



# Upgrade a HTML 5 e CSS 3

Bolzano 25-26 ottobre 2010

(C) 2010 Chris Mair <chris@l006.org>

Creative Commons Attribution 2.5 Italy License



# HTML 5



## Intro

# Browser: diffusione

<b>Fonte</b>	<b>IE</b>	<b>Firefox</b>	<b>Chrome</b>	<b>Safari</b>	<b>Opera</b>
Net Applications	56.65	22.96	7.98	5.27	2.39
W3Counter	43.30	30.40	11.40	5.5	1.60
Stat Counter	49.78	31.50	11.54	4.42	2.03
Clicky	48.37	29.95	11.82	8.43	1.24
Wikimedia	46.45	29.33	9.00	5.49	3.19
StatOwl	62.68	20.51	6.99	8.83	0.36

Fonte: Wikipedia settembre 2010

[http://en.wikipedia.org/wiki/Usage\\_share\\_of\\_web\\_browsers#Summary\\_table](http://en.wikipedia.org/wiki/Usage_share_of_web_browsers#Summary_table)

# Browser: zoologia

- Trident (Microsoft)
  - 6 (XP!), 7/8 → 9 (uscita imminente!)
- Gecko (Open Source, Mozilla, Google, Comunity)
  - Firefox 3.6 → 4.0 (uscita imminente)
- Webkit (Open Source, Apple, Google, Comunity)
  - Safari 5, Chrome 7, Smartphones (iPhone, Android, BB)!
- Presto (Opera Software)
  - Opera 10.63, vari software embedded

# HTML: storia I/2

- 1997 W3C: HTML 4 “ultima versione”
- 1999 W3C: XHTML 1.0 “basato su XML”
- 2001 W3C: XHTML 1.1 “draconiano”
- ???
- 2010 stiamo ancora usando standard anni '90!

# HTML: storia 2/2

- 2004 Web Hypertext Applications Technology Working Group:  
WHAT Working Group?  
fondata da Opera, Apple e Mozilla
- 2007 W3C rinuncia a XHTML 2.0 (non uscirà!) e collabora con WHATWG su HTML 5

# HTML 5

- HTML 5 viene disegnato in modo da essere *backward-compatible* con browser che capiscono HTML 4 (e/o XHTML 1.0)
- HTML 5 è una collezione di funzionalità che vengono aggiunte a HTML 4
- HTML 5 ha attualmente lo stato di working draft (vedi <http://www.w3.org/TR/html5/> )
- **Tutti** lo supportano (IE dalla 9, FF dalla 4)

# Le nuove funzionalità

(che vedremo in questo corso)

- nuovi elementi semantici
- audio/video nativo
- nuovi elementi per i *form*
- il *canvas*
- dati salvati lato *client*
- **SVG**  
(non c'entra con HTML 5, ma è di interesse, ora che anche IE lo supporta)

# Doctype semplificato

```
<!DOCTYPE html PUBLIC  
  "-//W3C//DTD XHTML 1.0 Strict//EN"  
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

diventa

```
<!DOCTYPE html>
```

# <html> semplificato

```
<html xmlns="http://www.w3.org/1999/xhtml"  
      lang="en"  
      xml:lang="en">
```

diventa

```
<html lang="en">
```

# <meta> semplificato

```
<meta http-equiv="Content-Type"  
      content="text/html; charset=utf-8" />
```

diventa

```
<meta charset="utf-8">
```

# Hello HTML 5

```
<!doctype html>  
<html lang="it">  
  <head>  
    <meta charset=utf-8>  
    <title>Corso HTML 5</title>  
  </head>  
  <body>  
    <p>Hello World</p>  
  </body>  
</html>
```

# Sintassi

- HTML 5 può usare sia la sintassi HTML che prevede tag in minuscolo o maiuscolo ed elementi vuoti senza slash `<br>`, che la sintassi XHTML che prevede tag solo in minuscolo ed elementi vuoti sempre con slash `<br />`

# È valido!

`http://validator.w3.org/`



## Markup Validation Service

Check the markup (HTML, XHTML, ...) of Web documents

**Jump To:**

[Notes and Potential Issues](#)

[Congratulations · Icons](#)

This document was successfully checked as HTML5!

<b>Result:</b>	Passed, 1 warning(s)	
<b>File :</b>	<input type="text"/>	<input type="button" value="Browse..."/>
	<i>Use the file selection box above if you wish to re-validate the uploaded file webdeveloper-localhost-1288040726770.html</i>	
<b>Modified:</b>	(undefined)	
<b>Server:</b>	Mozilla/5.0 (Macintosh; U; Intel Mac OS X 10.5; en-US; rv:1.9.2.11) Gecko/20101012 Firefox/3.6.11	
<b>Size:</b>	(undefined)	
<b>Content-Type:</b>	text/html	
<b>Encoding :</b>	utf-8	<input type="text" value="(detect automatically)"/>
<b>Doctype :</b>	HTML5	<input type="text" value="(detect automatically)"/>
<b>Root Element:</b>	html	

# Risorse

- sito W3C ( <http://www.w3.org/TR/html5/> )
- ottimo libro di Mark Pilgrim:  
“Dive Into HTML 5” ( <http://diveintohtml5.org/> )  
pubblicato da O’Reilly/Google con il titolo  
“HTML 5 Up and Running”





# HTML 5

---

## Elementi semantici

# Elementi semantici

- servono a dare significato al contenuto
- `<div id="nav_bar">` sarà chiaro al webdesigner ma non al browser / smartphone / googlebot / screenreader
- viene sostituito con `<nav>`
- non ci sono stili particolari associati, ma influenzano la struttura di un documento (outline)

# I più importanti 1/2

<code>&lt;section&gt;</code>	contiene sezioni, p.e. capitoli di un libro
<code>&lt;nav&gt;</code>	contiene gli elementi per la navigazione
<code>&lt;article&gt;</code>	contiene un articolo di un giornale o blog
<code>&lt;aside&gt;</code>	contiene una nota allegata (da mostrare a lato)
<code>&lt;hgroup&gt;</code>	vale come <code>&lt;h?&gt;</code> e contiene altri <code>&lt;h1&gt;...&lt;h6&gt;</code> <sup>[1]</sup>

[1] <http://www.whatwg.org/specs/web-apps/current-work/multipage/sections.html#headings-and-sections>

# I più importanti 2/2

<code>&lt;header&gt;</code>	contiene la testata di una sezione
<code>&lt;footer&gt;</code>	contiene il piè di pagina
<code>&lt;time&gt;</code>	indica data o tempo
<code>&lt;mark&gt;</code>	contiene testo evidenziato

lista completa: <http://www.w3.org/TR/html5/index.html>

# Compatibilità (non IE)

- i browser rendono elementi ignoti come `display:inline`, ma in HTML 5 molti di questi elementi sono considerati `display=block`
- Safari e Chrome conoscono questi elementi, Firefox dalla 4 e IE dalla 9
- per essere compatibili con browser più vecchi basta assegnare `display=block` via CSS

# Compatibilità (IE)

- IE (prima della 9) non permette di applicare stili CSS a elementi ignoti e non ne considera l'annidamento
- per fortuna si può definire elementi propri via Javascript
- Remy Sharp ha uno script (molto breve) per definire tutti gli elementi nuovi!
- <http://code.google.com/p/html5shim>

# Esempio

