

HTML5

accessibility

<http://weba.im/parisweb>

me



Steve Faulkner

@stevefaulkner

[Crazy HTML Standards Bueraucrat] Accessibility by any means
necessary...

paciellogroup.com/blog/

me - not

- ★ a professional speaker
- ★ a story teller
- ★ coherent
- ★ an expert
- ★ speak French

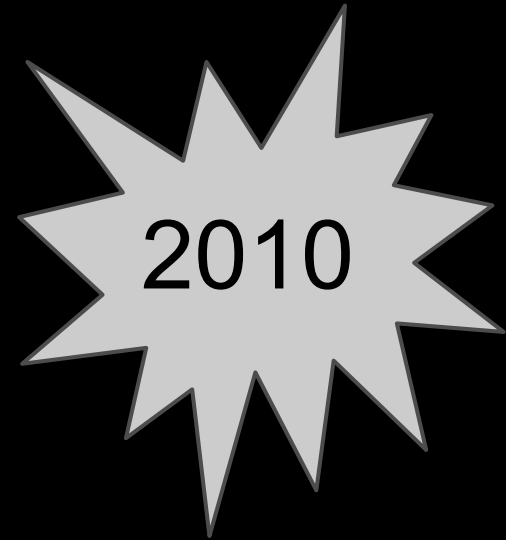


a learning journey

lets go back in time to 2010



so, is **it** ready yet?



HTML5

accessibility

that is????

Accessibility APIs

MSAA

IAccessible2

UI automation

AX

STK

A photograph of a man standing in a doorway, covered by a white sheet. The head of the man is replaced by a cutout of a different man's head, which has a beard and is smiling. The background shows a brick wall on the left and a wooden door frame.

Roles
states
properties
interaction

+ device independent
interaction = keyboard n'stuff

Accessibility API

Input device



Submit Query

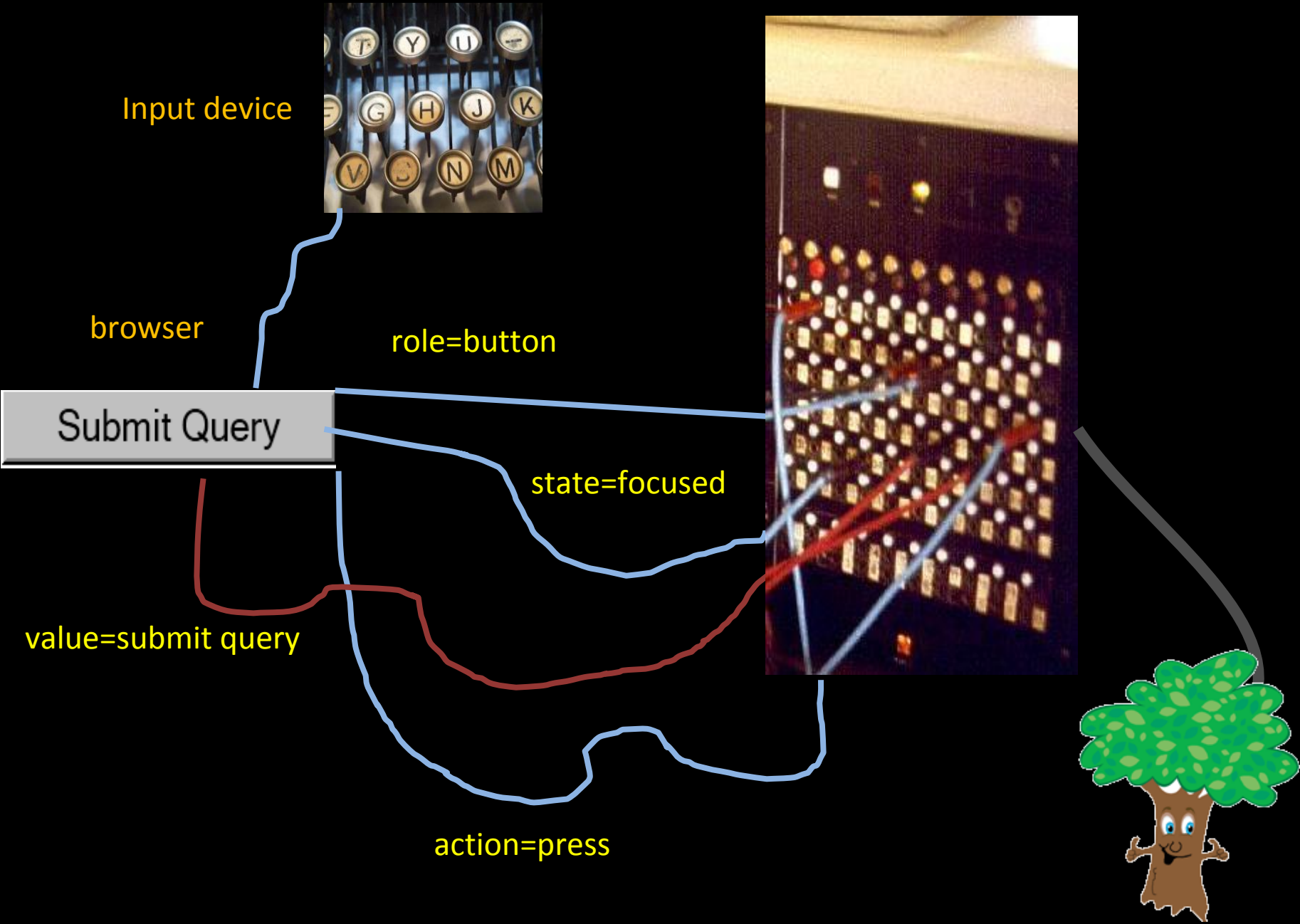
browser

role=button

state=focused

value=submit query

action=press



short answer:



2010

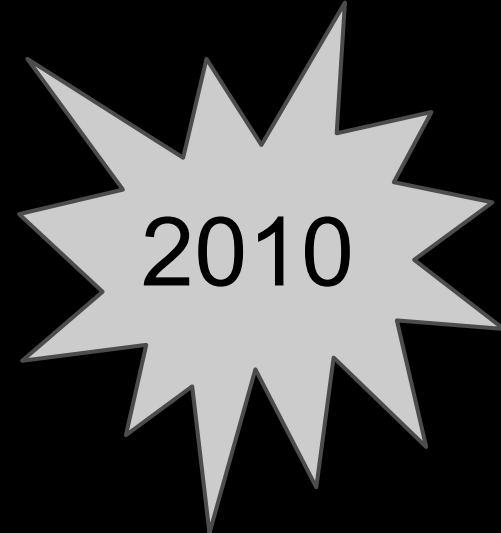
NO!

long answer is:

NOTBYAA

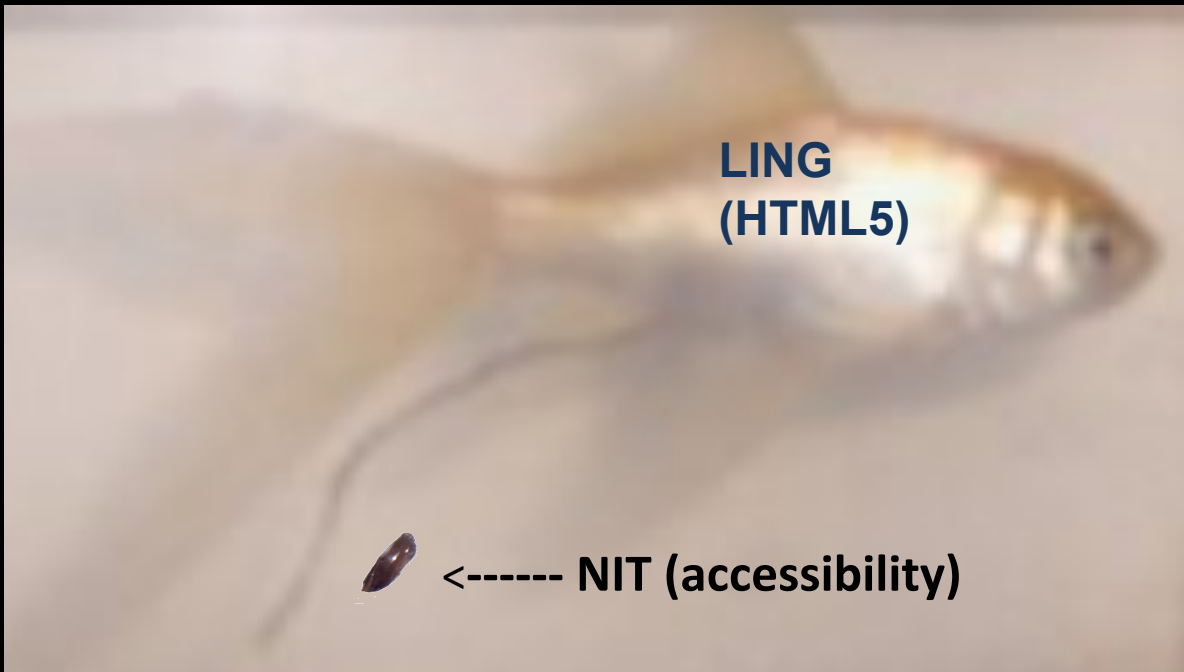
LONG

SHOT



NIT BY A LING SHIT

2010



html5accessibility.com circa Oct 2010

new HTML5 elements	 Chrome 6	 Firefox 4	 IE 9	 Opera 10.62	 Safari 5
article element	✗	✗	✗	✗	✗
aside element	✗	✗	✗	✗	✗
audio element	✗	✓✗	✓✗	✓✗	✗
canvas element	✗	✗	✓✗	✗	✗
datalist element	—	—	—	✓✗	—
details element	—	—	—	—	—
figcaption element	✗	✗	✗	✗	✗
figure element	✗	✗	✗	✗	✗
... element	✗	✗	✗	✗	✗

