

# **HTML5 in Your Pocket:**

## *Application Cache and Local Storage*

**Scott Davis**

ThirstyHead.com





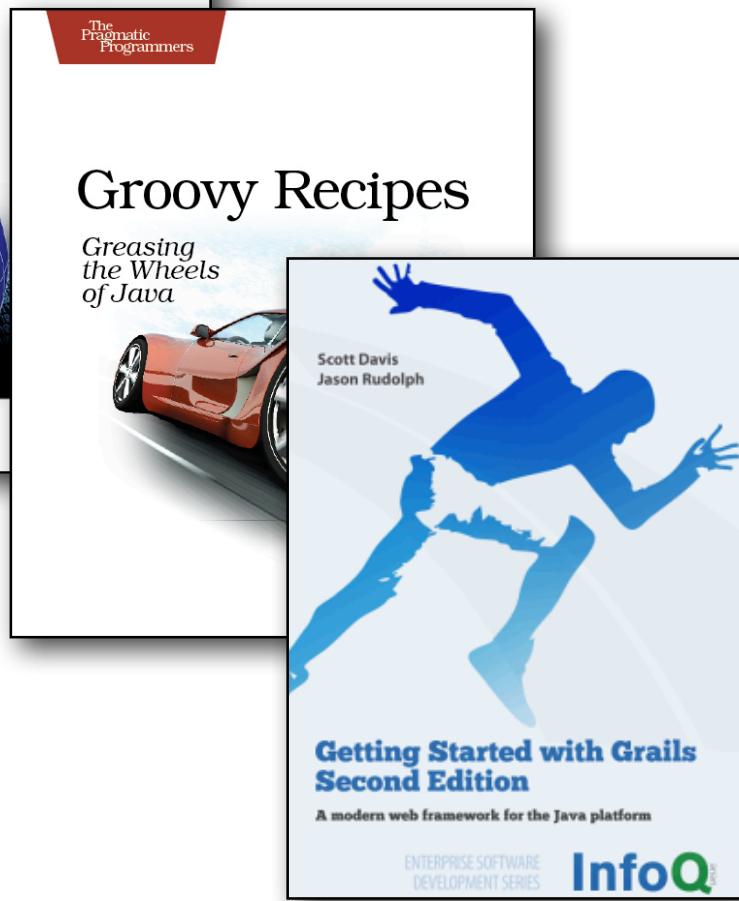
ThirstyHead.com

**training done right.**



# ThirstyHead.com

training done right.

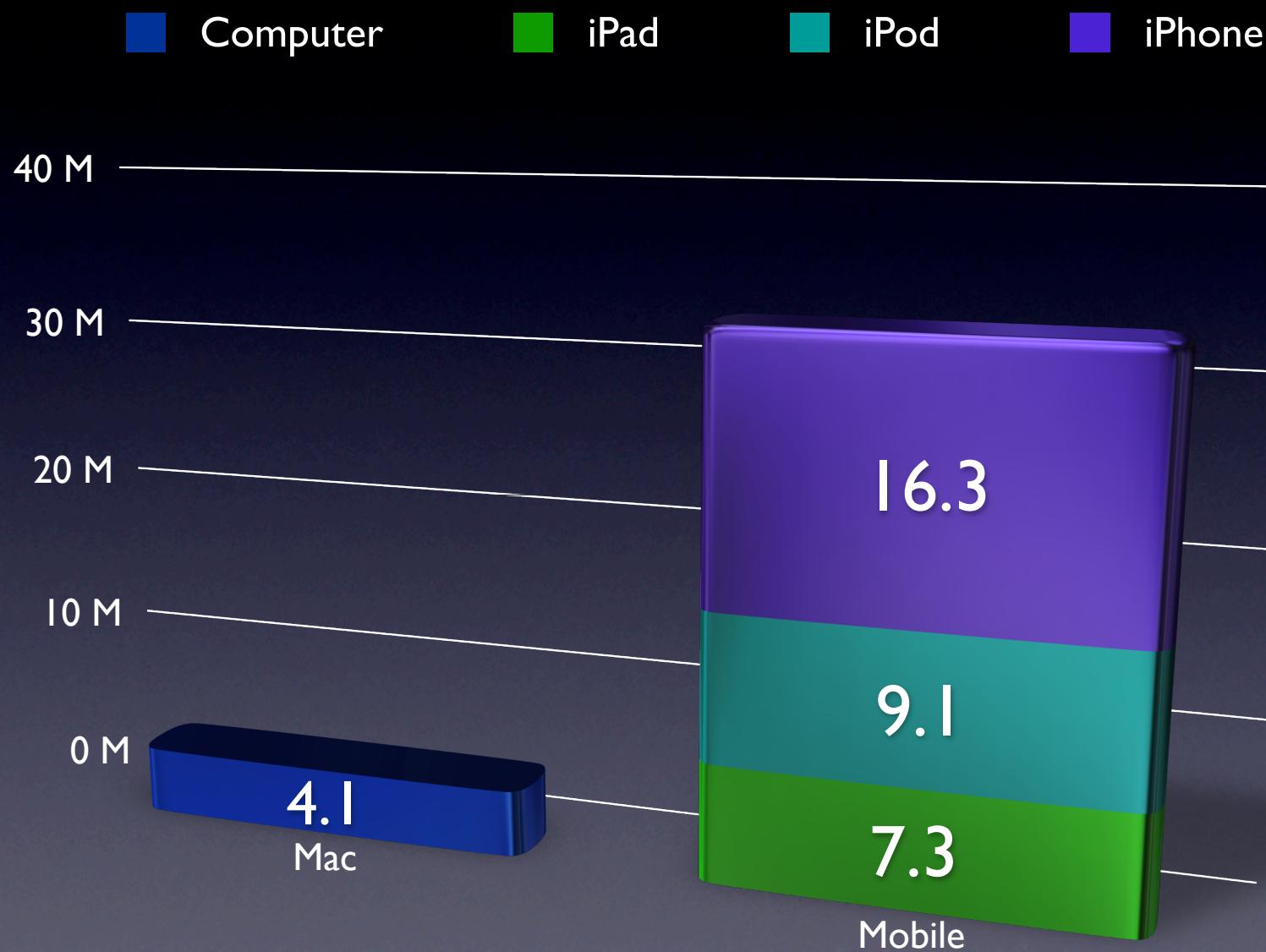


# HTML





# Apple Sales, Q4 2010



# Smartphones Outsell PCs

By Sarah Perez / February 8, 2011 7:17 AM / [8 Comments](#)

[Tweet](#)

[Hacker News](#)



According to IDC, smartphone manufacturers [shipped](#) 100.9 million devices in the fourth quarter of 2010, while PC manufacturers [shipped](#) 92.1 million units worldwide. Or, more simply put, smartphones just outsold PCs for the first time ever.

The number of smartphones sold in Q4 2010 was up 87.2% from the 53.9 million sold in Q4 2009. For the year, vendors shipped 302.6 million smartphones - an increase of 74.4% from the 173.5 million in 2009.

PC sales were up in Q4, too, but just barely. From Q4 2009 to Q4 2010, the increase was only 5.5%. When looking at the yearly totals, however, PCs were still king. Manufacturers shipped 346.2 million units during 2010, compared with the 302.6 million mentioned above from smartphone makers.

