

HTML5入门



吴君星

Web技术大致的时间轴

1991 HTML

1994 HTML 2

1996 CSS 1 + JavaScript

1997 HTML 4

1998 CSS 2

2000 XHTML 1

2002 使用DIV+CSS进行网页布局

2005 AJAX

2009 HTML 5

HTML5 \sim HTML + CSS + JS APIs

Html5的发展其实是html, css, js api的发展

HTML 5草案的前身名为Web Applications 1.0, 是在2004年由WHATWG提出, 再于2007年获W3C接纳, 并成立了新的HTML工作团队。

在2008年1月22日, 第一份正式草案发布。WHATWG表示该规范是目前仍在进行的工作, 仍须多年的努力。

目前Firefox、Chrome、Opera、Safari (版本4以上)、Internet Explorer 9已支持HTML5技术。

HTML 5的标准草案目前已进入W3C制定标准5大程序的第1步。

负责编纂标准格式文件的Google代表Ian Hickson预期, 可能得等到2012年才会推出建议候选版。

Html 标签

Video, Audio

Video:ogg,mp4,webm audio:ogg,mp3,wav

demo

```
<video width="320" height="240" controls="controls">
```

```
  <source src="movie.ogg" type="video/ogg">
```

```
  <source src="movie.mp4" type="video/mp4">
```

Your browser does not support the video tag.

```
</video>
```

属性

autoplay, controls, loop, preload, src

http://www.w3school.com.cn/tiy/t.asp?f=html5_video_all

http://www.w3school.com.cn/tiy/t.asp?f=html5_audio_all

Canvas

HTML5 的 canvas 元素使用 JavaScript 在网页上绘制图像。

画布是一个矩形区域，您可以控制其每一像素。

canvas 拥有多种绘制路径、矩形、圆形、字符以及添加图像的方法。

```
<canvas id="myCanvas" width="200" height="100">
```

Your browser does not support the canvas element.

```
</canvas>
```

正方形

```
var c=document.getElementById("myCanvas");  
var cxt=c.getContext("2d");  
cxt.fillStyle="#FF0000";  
cxt.fillRect(0,0,150,75);
```

线

```
cxt.moveTo(10,10);  
cxt.lineTo(150,50);  
cxt.lineTo(10,50);  
cxt.stroke();
```

图片

```
var img=new Image()  
img.src="flower.png"  
cxt.drawImage(img,0,0);
```

圆

```
cxt.beginPath();  
cxt.arc(70,18,15,0,Math.PI*2,true);//度,弧度  
cxt.closePath();  
cxt.fill();
```

线性渐变

```
var grd=cxt.createLinearGradient(0,0,175,50);  
grd.addColorStop(0,"#FF0000");  
grd.addColorStop(1,"#00FF00");  
cxt.fillStyle=grd;  
cxt.fillRect(0,0,175,50);
```

http://www.w3school.com.cn/html5/html_5_canvas.asp

<http://www.jb51.net/html5/32168.html>

Input

HTML5 拥有多个新的表单输入类型。这些新特性提供了更好的输入控制和验证。
Email, url, number, range, Date pickers, search, color

Demo

E-mail: `<input type="email" name="user_email" />`

http://www.w3school.com.cn/tiy/t.asp?f=html5_form_email

`<input type="url" name="user_url" />`

`<input type="number" name="points" min="1" max="10" />`

Max,min,step,value

http://www.w3school.com.cn/tiy/t.asp?f=html5_form_number

`<input type="range" name="points" min="1" max="10" />` 滑动条

http://www.w3school.com.cn/tiy/t.asp?f=html5_form_range

`<input type="date" name="user_date" />`

date, month, week, time, datetime, datetime-local

http://www.w3school.com.cn/tiy/t.asp?f=html5_form_datetime-local

Datalist, keygen, output

```
<input type="url" list="url_list" name="link" />
<datalist id="url_list">
<option label="W3School" value="http://www.W3School.com.cn" />
<option label="Google" value="http://www.google.com" />
<option label="Microsoft" value="http://www.microsoft.com" />
</datalist>
```

Opera 全部支持 chrome支持keygen

keygen 元素的作用是提供一种验证用户的可靠方法。

keygen 元素是密钥对生成器（key-pair generator）。当提交表单时，会生成两个键，一个是私钥，一个公钥。

私钥（private key）存储于客户端，公钥（public key）则被发送到服务器。公钥可用于之后验证用户的客户端证书（client certificate）。