```
<nav>
  <ul><li><a href="contatti.html">Contatti</a></li>
    <li><a href="sitemap.html">Sitemap</a></li></ul>
</nav>

<article>
  <header>
    <h1>Un altro post!</h1>
    <p>Dal nostro corrispondente su venere.</p>
  </header>
  <section>
    <h1>Oggi fa caldo!</h1>
    <p>Che afa su venere...</p>
  </section>
</article>

<footer><p>Questa pagina e` stata <mark>modificata</mark> il
  <time datetime="2010-10-25T09:00:00+02:00" pubdate>
  25 ottobre 2010, ore 9:00</time></p>
</footer>
```

# Esempio/CSS

```
article {
    margin: 10px;
    border-style: solid;
    border-width: 1px;
    border-color: red;
}
header {
    background-color: #FFFF99;
}
section {
    padding-left: 20px;
}
```

# Risultato

Safari ( sa HTML 5 )

- [Contatti](#)
- [Sitemap](#)

## Un post!

Dal nostro corrispondente su marte.

**A**

la prima lettera del alfabeto.

**B**

la seconda lettera del alfabeto.

## Un altro post!

Dal nostro corrispondente su venere.

**Oggi fa caldo!**

Che afa su venere...

Questa pagina e` stata modificata il 25 ottobre 2010, ore 9:00

Firefox 3.6 ( non sa HTML 5 )

- [Contatti](#)
- [Sitemap](#)

## Un post!

Dal nostro corrispondente su marte.

**A**

la prima lettera del alfabeto.

**B**

la seconda lettera del alfabeto.

## Un altro post!

Dal nostro corrispondente su venere.

**Oggi fa caldo!**

Che afa su venere...

Questa pagina e` stata modificata il 25 ottobre 2010, ore 9:00

# Esempio/css → fixed

```
article {
  display: block;
  margin: 10px;
  border-style: solid;
  border-width: 1px;
  border-color: red;
}
header {
  display: block;
  background-color: #FFFF99;
}
section {
  display: block;
  padding-left: 20px;
}
```

# Un esempio completo

- Il “blog” di Mark Pilgrim
- variante XHTML 1.0

`http://diveintohtml5.org/examples/blog-original.html`

- variante HTML 5

`http://diveintohtml5.org/examples/blog-html5.html`

- vedi stampe extra distribuite!



# HTML 5



## Audio e Video

# Motivazione

- Nel 1997 il video sul web non esisteva, nel 2010 è onnipresente - HTML dovrebbe gestirlo nativamente.
- Il Flash plugin non può essere la fine della storia: perchè dovrei avere un piccolo browser fatta da una software house unica (l'Adobe) dentro ogni altro browser fatto da chiunque?
- Altri plugin non si sono abbastanza diffusi.

# La buona notizia

- Tutti supportano `<video>` e `<audio>`
- IE dalla 9
- gli altri già nelle versioni correnti
- anche gli Smartphone (iPhone, Android)
- Safari e Chrome da tempo

# La cattiva notizia

- Il formato!
- container MP4 con codec video H.264 e codec audio AAC (MPEG LA - royalty free solo per contenuti gratis)
- container Ogg con codec video Theora e codec audio Vorbis (comunitario - royalty free)
- container WebM con codec video VP8 e codec audio Vorbis (Google - royalty free)

# Supporto *out-of-the-box*

Fonte	IE	Firefox	Chrome	Safari	Opera
MP4 H.264+AAC	9		5+	3+	
Ogg Theora+Vorbis		3.5+	5+		10.5+
WebM VP8+Vorbis		4	6		10.6+

Fonte: Dive into HTML 5

<http://diveintohtml5.org/video.html#what-works>

# La soluzione

- Offrire video in più di un formato!
- Il libro di Mark Pilgrim elenca anche il software e i passi per fare *l'encoding* in diversi formati
- **Firefogg** oppure **ffmpeg2theora** per Ogg
- **Handbrake** per MP4
- **ffmpeg** per WebM

# Esempio

- Il primo minuto di “Big Buck Bunny” codificato in 3 modi:

[http://camendesign.com/code/video\\_for\\_everybody/test.html](http://camendesign.com/code/video_for_everybody/test.html)