2010

- ugly fish



Accessible HTML5





2010



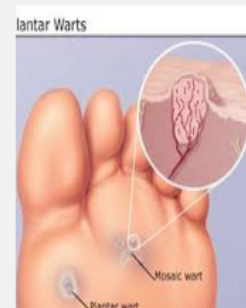
I had a dream

will be a beautiful thing

Flash forward to like NOW

new HTML5 elements	 Chrome 29	 Firefox 23	 IE 11	 Opera 16
article element	✓	✓	✗	✓
aside element	✓	✓	✗	✓
audio element	✓	✓	✓	✓
canvas element	✓	✓	✓	✓
datalist element	✓	✓	✓	✓
details element	✓	?	?	✓
dialog element	✓	?	?	?
figcaption element	✗	✓	✗	✗
figure element	✗	✓	✗	✗
footer element	✓	✓	✗	✓
header element	✗	✓	✗	✗
color <small>input</small>	✓	?	?	✓
Date <small>input</small>	✓	?	?	✓
Date and Time <small>input</small>	?	?	?	?
Local Date and Time <small>input</small>	✓	?	?	✓
E-mail <small>input</small>	?	✓	?	✗
Month <small>input</small>	✓	?	?	✓
Number <small>input</small>	✓	?	?	✓
Range <small>input</small>	✓	✓	✓	✓
Search <small>input</small>	✓	✓	✓	✓
Telephone <small>input</small>	?	?	?	✗
Time <small>input</small>	✓	?	?	✓
URL <small>input</small>	?	✓	?	✗
Week <small>input</small>	✓	?	?	✓
main element	✗	✓	✗	✗

still has a few warts



2013 - but all in all a beautifuler fish



WHY?

Browser implementers are
doing their job!



WHY?

still warty :-(




does not convey the
semantics of non
control type

elements *boo* - *hiss*

bug filed

HTML5 structural elements have UI

For users of aural UIs element description is announced and are navigable

`<article>`

`<aside>`

`<footer>`

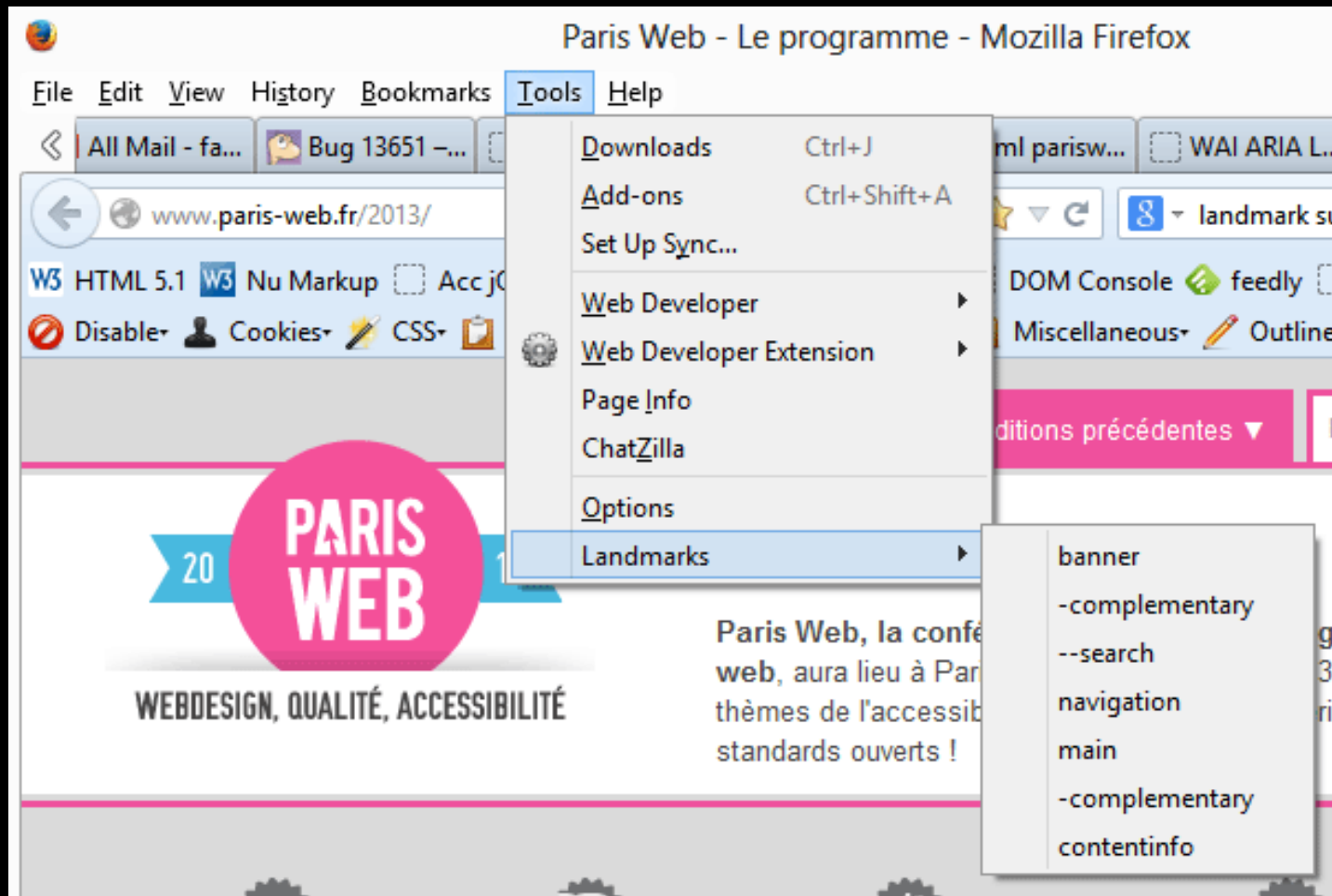
`<header>`

`<main>`

`<nav>`

`<section>`

They can also benefit keyboard users



Enabling landmark-based keyboard navigation in Firefox

on the topic of divitis

`<div>` (*and* ``)

have no default
semantics

This is a good thing!