浏览器对此元素的支持度不足

对比Structs token 令牌

output 元素用于不同类型的输出，比如计算或脚本输出

http://www.w3school.com.cn/html5/html_5_form_elements.asp

<http://html5.zqcampus.com/article/view14>

1. autocomplete
2. novalidate
3. autocomplete
4. autofocus
5. Form
6. form overrides (formaction, formenctype, formmethod, formnovalidate, formtarget)
 - formaction - 重写表单的 action 属性
 - formenctype - 重写表单的 enctype 属性
 - formmethod - 重写表单的 method 属性
 - formnovalidate - 重写表单的 novalidate 属性
 - formtarget - 重写表单的 target 属性

http://www.w3school.com.cn/tiy/t.asp?f=html5_form_override
7. height 和 width
8. list
9. min, max 和 step
10. multiple
11. pattern (regexp) pattern 属性规定用于验证 input 域的模式 (pattern)
12. Placeholder 水印

<input type="search" name="user_search" placeholder="Search W3School" />
13. Required 必填
14. http://www.w3school.com.cn/tiy/t.asp?f=html5_form_required

HTML 5 还包含了新的元素，比如：<nav>, <header>, <footer> 以及 <figure> 等等。

http://www.w3school.com.cn/html5/html5_reference.asp

HTML 5 标准属性

http://www.w3school.com.cn/html5/html5_ref_standardattributes.asp

HTML 5 事件属性

键盘事件

鼠标事件

媒介事件

表单事件

http://www.w3school.com.cn/html5/html5_ref_eventattributes.asp

CSS3 选择器

奇偶选择器

```
.row:nth-child(even) {  
    background: #dde;  
}  
.row:nth-child(odd) {  
    background: white;  
}
```

属性选择器

```
input[type="text"] {  
    background: #eee;  
}
```

<http://www.css3-html5.com/>

子元素

```
h2:first-child { ... }  
div.text > div { ... }  
h2 + header { ... }
```

反选

```
:not(.box) {  
    color: #00c;  
}  
:not(span) {  
    display: block;  
}
```

E[att^="val"]

E[att\$="val"]

E[att*="val"]

E:root

E:nth-child(n)

E:nth-last-child(n)

E:nth-of-type(n)

E:nth-last-of-type(n)

E:first-of-type

E:only-child

E:only-of-type

E:empty

E:first-child

E:last-child

E:checked

E:enabled

E:disabled

E::selection

E:not(s)

E:target (锚点)

E ~ F

浏览器引擎

Webkit, Safari, Google Chrome, 傲游3, (开源)
Gphone, IPHONE, Nokia's Series 60 browser

Gecko, Firefox

Presto: Opera

Trident: IE,(ie6,7,8不支持css3,只有ie9支持)

Webkit: -webkit-border-radius

Gecko: -moz-border-radius

Presto: -o-border-radius

Trident: border-radius (引用htc文件)

```
.test{  
    border-radius: 20px;  
    behavior: url(ie-css3.htc);"绝对路径"  
}
```

Demo

Htc:<http://www.iefans.net/wp-content/uploads/2010/12/iecss3.rar>

<http://www.chineselinuxuniversity.net/articles/44361.shtml>

Server Font

@font-face{

font-style: 设置文本样式。

font-variant: 设置文本是否大小写。

font-weight: 设置文本的粗细。

font-stretch: 设置文本是否横向的拉伸变形。

font-size: 设置文本字体大小。

src: 设置自定义字体的相对路径或者绝对路径

}

@font-face 能够加载服务器端的字体文件，让客户端显示客户端所没有安装的字体 (eot,otf)

<Style>

@font-face {

font-family: Runic;

src:url(RUNICMT0.eot);

}

@font-face {

font-family: Kingston;

src:url(KINGSTON.eot);

}

.king { FONT-SIZE: 45pt; FONT-FAMILY: "Kingston" }

.border { FONT-SIZE: 35px; COLOR: black; FONT-FAMILY: "BorderWeb" }

</style>

text-overflow

clip: 不显示省略标记 (...), 而是简单的裁切。

ellipsis: 当对象内文本溢出时显示省略标记 (...)