```
<video width="640" height="360" id="vid" controls>
  <source src="big_buck_bunny.mp4" type="video/mp4">
  <source src="big_buck_bunny.webm" type="video/webm">
  <source src="big_buck_bunny.ogv" type="video/ogg">
  Oops - no video per te
</video>
```

# Se tutto fallisce...

- I browser prendono la prima sorgente utile, solo se non ce n'è, considerano il contenuto di `<video>` che viene dopo `<source>`
- può essere una semplice scritta oppure ... può essere un `<object>` che embedda un plugin (p.e. flash)
- Per un esempio di fallback a flash vedi:  
[http://camendesign.com/code/video\\_for\\_everybody/test.html](http://camendesign.com/code/video_for_everybody/test.html)

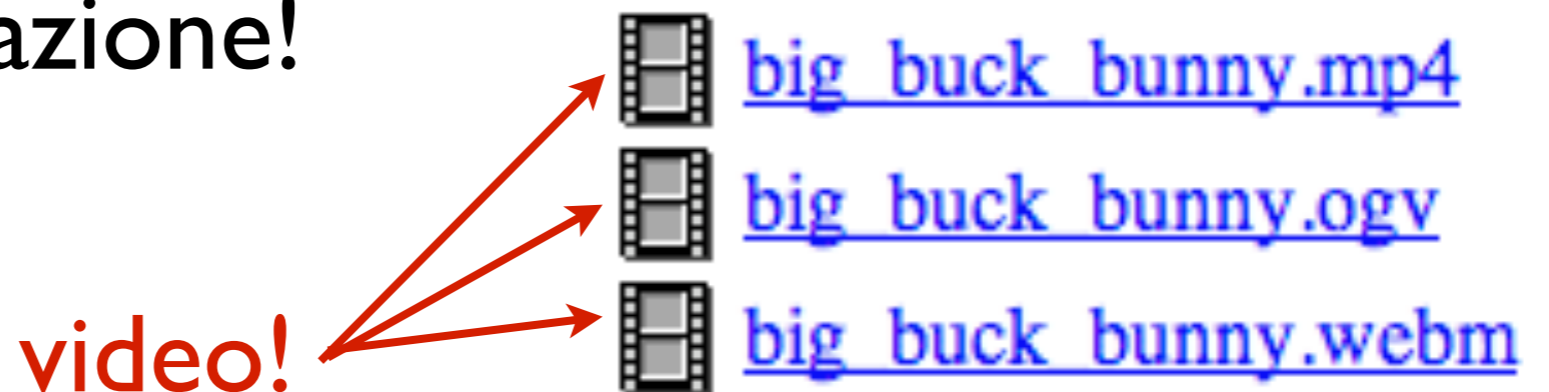
# Se lo provate a casa...

E cruciale che i file video siano forniti dal web-server con i tipi MIME corretti!

Se usate Apache dovete mettere qualcosa come

```
AddType video/ogg      .ogg  
AddType video/mp4      .mp4  
AddType video/webm     .webm
```

nella configurazione!



# Attributi di `<video>`

- `autoplay` - parte da solo
- `controls` - mostra comandi *built-in*
- `height` e `width`
- `loop` - ciclo
- `preload` - inizia subito a caricare

# E streaming?

- Streaming è un pò un cantiere aperto...
- Firefox e IE hanno rudimentale supporto per streaming - eseguono *byte range requests* (codice HTTP 206) per caricare il video un pezzo alla volta - nei miei test ho trovato che funziona molto bene!
- Safari ha una soluzione basata sempre su HTTP ma richiede la segmentazione del video su server (file .M3U8)

# Un altro esempio

- “Sintel” è un film di 15 minuti della Blender Foundation (Blender è un ben noto programma di grafica 3D Open Source) disponibile sotto licenza CC sia in MP4 che in Ogg e in dimensioni diverse fino a 2048x872px! Ideale per test più esigenti.

<http://www.sintel.org/wp-content/content/download.html>

# Controllo del player

```
<video width="1024" height="436" autoplay id="vid" controls>  
  <source src="sintel-1024-stereo.mp4" type='video/mp4'>  
  <source src="sintel-1024-stereo.ogv" type='video/ogg'>  
</video>
```

```
<script type="text/javascript">  
  function ff() {  
    vid = document.getElementById("vid");  
    vid.currentTime = 12*60+26;  
  }  
</script>
```