on the topic of divitis



Your average tweet contains
7 <divs> and 27 s

on the topic of divitis

Your average twitter page:

DOM tree for: Twitter

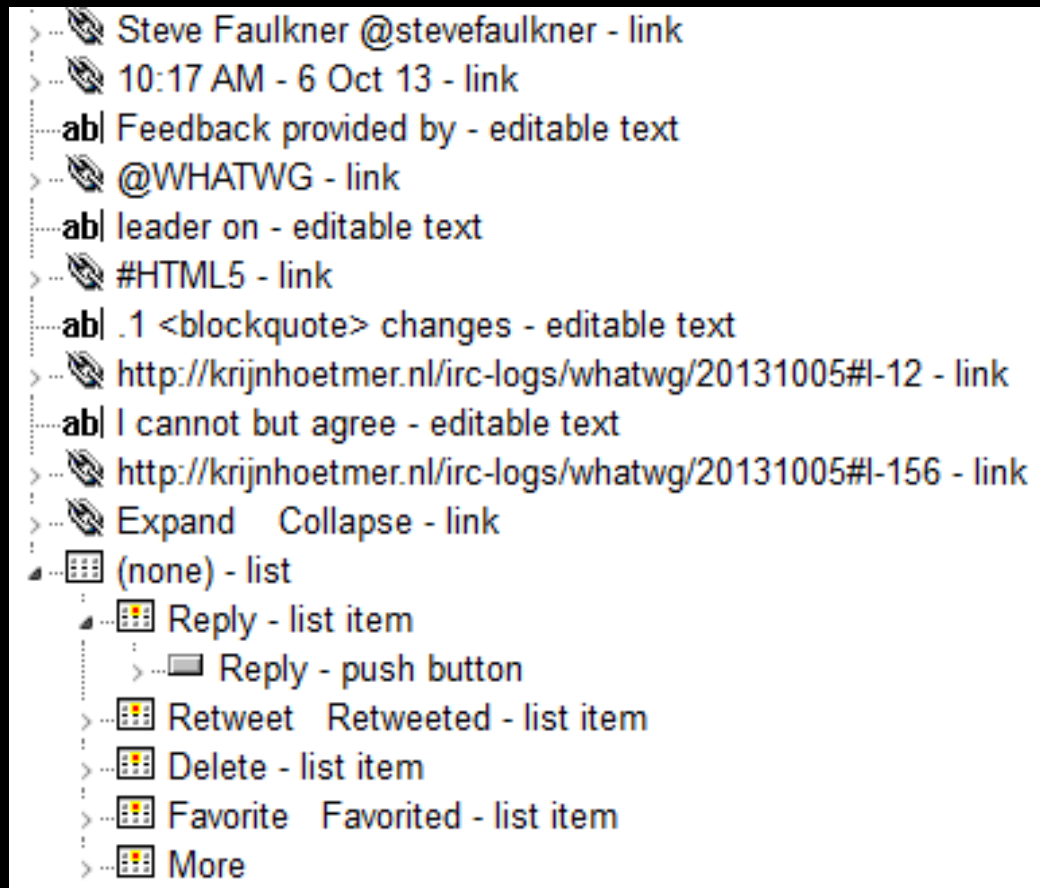
address: <https://twitter.com/>

max deep of a nested Element: 22

```
| HTML: 1 | HEAD: 1 | META: 4 | TITLE: 1 | LINK: 5 | STYLE: 4  
| SCRIPT: 9 | BODY: 1 | DIV: 753 | H2: 7 | P: 102 | BUTTON:  
90 | H1: 1 | SPAN: 666 | UL: 73 | LI: 342 | A: 461 | I: 203 |  
FORM: 10 | LABEL: 25 | INPUT: 48 | H4: 6 | IMG: 95 | S: 106 |  
B: 321 | SMALL: 136 | STRONG: 25 | BR: 6 | TEXTAREA: 6 | H3:  
24 | OL: 3 | TABLE: 3 | THEAD: 3 | TR: 29 | TH: 3 | TBODY: 3  
| TD: 52 | HR: 3 | FIELDSET: 2 | IFRAME: 1 |
```

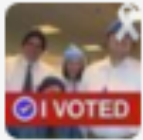
from forest to trees

<phluff> is stripped in the accessibility tree



HTML5 `<section>`

Unlike Divitis `sectionorrhea` has a negative effect on users.



Sam J @mixolydian

9 Jan

@stevefaulkner Unlabeled regions created by `<section>` and `<article>` elements are annoying especially as they appear in the region nav order.

[Details](#)



Léonie Watson @LeonieWatson

9 Jan

@stevefaulkner Utterly bloody irritating! `<section>` maps to ARIA region. Been meaning to look into possible change. Possible?

[Details](#)

Page with 254 section elements

The image shows a webpage layout with 254 section elements highlighted by yellow boxes. The page is divided into three main columns. The left column is dark grey and contains a list of links and a heading. The middle column is green and contains two sections: 'Why Terminix' with a person icon and 'Our services' with a shield icon. The right column is also green and contains a 'Manage account' section with a gear icon. The elements are labeled with terms like 'region start', 'region end', 'heading level 2', 'link', 'bullet', and 'list X items'.