columns: 宽度 || 栏目数

column-width, 可以定义每栏的宽度

column-span, 夸列

column-rule, 定义每栏之间边框的宽度,样式和颜色

column-gap: 定义两栏之间的间距距离

column-fill: 定义栏目的高度是否统一 (auto, balance)

column-count: 栏目数

nav-index 替代 **tabindex**

Nav-up:<id>

Nav-right

Nav-down

Nav-left

resize

box-shadow

border-radius

border-image

background-origin : border | padding | content

background-clip: border-box | padding-box | content-box | no-clip

background-size,100%,80px

background-image:url(),url()

background: -webkit-gradient(radial, 430 50, 0, 430 50, 200, from(red), to(#000))

Transition变换

Transforms,旋转, 拖放, 平移

text-shadow: color,length,opacity

text-stroke-width, 描边

text-stroke-color

Animation

```
@-webkit-keyframes pulse {  
  from {  
    opacity: 0.0;  
    font-size: 100%;  
  }  
  to {  
    opacity: 1.0;  
    font-size: 200%;  
  }  
}
```

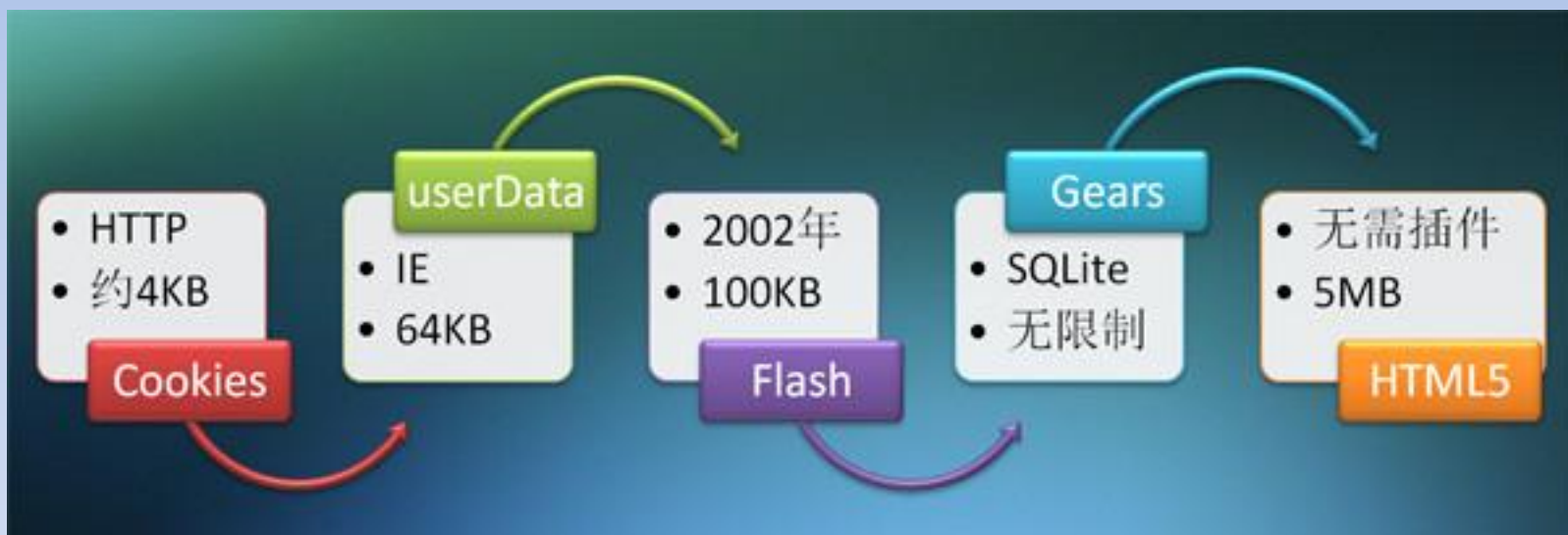
```
div {  
  -webkit-animation-name: pulse;  
  -webkit-animation-duration: 2s;  
  -webkit-animation-iteration-count: infinite;  
  -webkit-animation-timing-function: ease-in-out;  
  -webkit-animation-direction: alternate;  
}
```

<http://directguo.com/html5/#slide44>

<http://blog.moocss.com/tutorials/mootools-tutorials/1591.html>

离线存储

本地存储解决方案很多，比如Flash SharedObject、Google Gears、Cookie、DOM Storage、User Data、window.name、Silverlight、Open Database等。