```
<form>  
  <input value="salta ai titoli di coda"  
    type="button" onclick="ff()">  
</form>
```

# E infine...

I film di Youtube sono disponibili in  
HTML 5!

<http://www.youtube.com/html5>

# HTML 5

---

Elementi per i *form*

# Nuovi elementi

- Tutti i browser mostrano come campo testuale un campo con tipo sconosciuto (anche IE6!), perciò si può introdurre i nuovi tipi senza problemi in form esistenti.
- Tra i browser in circolazione correntemente il supporto migliore viene da Opera.
- Vedi l'esempio 06 sulle stampe distribuite!

# HTML 5



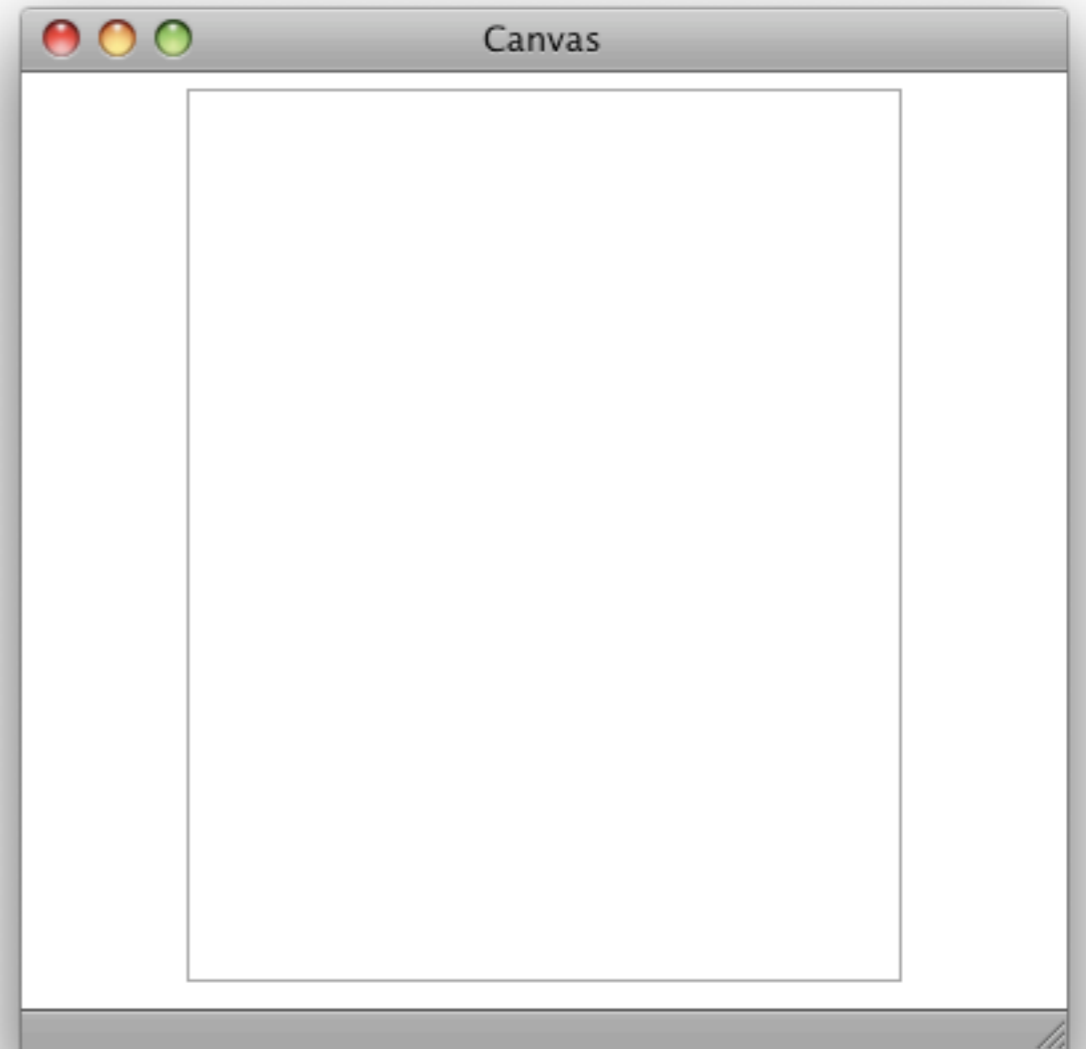
## II *canvas*

# <canvas>

- Il canvas permette di disegnare su una bitmap in due dimensioni da un'API Javascript
- Tutti i browser in circolazione supportano già il canvas
- Per IE 7/8 serve una libreria Javascript  
( <http://code.google.com/p/explorercanvas/> )  
IE 9 lo supporta nativamente
- Utile per disegnare grafici in modo dinamico, per animazioni, giochi e temo in futuro molto per le pubblicità...

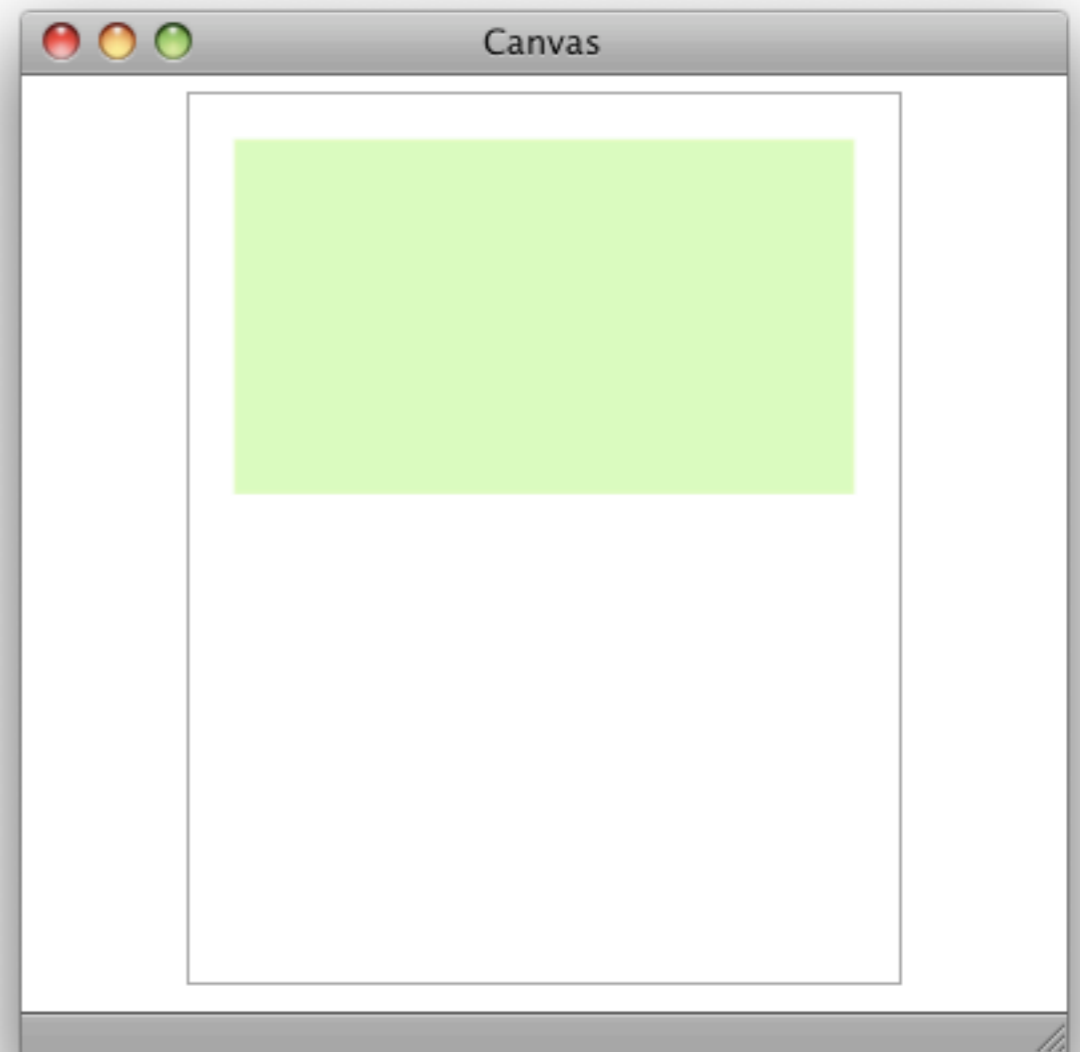
# Esempio 1/6

```
<div>  
<canvas id="c1"  
  width="320"  
  height="400">  
</canvas>  
</div>
```



# Esempio 2/6

```
var c1 = document.getElementById("c1");  
  
// ottieni il "drawing context"  
var con = c1.getContext("2d");  
  
