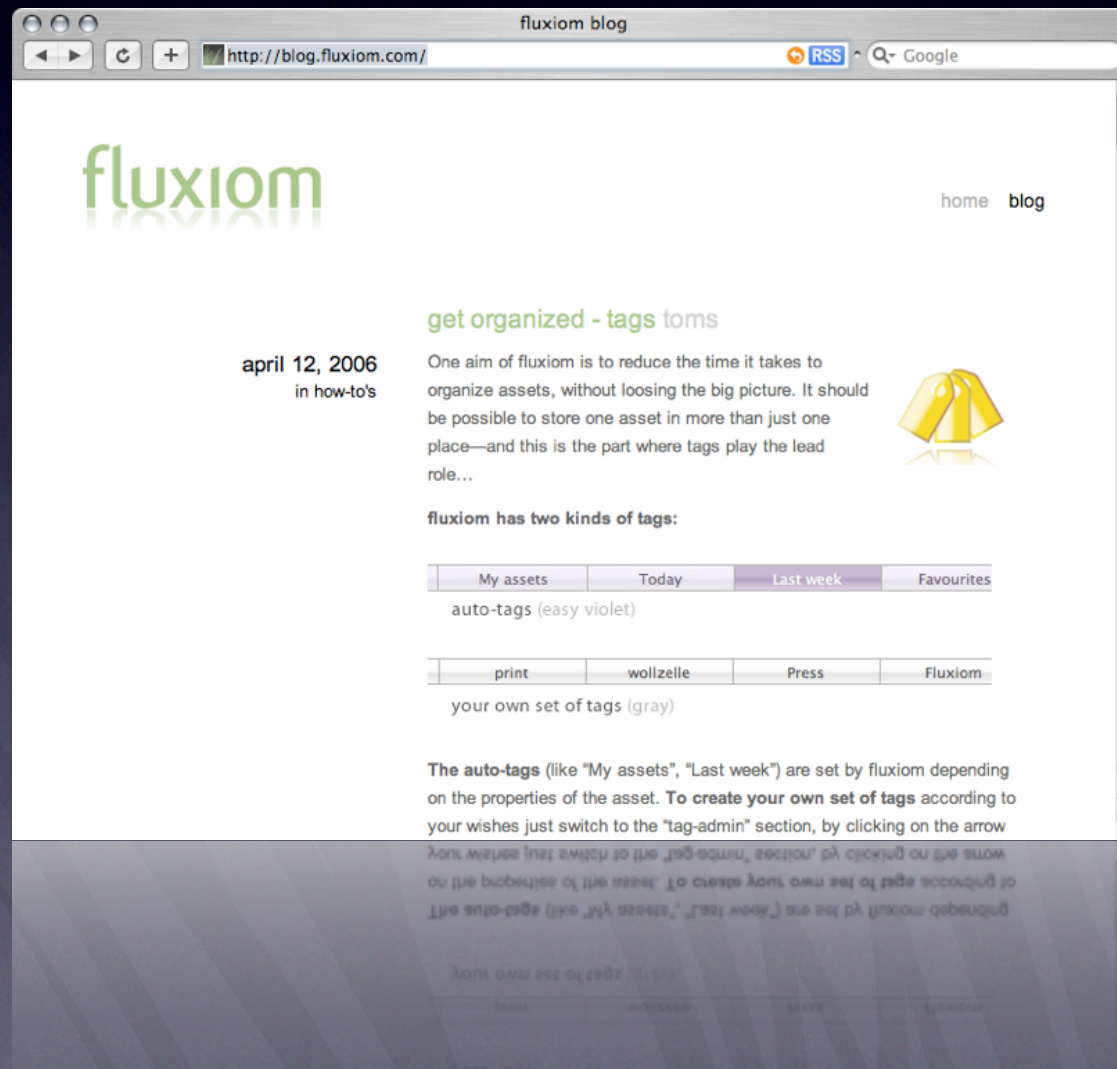
Microsoft Script Debugger for IE

- <http://blogs.msdn.com/ie/archive/2004/10/26/247912.aspx>

„It's more like Web 3.0“



blog.fluxiom.com



fluxiom

Launching April 2006

www.fluxiom.com