<http://www.css88.com/archives/3717>

Cookie

```
function SetCookie(sName, sValue){  
    var date = new Date();  
    document.cookie = sName + "=" + escape(sValue) + "; expires=" + date.toGMTString();  
}  
function GetCookie(sName){  
    var aCookie = document.cookie.split("; ");  
    for (var i=0; i < aCookie.length; i++){  
        var aCrumb = aCookie[i].split("=");  
        if (sName == aCrumb[0])  
            return unescape(aCrumb[1]);  
    }  
    return null;  
}  
function DelCookie(sName){  
    document.cookie = sName + "=" + escape(sValue) + "; expires=Fri, 31 Dec 1999 23:59:59 GMT";  
}
```

备注:

现代浏览器一般默认都支持cookie，但是使用cookie也有几个致命的弱点：存储的信息量太少，页面向服务器发送请求的同时cookie也会被发送，无形中浪费用户的带宽。

<http://baike.baidu.com/view/835.htm>

userData

1. userData将数据以XML格式保存在客户端计算机上（windows）
2. C:\Documents and Settings\Liming\UserData\
3. userData行为提供了一个比Cookie更具有动态性和更大容量的数据结构。
4. 每页的UserData 存储区数据大小可以达到64 Kb，每个域名可以达到640 Kb。
5. userData 行为通过sessions为每个对象分配UserData存储区。
6. HTML、HEAD、TITLE和STYLE标记上应用了userData行为后将会出错

成员

expires 设置或取得使用userData行为保存数据的失效日期

getAttribute() 取得指定的属性值；

load(存储区名) 从userData存储区载入存储的对象数据；

removeAttribute() 删除指定的属性值；

save(存储区名) 将对象存储到一个userData存储区；

setAttribute() 设置指定的属性值；

XMLDocument 取得存储该对象数据的XML?DOM引用。

userData Demo

1. `<STYLE>`
 `.userData {behavior:url(#default#userdata);}`
 `</STYLE>`
2. `<ELEMENT STYLE="behavior:url('#default#userData')" ID=sID>`
3. `object.style.behavior = "url('#default#userData')"`
4. `object.addBehavior("#default#userData")`

```
function fnSaveInput(){
    var oPersist=oPersistForm.oPersistInput;
    oPersist.setAttribute("sPersist",oPersist.value);
    oPersist.expires= new Date(1980,0,1,7,59,59).toUTCString();
    oPersist.save("oXMLBranch");
}
function fnLoadInput(){
    var oPersist=oPersistForm.oPersistInput;
    oPersist.load("oXMLBranch");
    oPersist.value=oPersist.getAttribute("sPersist");
}
```

<http://www.css88.com/archives/3717>

<http://blog.csdn.net/willvc123/article/details/1821395>

<http://www.cnblogs.com/icedog/archive/2011/03/16/1985957.html>

<http://www.blogjava.net/emu/articles/39485.html>

sessionStorage localStorage

核心方法

```
sessionStorage.setItem("key","value");  
sessionStorage.key = "value";  
sessionStorage.removeItem("key");  
sessionStorage.getItem("key");  
sessionStorage.key  
sessionStorage.clear();
```

localStorage与sessionStorage方法相同

Demo

```
if (!localStorage.pageLoadCount)  
    localStorage.pageLoadCount = 0;  
localStorage.pageLoadCount = parseInt(localStorage.pageLoadCount) + 1;  
document.getElementById('count').text= localStorage.pageLoadCount;
```

<http://www.jsmix.com/html5/sessionstorage.html>

<http://dev.w3.org/html5/webstorage/#dom-localstorage>

<http://dev.w3.org/html5/webstorage/#dom-sessionstorage>

<https://developer.mozilla.org/en/dom/storage#localStorage>

Web SQL Database

核心方法

1. `openDatabase("", "", "", "", function(t){})`

//创建数据库，数据库名，版本号，描述，大小(b),匿名函数(可省略)

2. `transaction(function(tx){})`

//执行查询，匿名函数，tx，事物类型，负责实际的查询

3. `executeSql("", [], function(tx, result){}, function(tx, error){})`

//真正执行查询，sql语句，参数，成功函数(事物参数，结果集)，失败函数(事物参数，结果集)