// riempi un rettangolo color verdino  
con.fillStyle = "#CCFFAA";  
con.fillRect(20, 20, 280, 160);
```



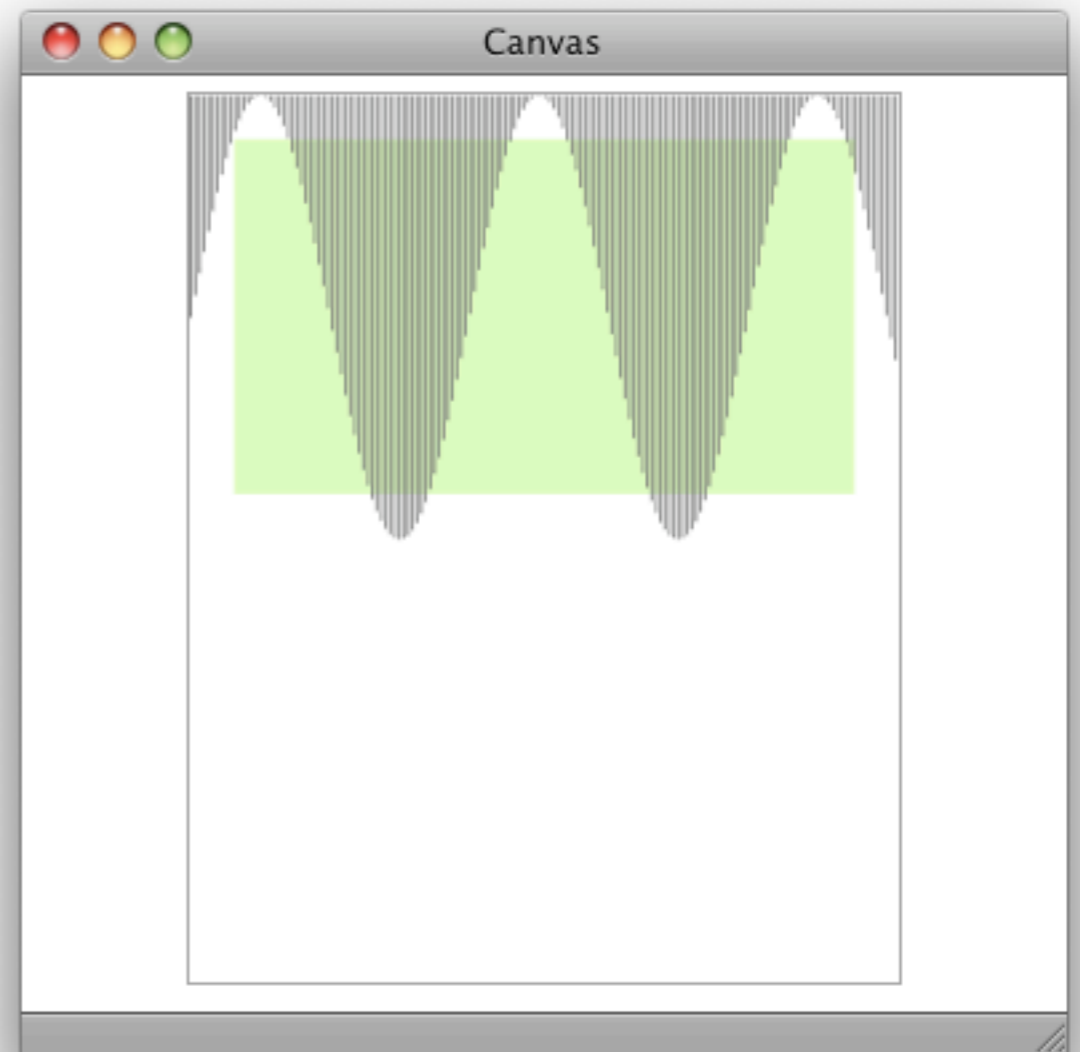
# Esempio 3/6

```
// traccia una serie di righe verticali
// (stroke() e` per renderli)

con.strokeStyle = "#666666";

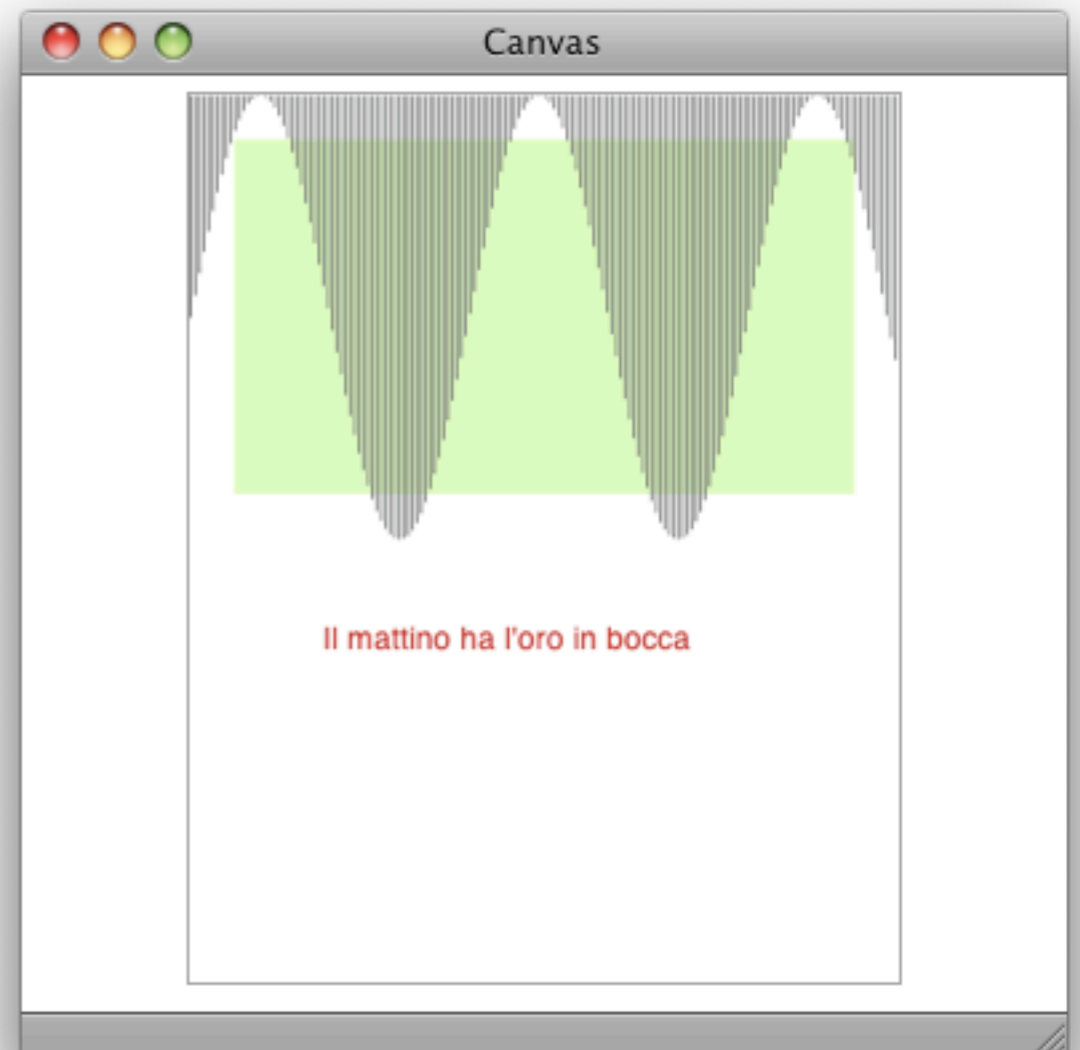
for (x = 0; x < 320; x += 2) {
  con.moveTo(x + 0.5, 0.5);
  con.lineTo(x + 0.5,
             -Math.sin(0.05 * x)
             *100.0 + 100.5);
}