[Home](#) [iPad News](#) [App Reviews](#) [Gear Reviews](#) [App Series](#) [Apps Tra](#)

## iPad Still the Market Leader But Kindle Fire is Hot

By [Jillian](#) | December 3, 2011 | [No comment yet](#)[Share](#)

Preliminary Estimated Global Ranking of the Top 5 Media Tablet Brands in the Fourth Quarter of 2011 (Ranking by Unit Shipments in Thousands)

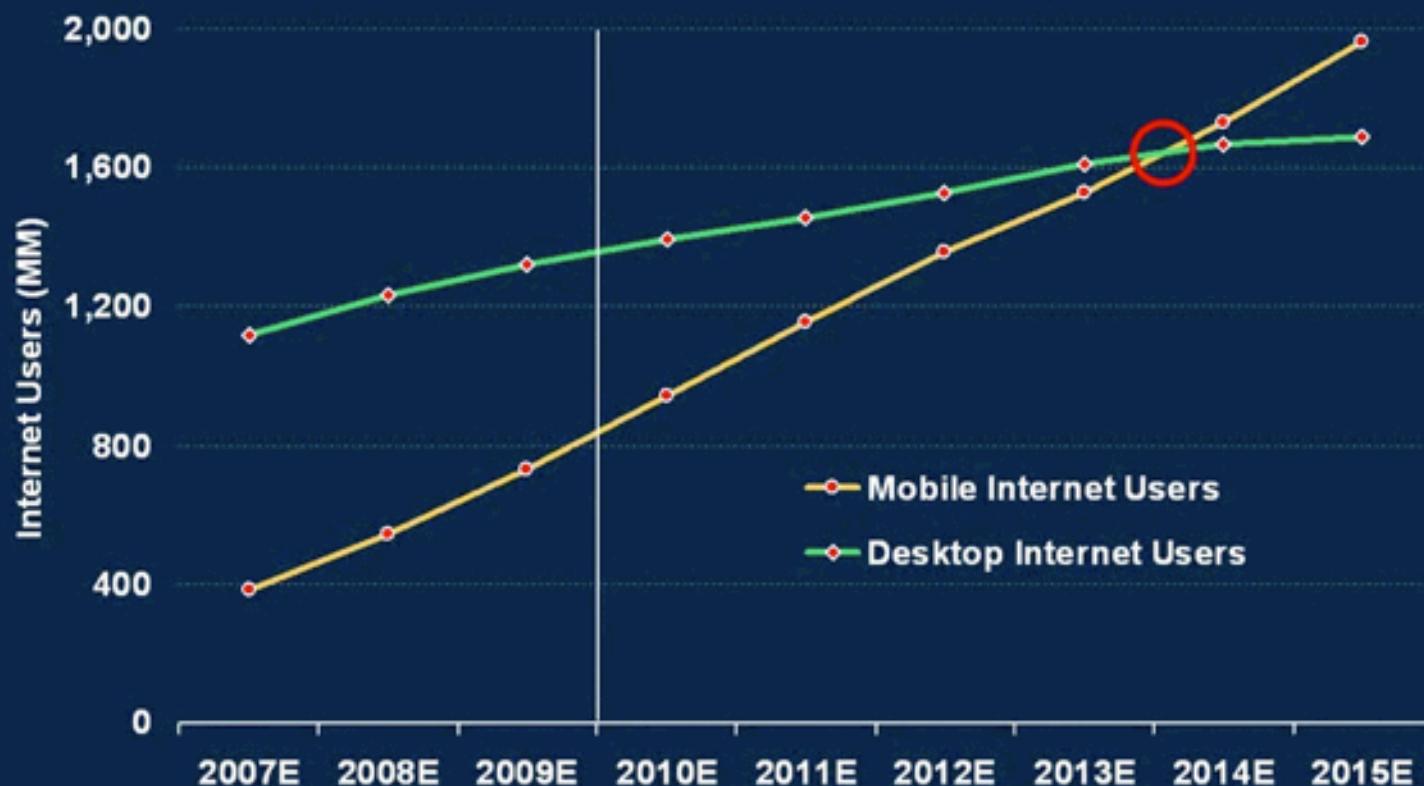
Q4 2011 Rank	Brand	Estimated Q4 2011 Shipments	Estimated Q4 2011 Market Share	Q3 2011 Shipments	Q3 2011 Market Share	Sequential Growth
1	Apple	18,598	65.6%	11,123	69.7%	67.2%
2	Amazon	3,900	13.8%	0	0.0%	NA
3	Samsung	1,370	4.8%	1,250	7.8%	9.6%
4	Barnes & Noble	1,320	4.7%	750	4.7%	76.0%
5	HTC	355	1.3%	253	1.6%	40.3%
	Others	2,800	4.6%	2,582	4.3%	90.0%
	Total	28,343	100.0%	15,958	100.0%	77.6%

Source: [IHS iSuppli](#) December 2011

If you ask iSuppli (and many watching the market do just that), the Kindle Fire is really starting to make some waves across the sea of tablets available to consumers these days.

## Mobile Users > Desktop Internet Users Within 5 Years

Global Mobile vs. Desktop Internet User Projection, 2007 – 2015E



Morgan Stanley

Source: Morgan Stanley Research.





Native  
or  
HTML 5 ?

## Welcome to the Amazon Appstore Developer Program



## Android Market

ANDROID APPS ▾

BOOKS

### iPhone

Features

Design

iOS 4

Apps for iPhone

## Over 350,000 ways to make iPhone even better

The apps that come with your iPhone are just the beginning. Browse the App Store to find hundreds of thousands more, all designed specifically for iPhone. Which means there's almost no limit to what your iPhone can do.



### The world's largest collection of mobile apps.

The App Store is the ultimate source for mobile apps — 350,000 and counting in practically every category. Many are even free.



### Download apps with a tap.

Getting apps onto your iPhone couldn't be simpler. Just find the ones you want, then tap to download them.



The  
Pragmatic  
Programmers

Android 2

# Hello, Android

Introducing Google's Mobile  
Development Platform

Third Edition

*Ed Burnette*



Edited by Susannah Davidson Pfalzer

The  
Pragmatic  
Programmers

# iPad Programming

A Quick-Start  
Guide for  
iPhone  
Developers



Daniel H Steinberg  
Eric T Freeman



## ANDROID & iPhone UPDATE HISTORY

Includes every iPhone & Android model released in the US before July 2010.

Data as of the end of October 2011.