Sql语句 demo

`SELECT COUNT(*) FROM tb`

`CREATE TABLE db (id REAL UNIQUE, label TEXT, timestamp REAL)`

`INSERT INTO db (label, timestamp) values(?, ?), ["text", new Date().getTime()]`

`INSERT INTO db (label, timestamp) values ("Test", 1265925077487)`

查询结果

`result.rows.item(i)`

`result.rows.item(i)['label']`

`data.rows[0].label`

`data.rows[0]['label']`

Web SQL Database Demo

```
var db = openDatabase("DBName", "1.0", "A list of to do items.", 5*1024*1024);//5MB
if(db)
{
    db.transaction(
        function (tx){
            tx.executeSql("SELECT COUNT(*) FROM TBName", [], function(tx,result){
                for(var i = 0; i < result.rows.length; i++)
                {
                    document.write('<b>' + result.rows.item(i)['label'] + '</b><br />');
                }
            }, function(tx, error){
                tx.executeSql("CREATE TABLE TBName (id REAL UNIQUE, label TEXT, timestamp REAL)", [],
                    null, null);
                tx.executeSql("INSERT INTO ToDo (label, timestamp) values(?, ?)",
                    ["test", new Date().getTime()], null, null);
            });
        })
}
```

web storage demo

<http://space.itpub.net/10896630/viewspace-680607>

<http://developer.51cto.com/art/201003/185970.htm>

<http://www.w3.org/TR/webdatabase/>

Web Application Cache

用法

1. 需要在页面的html标签中通过manifest属性引用一个manifest文件使得应用可缓存。
2. manifest文件是一个文本文件，它罗列了离线访问应用时所需缓存的文件清单。
3. 引用该manifest文件的页面，不管你有没有罗列到清单中，都会被缓存。

服务器要求

1. 配置manifest的文本文件（ served as [text/cache-manifest](#)， UTF-8 ）。
2. 增加服务器MIME类型信息（ text/cache-manifest ）
3. MIME：多用途 Internet 邮件扩展 (MIME) 类型说明了 Web 浏览器或邮件应用程序如何处理从服务器接收的文件。当 Web 浏览器请求服务器上的某一项目时，也会请求此对象的 MIME 类型。某些 MIME 类型（例如图形）可以在浏览器内部显示。其他的 MIME 类型（例如文字处理文档）则需要使用外部帮助应用程序来显示。

好处

1. 最直接的好处就是用户可以离线访问你的web应用
2. 因为文件被缓存在本地使得web页面加载速度提升许多
3. 离线应用只加载被修改过的资源，大大降低了用户请求对服务器造成的负载压力

Web Application Cache

Manifest语法

CACHE MANIFEST

CACHE

a.Jpg

b.Css

Index.aspx

NETWORK

*

Login.aspx

Register.aspx

FALLBACK

/ /default.aspx

/index.aspx /404.html

#注释符号

Demo

```
<html manifest="test.manifest"> ... </html>
```

Web Application Cache

更新

1. 删除缓存数据
2. 修改manifest文件
3. 用js更新

```
Var cache = window.applicationCache;  
Cache.swapCache();
```

监听事件

onchecking, onerror, onnoupdate, ondownloading, onprogress, onupdateready, oncached和onobsolete。

```
Cache. addEventListener('error', handleCacheError, false);  
function handleCacheError(e) {  
    alert('Error: Cache failed to update!');  
};
```

<http://www.w3.org/TR/html5/offline.html>

<http://www.cnblogs.com/heartstill/archive/2010/12/30/1921430.html>

Web Workers

Web Workers为WEB前端网页上的脚本提供了一种能在后台进程中运行的方法。一旦它被创建，Web Workers就可以通过postMessage()向任务池发送任务请求，执行完之后再通过postMessage()返回消息给创建者指定的事件处理程序(通过onmessage进行捕获)。

```
var worker = new Worker("worker.js");//并发线程，新线程  
worker.onmessage = function(message){ var a = message.Data; };//主线程  
worker.postMessage("aaa");//主线程异步函数，run thread
```

Worker.js

```
var onmessage = function(message){  
    var m = message.data;  
    //处理数据  
    postMessage(m);//参数可是obj，Json: [{name:"小明"},message.data.name]  
}
```