con.stroke();
```



# Esempio 4/6

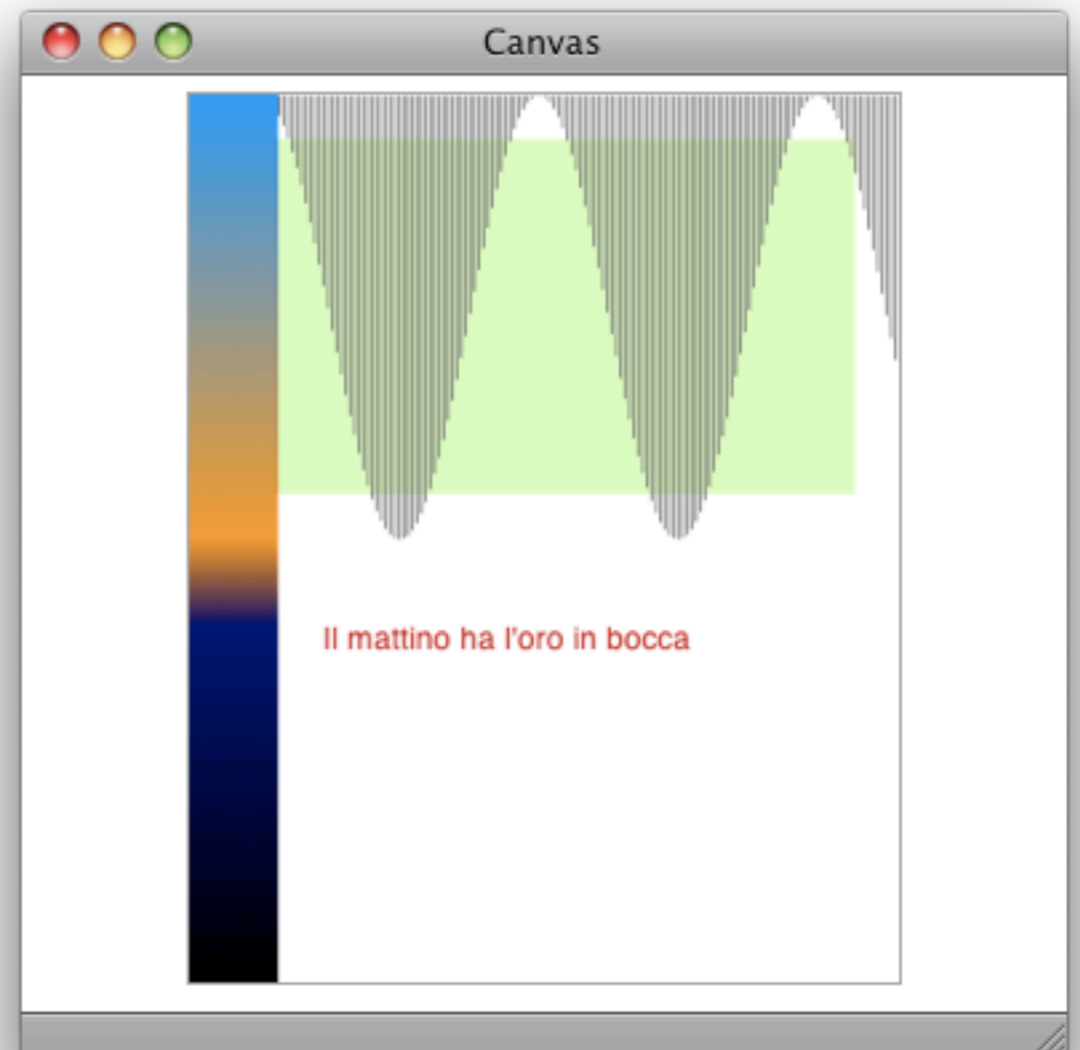
```
// disegna testo  
  
con.fillStyle = "#CC0000";  
con.font = "14px sans-serif";  
con.fillText("Il mattino ha l'oro  
in bocca", 60, 250);
```



# Esempio 5/6