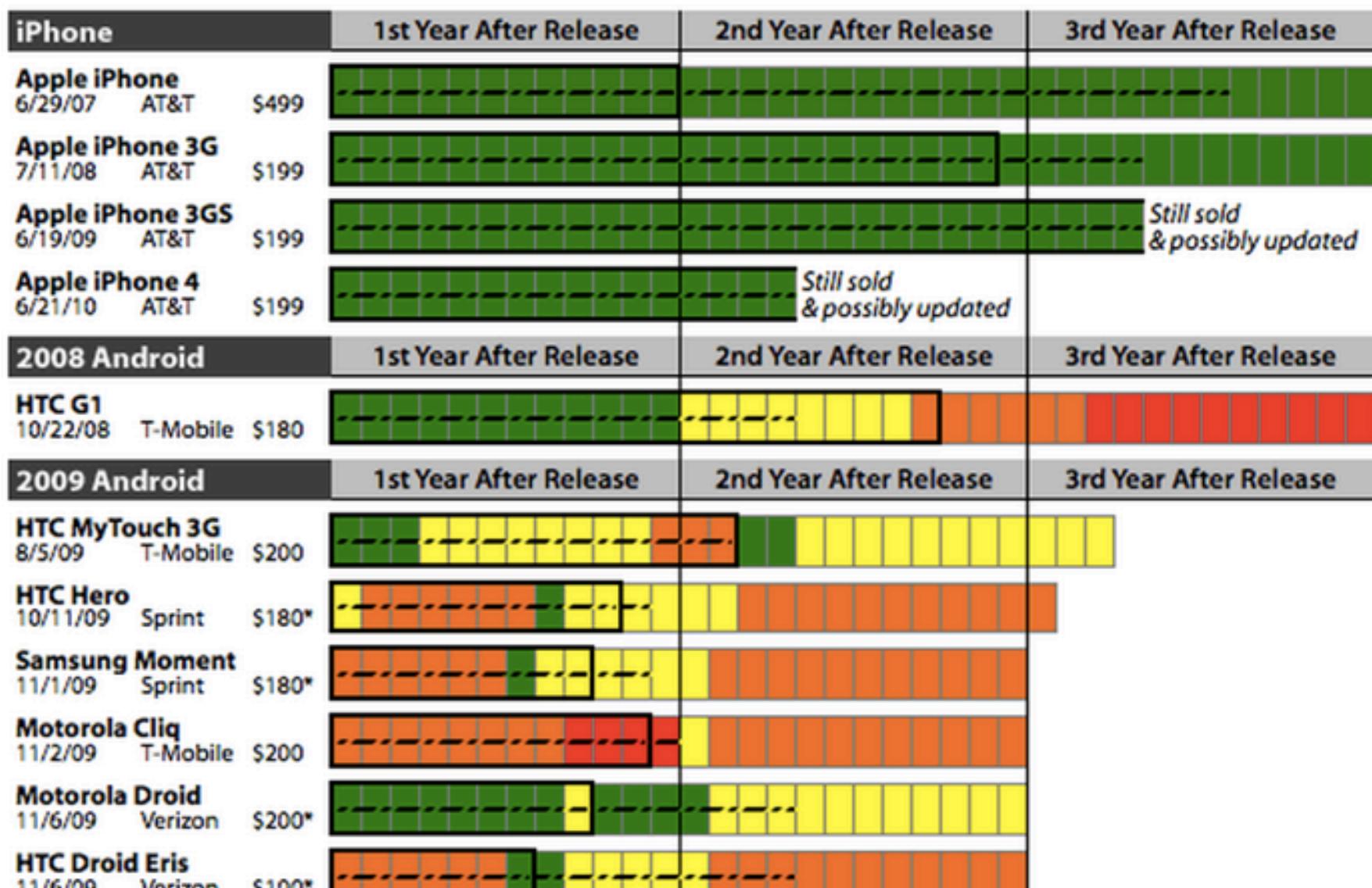
### KEY

	On current major version
	1 major version behind
	2 major versions behind
	3+ major versions behind

Actively for sale

Getting support updates

Prices are with 2 year contract



Launch: Basecamp Mobile - (37sig...)

+

# Launch: Basecamp Mobile Jason F. Feb 01

[129 comments](#) Latest by Tony Mobily

Today we launch Basecamp Mobile for phones and devices with WebKit browsers. This includes the iPhone 3GS, iPhone 4, iPad, Motorola Droid X, Motorola Droid 2, Samsung Galaxy S, HTC Incredible, HTC Evo, Palm Pre 2, BlackBerry Torch, or any other device running iOS 4+, Android 2.1+, webOS 2, or BlackBerry 6.



5





**Local Storage  
(data)**

**Application Cache  
(resources)**

# HTML5: Up and Running

**Dive into the Future**

By [Mark Pilgrim](#)

Publisher: O'Reilly Media

Released: August 2010

Pages: 224



**HTML5**

*Up and Running*

O'REILLY®

*Mark Pilgrim*

<http://diveintohtml5.info/>

# DIVE INTO HTML<sub>5</sub>

BY

MARK PILGRIM

WITH ILLUSTRATIONS FROM THE PUBLIC DOMAIN

A close-up photograph of a pair of blue denim jeans. The jeans feature a dark brown belt loop and a matching button. The waistband is light blue denim with the iconic red Levi's script logo and the number "506" printed on it. The denim has a classic textured pattern and some subtle wear.

# Local Storage

# Cookies



Persistent between requests

Sent back to server  
with every HTTP  
request

Limited to 4k



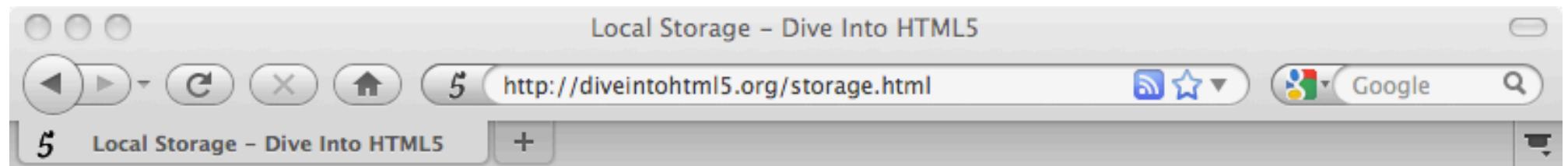


## Nº 7.

# THE PAST, PRESENT & FUTURE OF LOCAL STORAGE FOR WEB APPLICATIONS

Which browsers? Well, the latest version of pretty much every browser supports HTML5 Storage... even Internet Explorer!

HTML5 STORAGE SUPPORT						
IE	FIREFOX	SAFARI	CHROME	OPERA	IPHONE	ANDROID
8.0+	3.5+	4.0+	4.0+	10.5+	2.0+	2.0+



# USING HTML5 STORAGE

HTML5 Storage is based on named key/value pairs. You store data based on a named key, then you can retrieve that data with the same key. The named key is a string. The data can be any type supported by JavaScript,



5 Local Storage – Dive Into HTML5

+

How does it work? Every time a change occurs within the game, we call this function:

```
function saveGameState() {  
    if (!supportsLocalStorage()) { return false; }  
    localStorage["halma.game.in.progress"] = gGameInProgress;  
    for (var i = 0; i < kNumPieces; i++) {  
        localStorage["halma.piece." + i + ".row"] = gPieces[i].row;  
        localStorage["halma.piece." + i + ".column"] = gPieces[i].column;  
    }  
    localStorage["halma.selectedpiece"] = gSelectedPieceIndex;  
    localStorage["halma.selectedpiecehasmoved"] = gSelectedPieceHasMoved;  
    localStorage["halma.movecount"] = gMoveCount;  
    return true;  
}
```

As you can see, it uses the `localStorage` object to save whether there is a game in progress (`gGameInProgress`, a Boolean). If so, it iterates through the pieces (`gPieces`, a JavaScript Array) and saves the row and column number

Done

YSlow

The answers in order of importance are:  
“5 megabytes,”  
“QUOTA\_EXCEEDED\_ERR,”  
and “no.”

The answers in order of importance are:  
“5 megabytes,”  
“QUOTA\_EXCEEDED\_ERR,”  
and “no.”

“5 megabytes” is how much storage space each origin gets by default. This is surprisingly consistent across browsers

**SOURCE:** <http://diveintohtml5.info/storage.html>

The answers in order of importance are:  
“5 megabytes,”  
“QUOTA\_EXCEEDED\_ERR,”  
and “no.”

“QUOTA\_EXCEEDED\_ERR” is the exception  
that will get thrown if you exceed your storage  
quota of 5 megabytes.

**SOURCE:** <http://diveintohtml5.info/storage.html>

The answers in order of importance are:  
“5 megabytes,”  
“QUOTA\_EXCEEDED\_ERR,”  
and “no.”

“No” is the answer to the next obvious question,  
“Can I ask the user for more storage space?”

At time of writing, no browser supports any mechanism for web developers to request more storage space.

**SOURCE:** <http://diveintohtml5.info/storage.html>

# Demo



But what about my IE 6 users?



[PCWorld](#) » [Software](#) » [Browsers & Add-Ons](#)

Recommend:



58



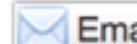
5



0



63



17 Comments Print

# Google to End Support for IE6

By [Nancy Gohring, IDG News](#) Jan 29, 2010 6:50 pm

Google will phase out support for Microsoft's Internet Explorer 6 Web browser starting in March, the company said Friday.