Left Column (Dark Grey):

- region start
- region start
- Have you seen a pest? heading level 2
- list X items
- bullet link Get a solution
- bullet link 866.319.6528
- bullet link Request a call from us
- region end
- region end

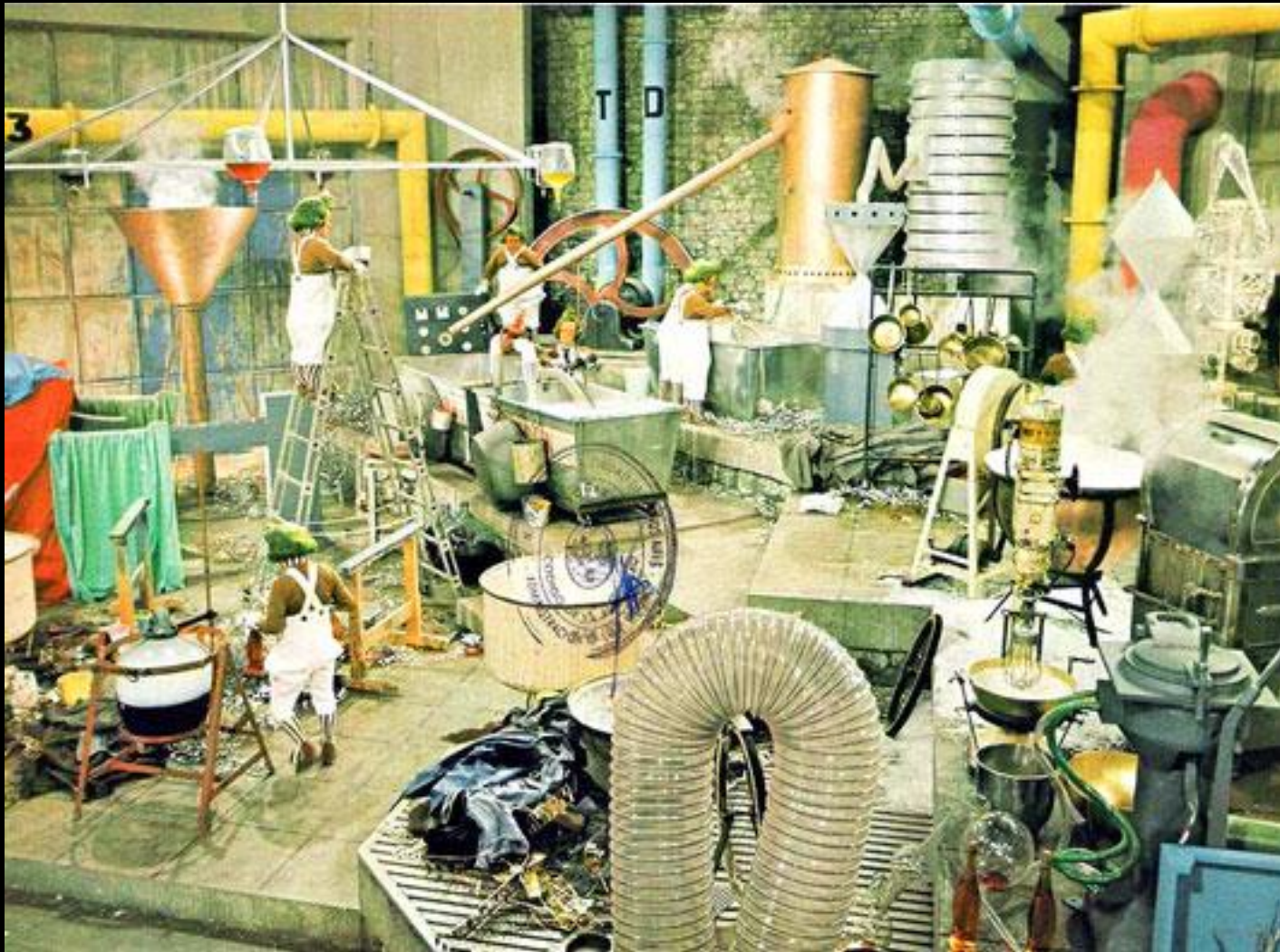
Middle Column (Green):

- region start
- region start
- region start
- region end
- Why Terminix® heading level 2
- See what makes Terminix different.
- region end
- region end
- region start
- link Learn more
- region end
- region end

Right Column (Green):

- region start
- region end
- region start
- region start
- region start
- region end
- region start
- Manage account heading level 2
- Pay bills, schedule services and more.
- region end
- region end
- link Log in
- region start
- region end
- region end

what tools are at our disposal?



1. Change definition/author requirements

“Authors **must not** use section unless it contains a heading.”



Validity constraints on <section>

2. Change the accessibility mapping

Currently browsers **must** implement

`<section>` has a `role=region`

could add

if `<section>` has an accessible name provided by `aria-label`, `aria-labelledby` or non-empty child heading (`h1-h6`) element

3. Evangelize

Seybold
SEMINARS SAN FRANCISCO 2003



*STEP RIGHT UP, FOLKS!
WITNESS AMAZING FEATS OF BRAVURA AND
DERRING-DO! WATCH AS BUNK MARKUP
VANISHES BEFORE YOUR VERY EYES!*

Why tables for layout is stupid: problems defined, solutions offered

Tables existed in HTML for one reason: To display tabular data. But then border="0" made it possible for designers to have a grid upon which to lay out images and text. Still the most dominant means of designing visually rich Web sites, the use of tables is now actually interfering with building a better, more accessible, flexible, and functional Web. Find out where the problems stem from, and learn solutions to create transitional or completely table-less layout.