```
// crea gradiente e usalo per riempire  
// un altro rettangolo
```

```
g = con.createLinearGradient(  
    0, 0, 0, 400);  
g.addColorStop(0.0, "#0088FF");  
g.addColorStop(0.5, "#FF8800");  
g.addColorStop(0.6, "#000066");  
g.addColorStop(1.0, "#000000");  
con.fillStyle = g;  
con.fillRect(0, 0, 40, 400);
```



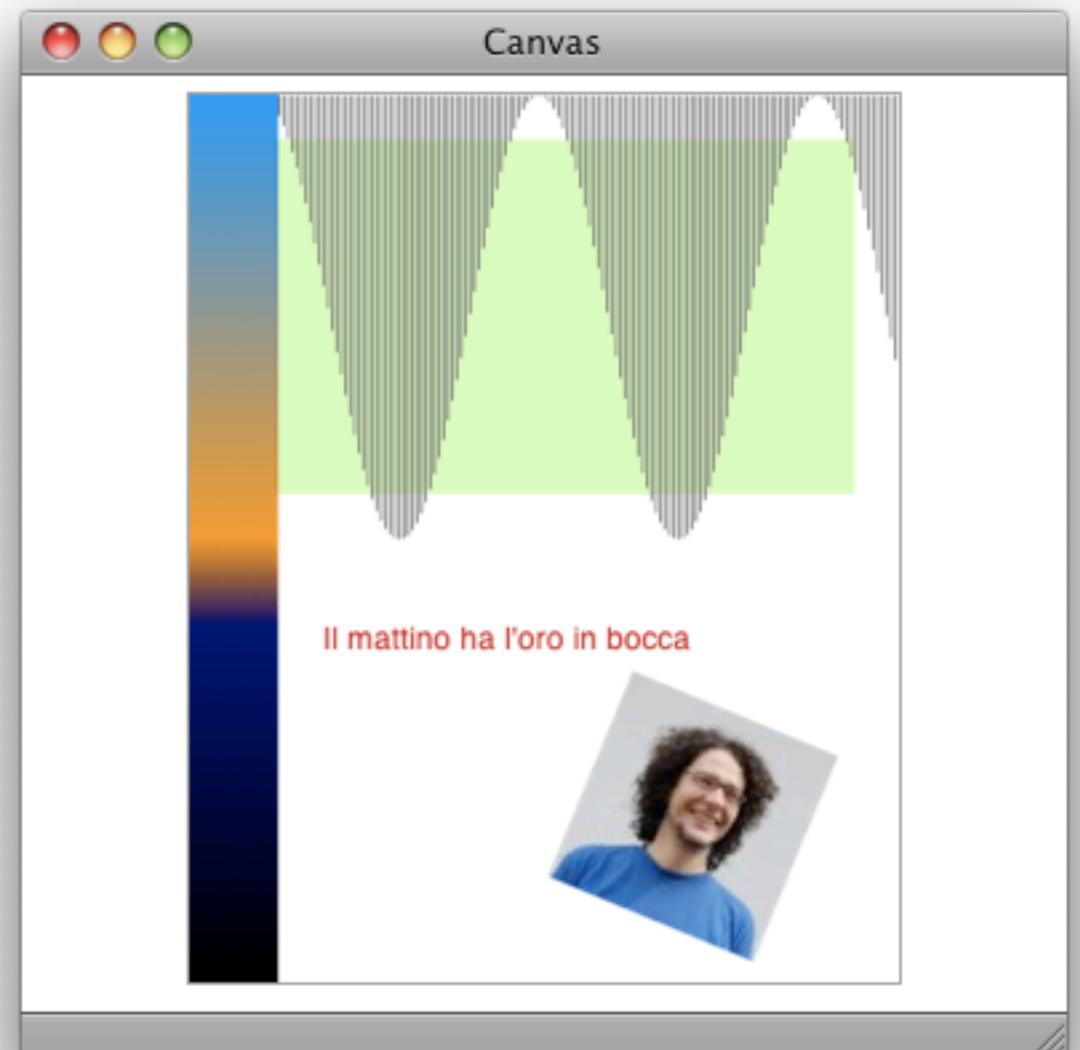
# Esempio 6/6

```
// esegui una trasformazione

con.save();
con.translate(200,260);
con.rotate(0.125 * Math.PI);

// carica un'immagine
// la funzione onload e`
// necessaria per assicurarsi
// che l'immagine e` caricata del tutto

chris = new Image();
chris.src = "chris.jpg";
chris.onload = function()
    { con.drawImage(chris, 0, 0); };
```



# risorse su `<canvas>`

- <http://diveintohtml5.org/canvas.html>
- [https://developer.mozilla.org/en/Canvas\\_tutorial](https://developer.mozilla.org/en/Canvas_tutorial)
- <http://www.whatwg.org/specs/web-apps/current-work/multipage/the-canvas-element.html>
- [https://developer.mozilla.org/en/Canvas\\_tutorial](https://developer.mozilla.org/en/Canvas_tutorial)
- <http://www.canvasdemos.com/>



# HTML 5

---

## Dati lato client

# localStorage

- permette di salvare coppie di chiave / valore localmente da Javascript
- la differenza rispetto a cookies è che **non** vengono mandati avanti e indietro con ogni richiesta HTTP
- *same origin policy*: due pagine diverse accedono ai dati solo se provengono dallo stesso server

# esempio

```
function save() {
    fname = document.getElementById("fname").value;
    lname = document.getElementById("lname").value;
    age    = document.getElementById("age").value;
    localStorage["corso_html5.fname"] = fname;
    localStorage["corso_html5.lnome"] = lname;
    localStorage["corso_html5.age"] = age;
    alert("ok");
}