"Many other companies have already stopped supporting older browsers like Internet Explorer 6.0 as well as browsers that are not supported by their own manufacturers. We're also going to begin phasing out our support, starting with Google Docs and Google Sites," Rajen Sheth, Google Apps senior product manager, wrote in a [blog post](#) Friday.

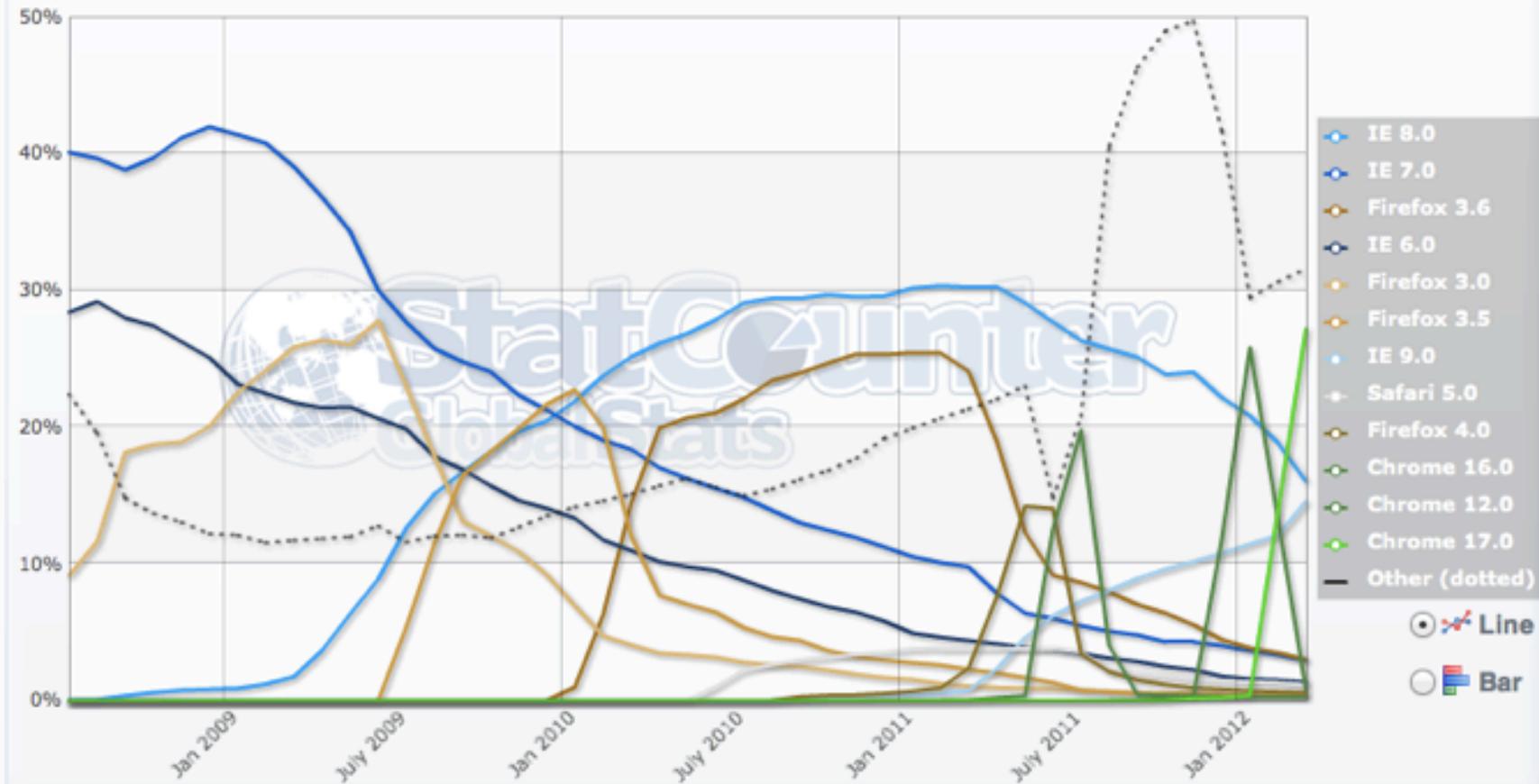
The announcement comes more than two weeks after Google reported that its servers had been the target of attacks originating in China. Those attacks targeted a vulnerability in IE 6, which Microsoft has since issued a fix.

Support for IE6 in Google Docs and Google Sites will end March 1, Sheth said in the post. At

[Main](#)[Press](#)[FAQ](#)[About](#)[Feedback](#)

### StatCounter Global Stats

Top 12 Browser Versions from July 2008 to March 2012

Statistic: Country/Region: Time Period: 

Line  
Bar

# remy sharp's b:log

About [code] and all that jazz

"Dear iOS developers: don't make your app 59p then make an update that switches the app to free and serve ads."

[Blog](#)[jQuery](#)[Projects](#)[Twitter Apps](#)[Talks](#)[About](#)[!\[\]\(f47579369abb76577b982a41567f829d\_img.jpg\) Work](#)[!\[\]\(73e75d230739d664d98b12543b385ee5\_img.jpg\) Subscr...](#)

« [WebSockets in PhoneGap Projects](#)

[CSS selector for :parent targeting \(please\)](#) »

## What is a Polyfill?

A polyfill, or polyfiller, is a piece of code (or plugin) that provides the technology that you, the developer, expect the browser to provide *natively*. Flattening the API landscape if you will.

[Alex Sexton](#) also [classifies polyfilling as a form of Regressive Enhancement](#). I think that sums it up nicely.

[Paul](#) also [defines it as](#):

A shim that mimics a future API providing fallback functionality to older browsers.

# Some Examples

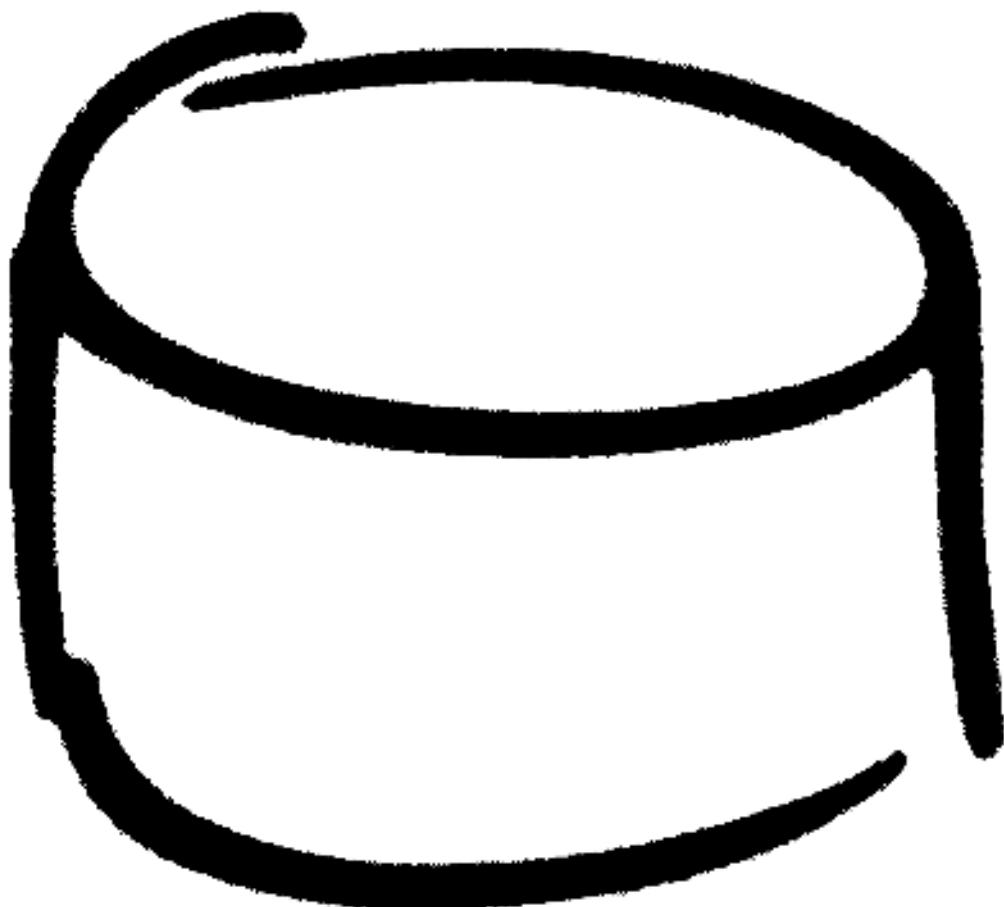
Here's an example: `sessionStorage` is available in all the latest browsers (IE8 and upwards) but isn't in IE7 and below.