Bill Merikallio ; Scott Design, Inc.
Adam Pratt ; Adobe Systems Incorporated

[International Versions](#)

We're going global! If English isn't your native

next >

Who finds the thought of layout tables distasteful?




Why tables for layout is stupid

circa 2003

2013

Google
France

<td>		x	
		<td> Surfez encore plus vite	
		<td> Installer Google Chrome	

<td>	<td>	<td>
------	------	------

Recherche Google

J'ai de la chance



		October 2010						
Week	Mon	Tue	Wed	Thu	Fri	Sat	Sun	
39	27	28	29	30	1	2	3	
40	4	5	6	7	8	9	10	
41	11	12	13	14	15	16	17	
42	18	19	20	21	22	23	24	
43	25	26	27	28	29	30	31	
44	1	2	3	4	5	6	7	



Email

URL



?



?



?



?



?

when will browsers
 implement HTML5 UI
 features in a way that
 developers will want to
 use them?

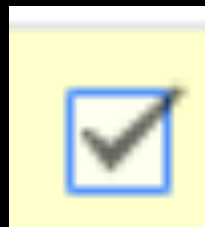
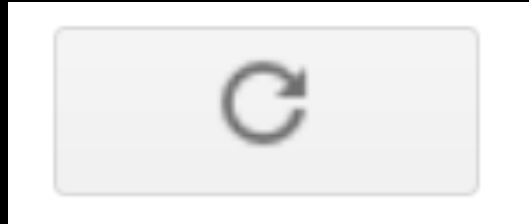
Progress:

You have to specify a value

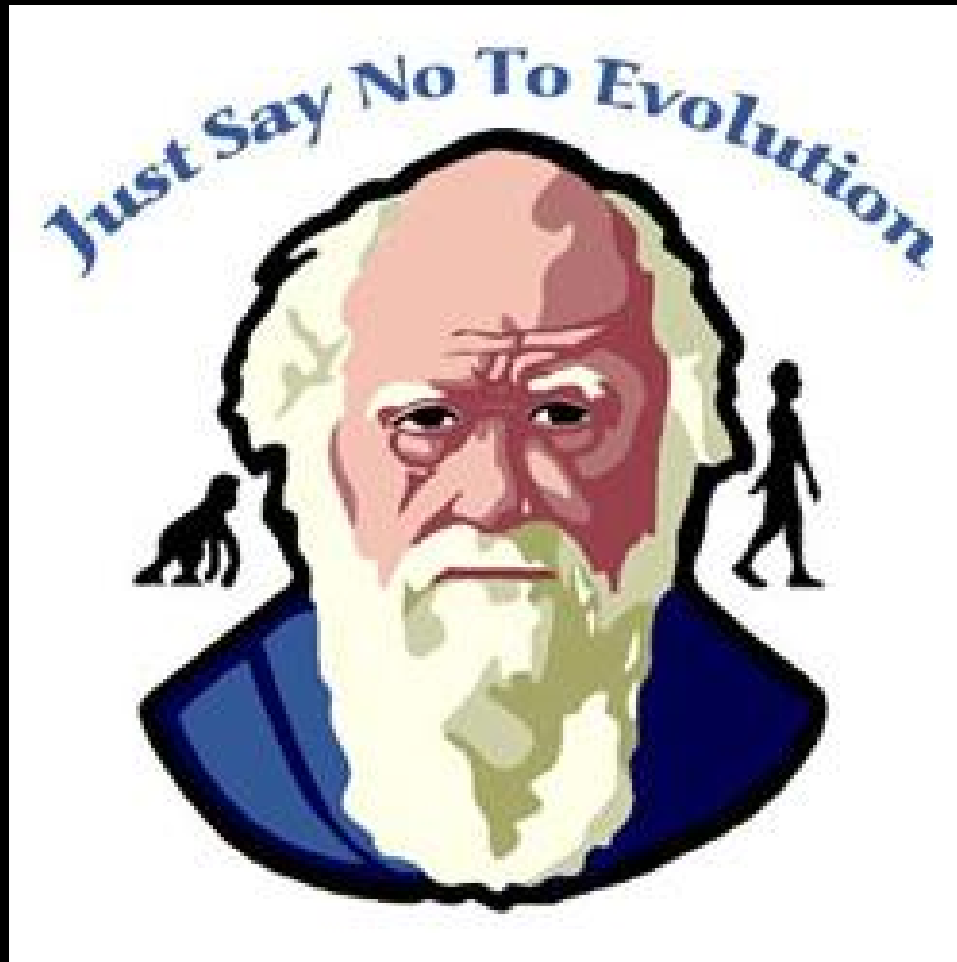
Please fill out this field.