function load() {
    alert(localStorage["corso_html5.fname"] + " " +
          localStorage["corso_html5.lnome"] + " (" +
          parseInt(localStorage["corso_html5.age"]) + ")");
}
```

# beyond localStorage

- Web SQL Database
- offline applications

**SVG**



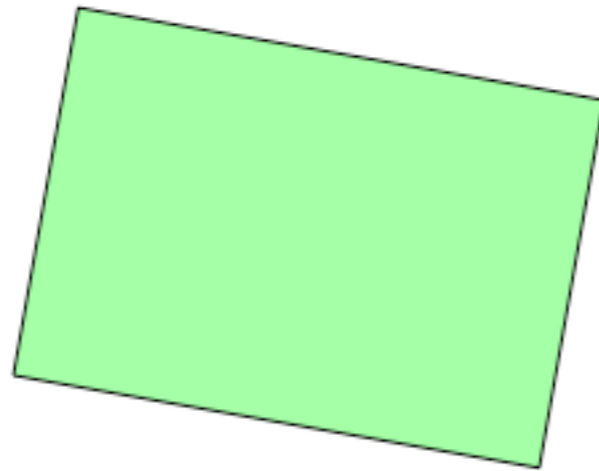
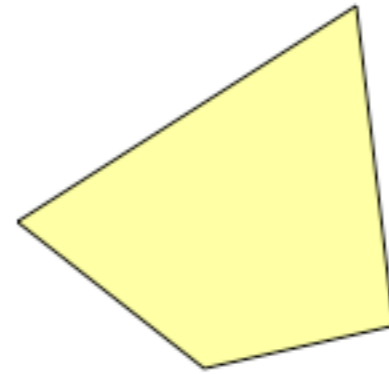
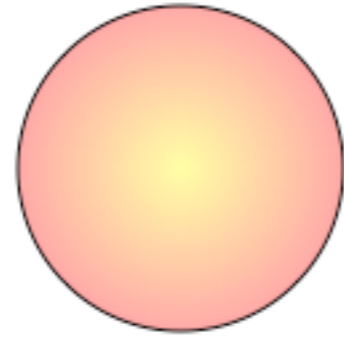
**Grafica vettoriale**

# SVG

- 2001: W3C SVG 1.0  
( <http://www.w3.org/TR/SVG/> )
- Import / Export da applicativi di disegno vettoriale  
( è un pò l'erede di EPS )
- 2010: supporto in IE per la prima volta!
- usato come immagine in HTML
- comodo editor online:  
<http://code.google.com/p/svg-edit/>

# esempio

```
<svg width="640" height="480" xmlns="http://www.w3.org/2000/svg">
  <defs>
    <radialGradient id="gradiente" r="0.5" cx="0.5" cy="0.5">
      <stop stop-color="#FFFF99" offset="0" />
      <stop stop-color="#FF9999" offset="1" />
    </radialGradient>
  </defs>
  <rect id="rettangolo"
    height="140" width="200"
    x="128" y="200"
    transform="rotate(10.0, 0.0, 0.0)"
    stroke-width="1" stroke="#000000" fill="#99FF99" />
  <path id="poligono"
    d="m358,211l127,-81l14,120l-71,161l-70,-55z"
    stroke-width="1" stroke="#000000" fill="#FFFF99" />
  <circle id="cerchio"
    r="61.0"
    cx="142" cy="80"
    stroke-width="1" stroke="#000000" fill="url(#gradiente)" />
</svg>
```



# CSS 3

---

## Le novità

# CSS 3

- 1996 W3C: CSS 1
- 1998 W3C: CSS 2 (corrente: 2.1)
- dal 2005 W3C: CSS 3 under development
- status dettagliato:  
<http://www.w3.org/Style/CSS/current-work>

# Nuovi selettori

- operatori `^=`, `$=` e `*=` per cercare elementi con attributi che hanno un valore che inizia con una data stringa, finisce con una data stringa o contiene un data stringa
- nuovi pseudo selettori come `:nth-of-type()` e `:not()`
- vedi esempio 09

# Attributi in *media queries*

per specificare condizioni aggiuntive

```
@media screen    and (max-width: 1200px) {}
```

```
@media handheld and (min-width: 200px) {}
```

```
@media print    and (color) {}
```

etc...

# Trasparenze

```
div#box {  
    background-color: red;  
    opacity: 0.30;  
}
```

A partire dalla 9 non serve più la proprietà non-standard `filter` per IE (almeno per le trasparenze)!

# Ombre

```
p#ombra {  
    font-size: 20pt;  
    text-shadow: #BBBBBB 2px 2px  
}
```

Ombre sui testi.

# Layout in colonne

```
p#colonne {  
  column-count: 3;  
  -moz-column-count: 3;  
  -webkit-column-count: 3;  
}
```

Finalmente: layout in colonne!

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus ipsum tellus, lacinia non sagittis vitae, consectetur vel est. Pellentesque nibh turpis, consectetur eget bibendum a, aliquam sit amet odio. Vivamus varius tincidunt ante quis ornare. Nam a metus neque, in tincidunt uma. Donec placerat, lacus eget porttitor pharetra, ipsum dui porta sem, id posuere uma est et ligula. Morbi

nisi augue, convallis vitae malesuada id, lacinia et sapien. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Ut sed diam est, ac vestibulum risus. Phasellus malesuada, eros a placerat varius, purus purus feugiat eros, sed tincidunt dolor enim quis leo. Aenean cursus bibendum mattis. Nam fermentum aliquet enim, at tincidunt elit ullamcorper

ut. Pellentesque nisi risus, hendrerit id convallis at, iaculis vel ante. In nec mauris id tortor tincidunt ultrices. In dictum sagittis lacus congue scelerisque. Duis eget faucibus uma. Nunc consectetur fringilla pharetra. Etiam vitae pulvinar sem. Nulla lectus massa, dictum non posuere id, elementum at metus.

# Angoli smussatissimi

```
p#angoli {  
  border-radius: 10px;  
  box-shadow: #AAAAAA 3px 3px 5px;  
  -moz-border-radius: 10px;  
  -moz-box-shadow: #AAAAAA 3px 3px 5px;  
  -webkit-border-radius: 10px;  
  -webkit-box-shadow: #AAAAAA 3px 3px 5px;  
}
```

Angoli smussatissimi

# Traformazioni 2D

```
#c1 {  
  transform: rotate(22.5deg);  
  -moz-transform: rotate(22.5deg);  
  -webkit-transform: rotate(22.5deg);  
}  
#c2 {  
  transform: rotate(-22.5deg) scale(0.75);  
  -moz-transform: rotate(-22.5deg) scale(0.75);  
  -webkit-transform: rotate(-22.5deg) scale(0.75);  
}  
#c3 {  
  transform: rotate(120deg) scale(1.20);  
  -moz-transform: rotate(120deg) scale(1.20);  
  -webkit-transform: rotate(120deg) scale(1.20);  
}
```



# Fonts!

- la regola `@font-face` permette di definire una font-family custom
- viene data l'URL del font da scaricare
- purtroppo non c'è accordo sul formato: tutti supportano TTF e OTF, ma IE supporta EOT → ci sono considerazioni legali
- NB: occhio alle licenze!

# Esempio uso font

```
@font-face {  
    font-family: Mynumberfont;  
    src: url('GFSBaskerville.otf');  
}  
  
p {  
    font-family: Mynumberfont;  
    font-size: 14pt;  
}
```