A polyfill can be used to [plug the support](#) for older browsers that don't provide `sessionStorage`.

Now with the polyfiller in place, as a developer I can rely on using the Web Storage API (for sessions) and not have to feature test in my code or fork to handle different situations.

# gist.github

```
gist: 350433 download fork star  
Description: Storage polyfill  
Public Clone URL: git://gist.github.com/350433.git  
Embed All Files: show embed  
  
function setData(data) {  
  data = JSON.stringify(data);  
  if (type == 'session') {  
    window.name = data;  
  } else {  
    createCookie('localStorage', data, 365);  
  }  
}  
  
function clearData() {  
  if (type == 'session') {  
    window.name = '';  
  } else {  
    createCookie('localStorage', '', 365);  
  }  
}  
  
function getData() {  
  var data = type == 'session' ? window.name : readCookie('localStorage');  
  return data ? JSON.parse(data) : {};  
}
```



5 Local Storage – Dive Into HTML5

+

The Web SQL Database specification has been implemented by four browsers and platforms.

### WEB SQL DATABASE SUPPORT

IE	FIREFOX	SAFARI	CHROME	OPERA	IPHONE	ANDROID
.	.	4.0+	4.0+	10.5+	3.0+	2.0+

Of course, if you've used more than one database product in your life, you are aware that "SQL" is more of a marketing term than a hard-and-fast standard. (Some would say the same of "HTML5," but never mind that.) Sure, there is an actual SQL specification (it's called [SQL-92](#)), but there is no database server in the world that conforms to that and only that specification. There's Oracle's SQL, Microsoft's SQL, MySQL's SQL, PostgreSQL's SQL, and SQLite's SQL. Indeed, each of these products adds new SQL features over time, so even saying "SQLite's SQL" is not sufficient to pin down exactly what you're talking about. You need to say "the version of SQL that shipped with SQLite version X.Y.Z."

All of which brings us to the following disclaimer, currently residing at the top of the Web SQL Database specification:

This specification has reached an impasse: all interested implementors have used the same SQL backend (Sqlite), but we need multiple independent implementations to proceed along a standardisation path. Until another implementor is interested in implementing this spec, the description of the SQL dialect has been left as simply a reference to Sqlite, which isn't acceptable for a standard.

## Example 2 – Storing Kids in the Database

This example stores several kids into the appropriate table or object store.

This example demonstrates one of the risks that have to be dealt with when using WebDatabase: SQL injection attacks. In WebDatabase explicit transactions must be used, but in IndexedDB a transaction is provided automatically if only one object store is accessed. Transaction locking is per-object store in IndexedDB. Additionally, IndexedDB takes a JavaScript object to insert, whereas with WebDatabase callers must bind specific columns. In both cases you get the insertion id in the callback.

### WebDatabase

```
var kids = [
  { name: "Anna" },
  { name: "Betty" },
  { name: "Christine" }
];

var db = window.openDatabase("CandyDB", "1",
                            "My candy store database",
                            1024);
db.transaction(function(tx) {
  for (var index = 0; index < kids.length; index++) {
    var kid = kids[index];
    tx.executeSql("INSERT INTO kids (name) VALUES (:name);", [kid],
                  function(tx, results) {
                    document.getElementById("display").textContent =
                      "Saved record for " + kid.name +
                      " with id " + results.insertId;
                });
  }
});
```

## Example 3 – List All Kids

This example lists all of the kids stored in the `kids` table or the `kids` object store. WebDatabase uses a result set object which will be passed to the callback method provided after all rows have been retrieved. IndexedDB, on the other hand, passes a cursor to the event handler as results are retrieved. Results should come back faster, as a result. While not shown in this example, you can also stop iterating data with IndexedDB by simply not calling `cursor.continue()`.

### WebDatabase

```
var db = window.openDatabase("CandyDB", "1",
                            "My candy store database",
                            1024);
db.readTransaction(function(tx) {
  // Enumerate the entire table.
  tx.executeSql("SELECT * FROM kids", function(tx, results) {
    var rows = results.rows;
    for (var index = 0; index < rows.length; index++) {
      var item = rows.item(index);
      var element = document.createElement("div");
      element.textContent = item.name;
      document.getElementById("kidList").appendChild(element);
    }
  });
});
```

### IndexedDB



[Home](#) » [Articles](#) »

[« Older Article](#)

## Beyond HTML5: Database APIs and the Road to IndexedDB

on June 1, 2010 by [Arun Ranganathan](#)

in

[Firefox 4](#)

[IndexedDB](#)

[Standards](#)

 [58 comments](#)

While versions of Safari, Chrome, and Opera support a technology called **Web SQL Database**, which uses **SQL** statements as string arguments passed to a JavaScript API, we think *developer aesthetics* are an important consideration, and that this is a particularly inelegant solution for client-side web applications.

# Firefox 4: An early walk-through of IndexedDB

on June 1, 2010 by Arun Ranganathan and Shawn Wilsher

in Firefox 4, IndexedDB, Standards

169 comments

Web developers already have [localStorage](#), which is used for client side storage of simple key-value pairs. This alone doesn't address the needs of many web applications for structured storage and indexed data. Mozilla is working on a structured storage API with indexing support called [IndexedDB](#), and we will have some test builds in the next few weeks. This can be compared to the [WebDatabase API](#) implemented by several browsers that uses a subset of the allowable language of [SQLite](#). Mozilla has chosen to not implement WebDatabase for [various reasons discussed in this post](#).

In order to compare [IndexedDB](#) and [WebDatabase](#), we are going to show four examples that use most parts of the asynchronous APIs of each specification. The differences between SQL storage with tables ([WebDatabase](#)) and JavaScript object storage with indexes ([IndexedDB](#)) becomes pretty clear after reading the examples. The synchronous versions of these APIs are only available on worker threads. Since not all browsers currently implement worker threads, the synchronous APIs will not be discussed at this time. The [IndexedDB](#) code is based off a [proposal that Mozilla has submitted to the W3C WebApps working group](#) that has gotten positive feedback so far. The code for both APIs does not include any error handling (for brevity), but

## ABOUT THE AUTHORS

[Arun Ranganathan](#)

Read more articles by Arun Ranganathan...



[Shawn Wilsher](#)

Read more articles by Shawn Wilsher...



## ARTICLES BY CATEGORY

<a href="#">35 Days</a> (45)	<a href="#">HTML5</a> (102)
<a href="#">@font-face</a> (9)	<a href="#">Identity</a> (7)
<a href="#">about:hacks</a> (2)	<a href="#">Images</a> (4)
<a href="#">Add-ons</a> (10)	<a href="#">IndexedDB</a> (9)
<a href="#">Animations</a> (3)	<a href="#">JägerMonkey</a> (5)
<a href="#">Apps</a> (8)	<a href="#">JavaScript</a> (69)

## Example 2 – Storing Kids in the Database

This example stores several kids into the appropriate table or object store.