HTML Accessibility Challenges

Developers want to **re-make** stuff that exists in HTML



Just Say No - does not work

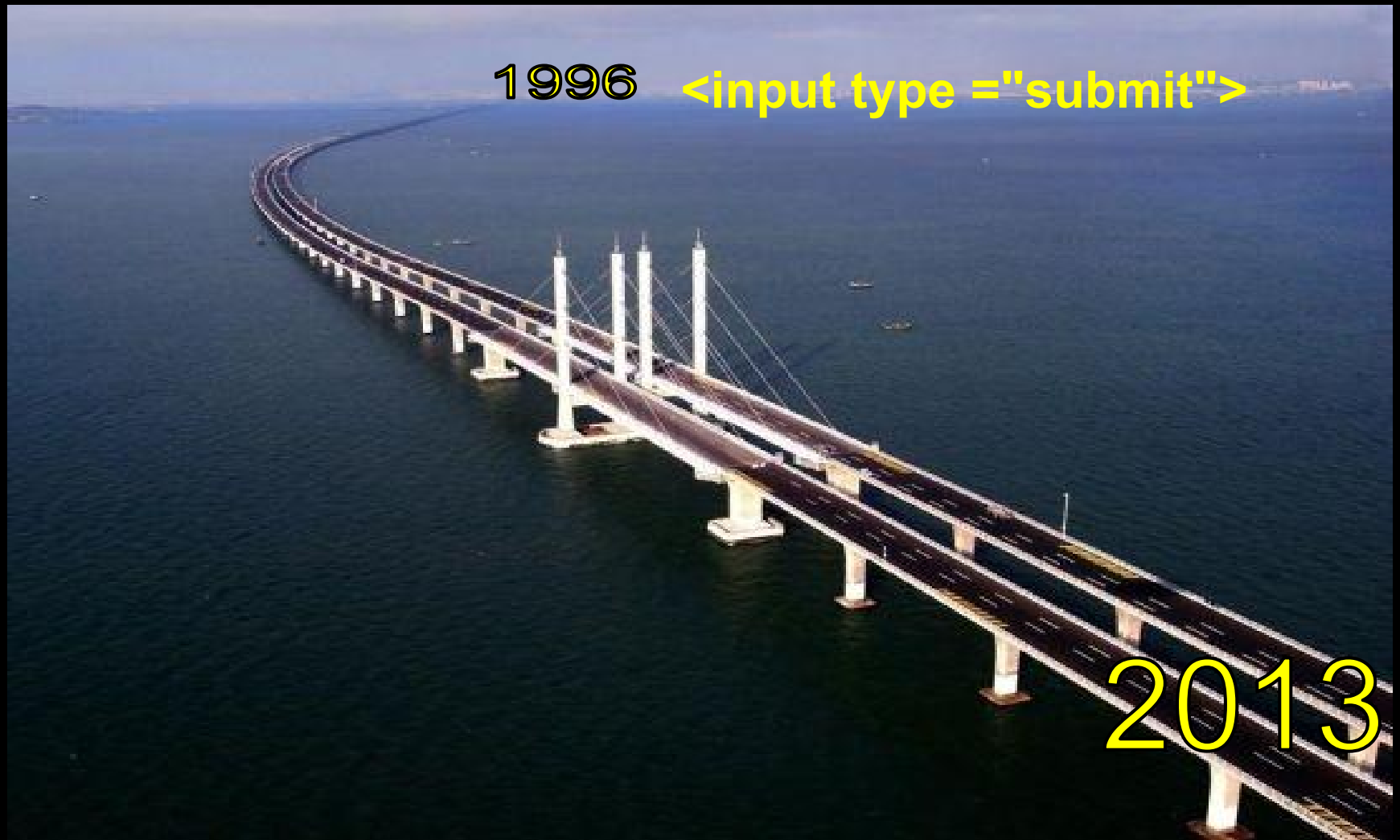


ARIA- attributes to add accessibility information

- Use of ARIA in HTML is defined in HTML5 Section [3.2.7 WAI-ARIA](#)



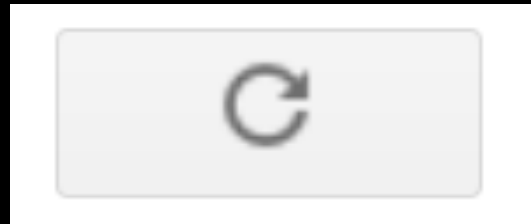
ARIA - a bridging technology



1996

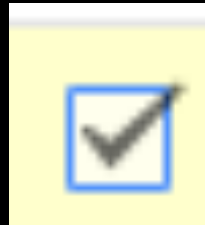
2013

2013



```
<div tabindex="0" role="button" act="20" class="T-I
  J-J5-Ji nu T-I-ax7 L3" style="-moz-user-select:
none;" aria-label="Refresh" data-tooltip="Refresh">
  <div class="asa"><span class="J-J5-Ji ask">&nbsp;
    </span>
  <div class="asf T-I-J3 J-J5-Ji"></div>
  </div>
</div>
```