<http://blog.csdn.net/xiaojianpitt/article/details/4279611>

<http://dev.w3.org/html5/workers/>

<http://www.w3.org/TR/workers/>

<http://www.html5rocks.com/en/tutorials/workers/basics/>

<http://www.ibm.com/developerworks/cn/xml/x-html5mobile4/>

异步和多线程

相同点：防止阻塞

不同点：异步是目的，有多中实现方式，多线程只是实现异步的一个手段。

举例（A叫B，C两人吃饭）

异步：A叫B吃饭，不用等B回答，就可以叫C去吃饭。如果B应答，A收应答。

多线程：A和D两人分别叫B,C两人吃饭，一人叫一个，互不影响

<http://hi.baidu.com/wzx2pp/blog/item/2d4be7ec00cd432063d09f1e.html>

<http://topic.csdn.net/u/20090113/19/76384901-f446-4e4c-a58a-4980be6b42d8.html>

WebSocket

背景 <http://zh.wikipedia.org/wiki/WebSocket>

1. WebSocket 是下一代客户端-服务器的异步通信方法.
2. WebSocket 最伟大之处在于服务器和客户端可以在任意时刻相互推送信息
3. Ajax技术需要客户端发起请求，WebSocket服务器和客户端可以彼此相互推送信息
4. XHR受到域的限制，而WebSocket允许跨域通信

Client

```
var socket = new WebSocket('ws://localhost:8080');  
socket.onopen = function(event) { }  
socket.send('I am the client and I\'m listening!');  
socket.onmessage = function(event) { };  
socket.onclose = function(event) { };  
socket.close();
```

Server

<http://site.douban.com/widget/forum/33650/discussion/37917149/>

代替技术：Flash技术，AJAX Long-Polling技术

<http://www.cnblogs.com/wei2yi/archive/2011/03/23/1992830.html>

http://blog.sina.com.cn/s/blog_575b0ca50100q17k.html

Drag and Drop

HTML5为元素新增了用于拖拽的属性**draggable**，这个属性决定了元素是否能被拖拽，如果**draggable="true"**，则元素可被拖拽，否则只能选择元素的文本。

值得一提的是HTML5支持拖拽数据存储，使用**dataTransfer**对象。

dragstart: 要被拖拽的元素开始拖拽时触发，这个事件对象是被拖拽元素

dragenter: 拖拽元素进入目标元素时触发，这个事件对象是目标元素

dragover: 拖拽某元素在目标元素上移动时触发，这个事件对象是目标元素

dragleave: 拖拽某元素离开目标元素时触发，这个事件对象是目标元素

dragend: 在**drop**之后触发，就是拖拽完毕时触发，这个事件对象是被拖拽元素

drop: 将被拖拽元素放在目标元素内时触发，这个事件对象是目标元素

dragstart → dragenter → dragover → drop → dragend

Demo

<http://www.zhangxinxu.com/wordpress/2011/02/html5-drag-drop-%E6%8B%96%E6%8B%BD%E4%B8%8E%E6%8B%96%E6%94%BE%E7%AE%80%E4%BB%8B/>

Geolocation地理定位

```
if (navigator.geolocation) {  
    navigator.geolocation.getCurrentPosition(function(position) {  
        var lat = position.coords.latitude;  
        var lng = position.coords.longitude;  
        map.setCenter(new GLatLng(lat, lng), 13);  
        map.addOverlay(new GMarker(new GLatLng(lat, lng)));  
    });  
}
```

Demo

<http://directguo.com/html5/#slide14>

说明

<http://www.jsmix.com/html5/try-out-geolocation.html>

<http://www.scriptlover.com/post/723>

html5 ppt

<http://directguo.com/html5/>

html5 demo

<http://html5demos.com/>

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

<http://mrdoob.com/>

html5 介绍

<http://diveintohtml5.org/introduction.html>

<http://www.mhtml5.com/>

html5 教程

<http://www.w3school.com.cn/html5/index.asp>

<http://www.w3schools.com/html5/default.asp>

<http://developer.51cto.com/art/200907/133407.htm>

http://www.ibm.com/developerworks/cn/web/lp/html5/?cmp=dwnpr&cpb=dw&ct=dwcon&cr=cn_51CTO_run&ccy=cn

<http://developer.51cto.com/art/200907/133407.htm>

html5浏览器测试

<http://www.html5test.com/>

<http://www.findmebyip.com/litmus#html5-web-applications>

编程

<http://www.html51.com/topic-html5-ide.html>

资源

<http://www.css-html5.com/>

<http://html5cn.net/>

<http://kb.cnblogs.com/zt/html5/>

<http://www.mhtml5.com/events/html5-in-china>

Work

1. [Node.js](#)
2. [V8](#)
3. [HTML5 File API](#)
4. [HTML5 互动游戏](#)
5. ○ ○ ○ ○

THANK YOU