This example demonstrates one of the risks that have to be dealt with when using WebDatabase: SQL injection attacks. In WebDatabase explicit transactions must be used, but in IndexedDB a transaction is provided automatically if only one object store is accessed. Transaction locking is per-object store in IndexedDB. Additionally, IndexedDB takes a JavaScript object to insert, whereas with WebDatabase callers must bind specific columns. In both cases you get the insertion id in the callback.

---

### IndexedDB

```
var kids = [
  { name: "Anna" },
  { name: "Betty" },
  { name: "Christine" }
];

var request = window.indexedDB.open("CandyDB",
                                    "My candy store database");
request.onsuccess = function(event) {
  var objectStore = event.result.objectStore("kids");
  for (var index = 0; index < kids.length; index++) {
    var kid = kids[index];
    objectStore.add(kid).onsuccess = function(event) {
      document.getElementById("display").textContent =
        "Saved record for " + kid.name + " with id " + event.result;
    };
  }
};
```

## Example 3 – List All Kids

This example lists all of the kids stored in the `kids` table or the `kids` object store. WebDatabase uses a result set object which will be passed to the callback method provided after all rows have been retrieved. IndexedDB, on the other hand, passes a cursor to the event handler as results are retrieved. Results should come back faster, as a result. While not shown in this example, you can also stop iterating data with IndexedDB by simply not calling `cursor.continue()`.

### IndexedDB

```
var request = window.indexedDB.open("CandyDB",
                                    "My candy store database");
request.onsuccess = function(event) {
    // Enumerate the entire object store.
    request = event.result.objectStore("kids").openCursor();
    request.onsuccess = function(event) {
        var cursor = event.result;
        // If cursor is null then we've completed the enumeration.
        if (!cursor) {
            return;
        }
        var element = document.createElement("div");
        element.textContent = cursor.value.name;
        document.getElementById("kidList").appendChild(element);
        cursor.continue();
    };
};
```

## Example 4 – List Kids Who Bought Candy

TOP

## Show all tables

### # IndexedDB - Working Draft

**\*Usage stats:**

Support:

**Global**

46.21%

*Method of storing data client-side, allows indexed database queries. Previously known as WebSimpleDB API.*

Resources: [Mozilla Hacks article](#)

Show all versions	IE	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Opera Mobile	Android Browser
								10.0	2.1
	6.0	3.6				3.2		11.0	2.2
	7.0	9.0 <small>moz</small>				4.0-4.1		11.1	2.3
	8.0	10.0 <small>moz</small>	17.0 <small>webkit</small>	5.0		4.2-4.3		11.5	3.0
Current	9.0	11.0 <small>moz</small>	18.0 <small>webkit</small>	5.1	11.6	5.0	5.0-6.0	12.0	4.0
Near future	10.0 <small>ms</small>	12.0 <small>moz</small>	19.0 <small>webkit</small>	5.2	12.0				
Farther future		13.0 <small>moz</small>	20.0 <small>webkit</small>						

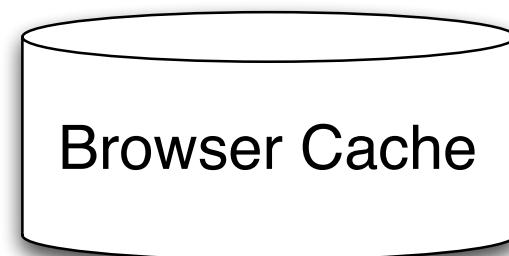
**Note:** Microsoft is currently [experimenting](#) with the technology.

**Feedback**



# Application Cache





GET /index.html



# High Performance Web Sites

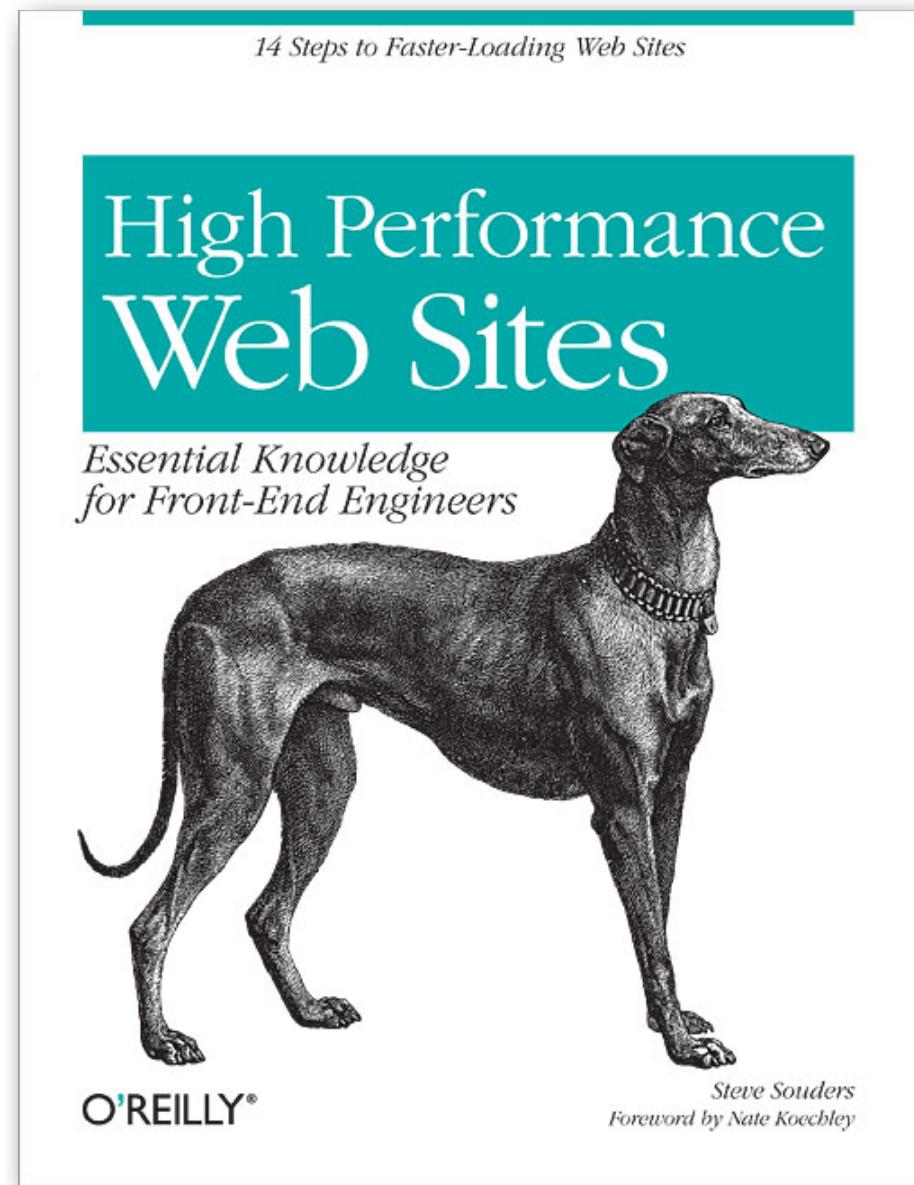
## Essential Knowledge for Front-End Engineers

By [Steve Souders](#)

Publisher: O'Reilly Media

Released: September 2007

Pages: 176



## Components View

YSlow lists all the components in the page including their type, URL, Expires date, gzip status, load time, size, and ETag. You can also view the HTTP response headers for any component.

The screenshot shows the YSlow extension's interface within a browser. At the top, there's a toolbar with icons for Inspect, Performance, Stats, Components (which is selected), Tools, and Help. Below the toolbar is a navigation bar with tabs: Console, HTML, CSS, Script, DOM, Net, YSlow (selected), and Options. A search bar and some control buttons are also at the top.