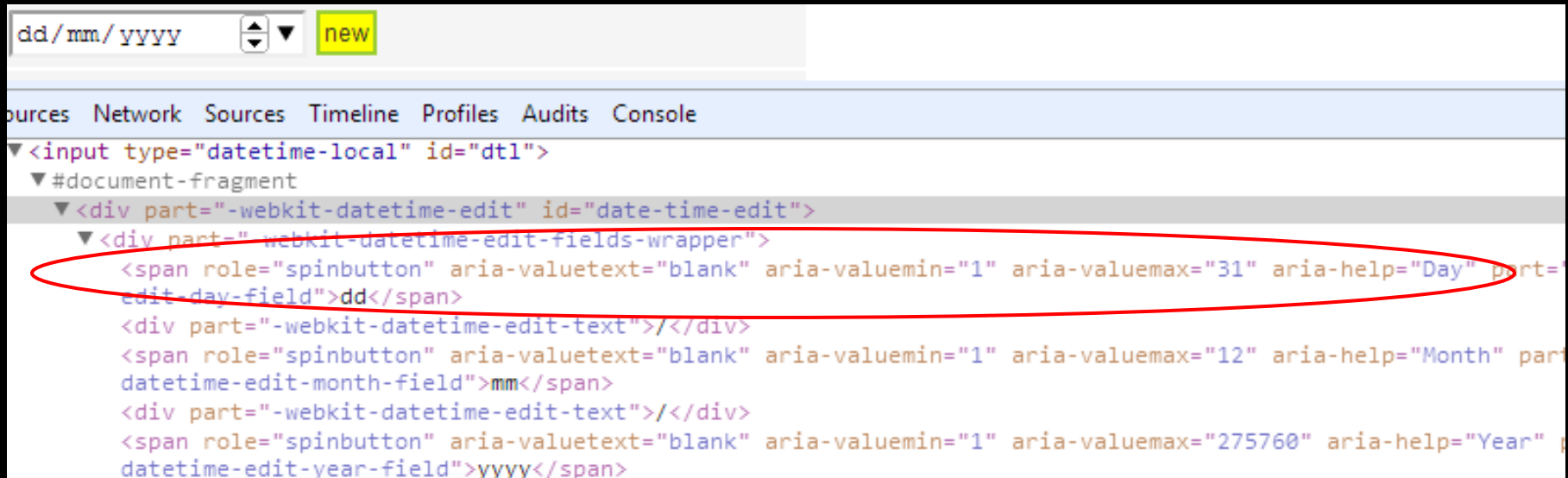
2013



```
<div tabindex="0" aria-checked="false" dir="ltr"
aria-labelledby=":2n4" role="checkbox" class="oZ-jc
T-Jo J-J5-Ji" id=":235">
<div class="T-Jo-auh"></div>
</div>
```


2013

ARIA used in browser shadow DOM



```
dd/mm/yyyy
new

Sources Network Sources Timeline Profiles Audits Console
▼ <input type="datetime-local" id="dt1">
  ▼ #document-fragment
    ▼ <div part="-webkit-datetime-edit" id="date-time-edit">
      ▼ <div part="-webkit-datetime-edit-fields-wrapper">
        <span role="spinbutton" aria-valuetext="blank" aria-valuemin="1" aria-valuemax="31" aria-help="Day" part="datetime-edit-day-field">dd</span>
        <div part="-webkit-datetime-edit-text"></div>
        <span role="spinbutton" aria-valuetext="blank" aria-valuemin="1" aria-valuemax="12" aria-help="Month" part="datetime-edit-month-field">mm</span>
        <div part="-webkit-datetime-edit-text"></div>
        <span role="spinbutton" aria-valuetext="blank" aria-valuemin="1" aria-valuemax="275760" aria-help="Year" part="datetime-edit-year-field">yyyy</span>
```

The rules of ARIA



First rule of ARIA use

If you *can* use a native HTML element or attribute with the semantics and behaviour you require **already built in**, instead of re-purposing an element and adding an ARIA role, state or property to make it accessible, **then do so**.

First rule of ARIA use

Under what circumstances may this not be possible?

- If the visual design constraints rule out the use of a particular native element, because the element cannot be styled as required.
- If the feature is not currently available in HTML.
- If the feature is available in HTML [HTML5] but it is not implemented or it is implemented, but accessibility support is not.

Second rule of ARIA use

Do not change native semantics, unless you really have to.

For example: Developer wants to build a heading that's a button.

Do **not** do this:

```
<h1 role=button>heading button</h1>
```

Do this:

```
<h1><button>heading button</button></h1>
```

Or if you can't possibly use the correct element, do this:

```
<h1><span role=button>heading  
button</span></h1>
```

Third rule of ARIA use

All interactive ARIA controls must be usable with the keyboard.

If you create a widget that a user can click or tap or drag or drop or slide or scroll, a user must also be able to navigate to the widget and perform an equivalent action using the keyboard.

All interactive widgets must be scripted to respond to standard key strokes or key stroke combinations where applicable.

For example if using `role=button` the element must be able to receive focus and a user must be able to activate the action associated with the element using both the enter (on WIN OS) or return (MAC OS) and the space key.

ARIA validation

- Use of ARIA in HTML<5 is non-conforming and probably always will be.
- It doesn't make any difference, it still works
- The easiest method is to use the HTML5 DOCTYPE with ARIA markup and validate using the W3C Nu Markup Validation Service.

```
<!DOCTYPE html>
```



Nu Markup Validation Service



<http://validator.w3.>

stuff to read (*that I scribble a11y on*)

→ [HTML 5.1 Nightly](#)

→ [HTML to Platform](#)

[Accessibility APIs](#)

[Implementation Guide](#)

→ [Using WAI-ARIA in HTML](#)

thank you!

www.twitter.com/stevefaulkner

www.paciellogroup.com/blog

www.html5accessibility.com

sfaulkner@paciellogroup.com

this talk: <http://weba.im/parisweb>