The main area displays a table of network components:

Type	URL	Expires	Gzip	RespTime	Size (Ungz)
doc	<a href="http://www.yahoo.com/">http://www.yahoo.com/</a>		gzip	282	(10K)
css	<a href="http://us.js2.yimg.com/.../onload_1.3.4.css">http://us.js2.yimg.com/.../onload_1.3.4.css</a>	3/8/2017	gzip	47	1.8K (1.8K)

Below the table, a "Headers" tab is selected, showing the response headers for the selected component (the CSS file). The "Expires" header is highlighted with a red oval.

**Response Headers**

Last-Modified: Wed, 24 Jan 2007 01:57:21 GMT  
Accept-Ranges: bytes  
X-Yahoo-Compressed: true  
Vary: Accept-Encoding  
Content-Type: text/css  
Content-Encoding: gzip  
Cache-Control: public, max-age=604800  
**Expires: Wed, 08 Mar 2017 18:26:40 GMT**  
Content-Length: 1827

At the bottom, there's another table showing other resources:

js	<a href="http://us.js2.yimg.com/.../onload_1.5.5.js">http://us.js2.yimg.com/.../onload_1.5.5.js</a>	7/10/2017	gzip	172	41.4K (16K)
flash	<a href="http://ads.yimg.com/.../072007yahoo_eastman_jelloball_300x250.swf">http://ads.yimg.com/.../072007yahoo_eastman_jelloball_300x250.swf</a>	7/20/2017		109	
cssimage	<a href="http://us.js2.yimg.com/.../toolbar.aif">http://us.js2.yimg.com/.../toolbar.aif</a>	5/6/2017		16	

At the very bottom, there are some status icons and performance metrics: 189.7K, 1.594s, and a progress bar.

## Stats View

YSlow calculates the total size of the web page for both empty cache and primed cache scenarios, as well as information about cookies.

The screenshot shows the YSlow extension interface within a browser. The top navigation bar includes tabs for Inspect, Performance, Stats (which is selected), Components, Tools, Help, and a search bar. Below the tabs are sub-tabs: Console, HTML, CSS, Script, DOM, Net, and YSlow (which is also selected). On the right side of the top bar are buttons for Options and Help.

The main content area is divided into two sections: "Empty Cache" and "Full Cache".

**Empty Cache:**

- 35.3K 1 HTML document (est)
- 1.8K 1 Style Sheet File
- 69.1K 5 JavaScript Files
- 10.8K 1 Flash Object
- 39.2K 14 Images
- 33.2K 10 CSS Images

**Total size:**  
**189.7K Total size**  
**32 HTTP requests**

**Full Cache:**

- 35.3K 1 HTML document (est)
- 0.0K 1 Image

**Total size:**  
**35.3K Total size**  
**2 HTTP requests**

**Cookies:** 42 bytes

- (24 bytes) B=a34qq1p3aagfd&b=3&s=80;
- (6 bytes) FPS=dl;
- (8 bytes) SO=v=0.4;

At the bottom, there are status icons: a green checkmark, a red error icon, and a blue info icon. To the right of these are the values: 189.7K, 1.594s, and a three-dot ellipsis button.

***Then:***

Server-side hints

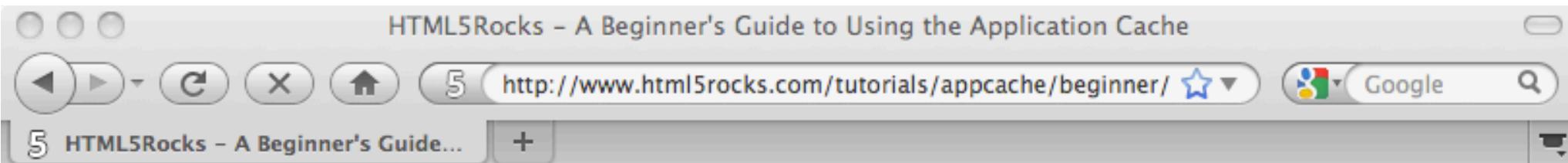
***Now:***

HTML5 Application Cache

# Nº 8.

# LET'S TAKE THIS OFFLINE

OFFLINE SUPPORT						
IE	FIREFOX	SAFARI	CHROME	OPERA	IPHONE	ANDROID
✓	✓	✓	✓	.	✓	✓



Presentation Playground Studio **Tutorials** Resources

[Home](#) > [Tutorials](#) > A Beginner's Guide to Using the Application Cache

# A Beginner's Guide to Using the Application Cache

Eric Bidelman, June 18, 2010

Supported browsers:



70



1

Your browser appears to support all of the functionality used in this sample.

- [Introduction](#)
- [The cache manifest file](#)
  - [Referencing a manifest file](#)
  - [Structure of a manifest file](#)
- [Updating the cache](#)
- [References](#)

Transferring data from www.google.com...



YSlow

# The cache manifest file

The cache manifest file is a simple text file that lists the resources the browser should cache for offline access.

## Referencing a manifest file

To enable the application cache for an app, include the manifest attribute on the document's `html` tag:

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



**Note:** Sites are limited to 5MB worth of cached data.

**Note:** If the manifest file or a resource specified in it fails to download, the entire cache update process fails. The browser will keep using the old application cache in the event of failure.

Lets take a look at a more complex example:

```
CACHE MANIFEST
# 2010-06-18:v2

# Explicitly cached entries
CACHE:
index.html
stylesheet.css
images/logo.png
scripts/main.js

# Resources that require the user to be online.
NETWORK:
login.php
/myapi
http://api.twitter.com

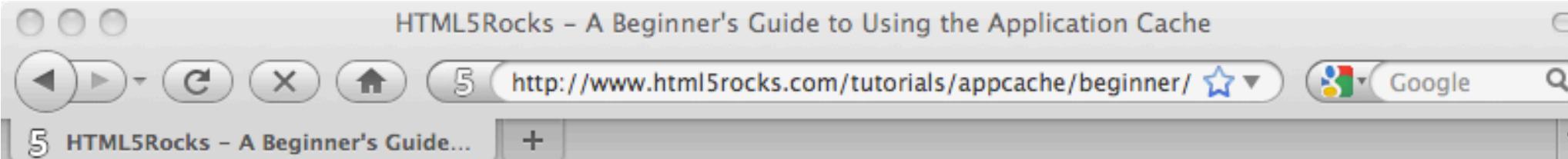
# static.html will be served if main.py is inaccessible
# offline.jpg will be served in place of all images in images/large/
FALLBACK:
/main.py /static.html
images/large/ images/offline.jpg
```

5 HTML5Rocks – A Beginner's Guide...

+

```
var appCache = window.applicationCache;

switch (appCache.status) {
  case appCacheUNCACHED: // UNCACHED == 0
    return 'UNCACHED';
    break;
  case appCache.IDLE: // IDLE == 1
    return 'IDLE';
    break;
  case appCache.CHECKING: // CHECKING == 2
    return 'CHECKING';
    break;
  case appCache.DOWNLOADING: // DOWNLOADING == 3
    return 'DOWNLOADING';
    break;
  case appCache.UPDATEREADY: // UPDATEREADY == 5
    return 'UPDATEREADY';
    break;
  case appCache.OBSOLETE: // OBSOLETE == 5
    return 'OBSOLETE';
    break;
  default:
    return 'UNKNOWN CACHE STATUS';
    break;
};
```



To programmatically update the cache, first call `applicationCache.update()`. This will attempt to update the user's cache (which requires the manifest file to have changed). Finally, when the `applicationCache.status` is in its `UPDATEREADY` state, calling `applicationCache.swapCache()` will swap the old cache for the new one.

```
var appCache = window.applicationCache;  
  
appCache.update(); // Attempt to update the user's cache.  
  
...  
  
if (appCache.status == window.applicationCache.UPDATEREADY) {  
    appCache.swapCache(); // The fetch was successful, swap in the new cache.  
}
```

As you may expect, many events are exposed to monitor these states. The cache interface fires events for things like download progress, updating the app cache, and error conditions. The following snippet sets up event listeners for each type of cache event:

```
function handleCacheEvent(e) {  
    //...  
}  
  
function handleCacheError(e) {  
    alert('Error: Cache failed to update!');  
};
```



Style Master

Tools

Linear  
Gradients

Radial  
Gradients

Text Properties

Box Properties

Transforms

CSS3 Data

## manifestR: get your sites offline

HTML5 features appcaching, a way to make your web sites and apps work offline, and to increase their performance as well.

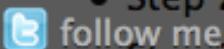
To learn more about the HTML5 appcache, [read our comprehensive overview](#) to get you started.

## So what does manifestR do?

manifestR is a bookmarklet, which you drag to your bookmarks bar. Then, when you visit any page, you can click the manifestR button, and it will create an HTML5 appcache manifest file for that page. For more on how you then use the manifest see [our article](#).

## How do I use it?

- Step 1: drag **manifestR** (that's the link just to the left of this) to the bookmarks bar in your browser
- Step 2: visit any web page you want to create an HTML5 appcache manifest for
- Step 3: click the manifestR button in your bookmarks bar



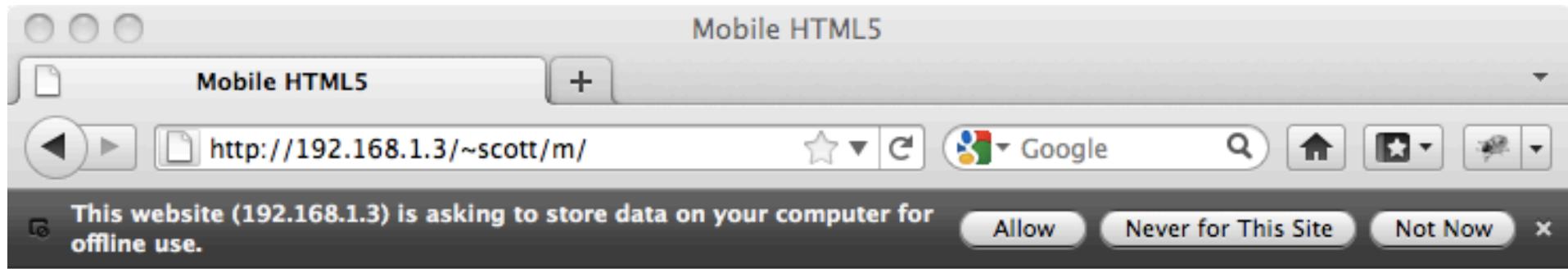
My Book



My Courses



Conferences



- [Dimensions](#)
- [Dimensions 2](#)
- [Orientation](#)
- [News \(Before\)](#)
- [News](#)
- [Touch](#)
- [Drag](#)
- [Gesture](#)

- 
- [Touch/Gesture Events](#)
  - [Simple Draw](#)
  - [YUI Horizontal Scroll](#)

**NOTE: Only Firefox prompts the user when using appcache**

# As of 2011-02-11:

## Application Cache (AppCache) Offline Storage Limits

Offline storage is nowhere near consistent across browsers. This particular application requires (or will when all the audio is recorded) somewhere between 150 MB – 300 MB of storage. At the time of this writing (and my testing) the following holds true:

Browser	Application Cache (AppCache) Storage Limit
Safari Desktop (Mac & Win)	Unlimited
Safari Mobile (iOS)	10 MB
Chrome Desktop (Mac & Win)	5 MB *
Chrome Mobile (Android)	Unlimited **
Firefox 4 Beta	Unlimited (with user prompt)
IE	No idea. It sucks. ***

\* Chrome only allows a limited amount of storage UNLESS you jump through some hoops. More on that later.

\*\* Android appears to allow unlimited storage, at least with the 2.3 API/emulator. However it continually errored out after 1092 files when I had more than that in a particular directory.

<http://grinninggecko.com/developing-cross-platform-html5-offline-app-1/>

# Firefox

The screenshot shows a Firefox browser window with the title "Mobile HTML5". The address bar displays "Mobile HTML5" and the URL "http://192.168.1.3/~scott/m/". The main content area lists several links: Dimensions, Dimensions 2, Orientation, News (Before), News, Touch, Drag, and Gesture.

Below this, a section titled "Touch/Gesture Events" is shown.

The bottom half of the screenshot displays the Firebug developer tools. The toolbar at the top includes icons for file, edit, search, and various developer features. The main pane shows the "DOM" tab selected. A red box highlights the "applicationCache" section of the DOM tree under the "window" object. The "applicationCache" node has four children: "CHECKING" (value 2), "DOWNLOADING" (value 3), "IDLE" (value 1), and "UNCACHED" (value 0). The "UPDATEREADY" node has a value of 4. The "mozItems" node is expanded, showing a "DOMStringList { length=0 }" object. Other properties listed include "mozLength" (0), "oncached" (null), "onchecking" (null), and "ondownloading" (null).

**NOTE: Firebug is broken for AppCache...**

# Firefox

**NOTE: Use this instead...**

## Information about the Cache Service

### Disk cache device

<b>Number of entries:</b>	3676
<b>Maximum storage size:</b>	640000 KiB
<b>Storage in use:</b>	40868 KiB
<b>Cache Directory:</b>	/Users/scott/Library/Caches/Firefox/Profiles /34ivlzc9.default/Cache

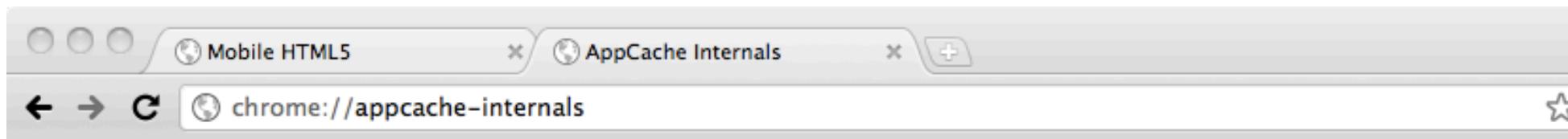
[List Cache Entries](#)

### Offline cache device

<b>Number of entries:</b>	10
<b>Maximum storage size:</b>	512000 KiB
<b>Storage in use:</b>	11 KiB
<b>Cache Directory:</b>	/Users/scott/Library/Caches/Firefox/Profiles /34ivlzc9.default/OfflineCache

[List Cache Entries](#)

# Google Chrome



Manifest: <http://192.168.1.3/~scott/m/m.appcache>

[Remove this AppCache](#)

- Size: 14.1 kB
- Creation Time: Thursday, July 14, 2011 8:55:13 AM
- Last Access Time: Thursday, July 14, 2011 8:55:13 AM
- Last Update Time: Thursday, July 14, 2011 8:55:13 AM

# Google Chrome



- [Dimensions](#)
  - [Dimensions 2](#)
  - [Orientation](#)
  - [News \(Before\)](#)
  - [News](#)
  - [Touch](#)
  - [Drag](#)
  - [Gesture](#)
- 
- [Touch/Gesture Events](#)
  - [Simple Draw](#)
  - [YUI Horizontal Scroll](#)

A screenshot of the Google Chrome DevTools Network tab. The sidebar on the left shows "Frames", "Databases", "Local Storage", "Session Storage", "Cookies", and "Application Cache" sections. The "192.168.1.3" section is selected and expanded, showing a list of resources. The main table lists these resources with their URLs, types, and sizes.

Resource	Type	Size
http://192.168.1.3/~scott/m/	Master	1.46KB
http://192.168.1.3/~scott/m/dim.html	Explicit	2.51KB
http://192.168.1.3/~scott/m/dim2.html	Explicit	981B
http://192.168.1.3/~scott/m/drag/drag.html	Explicit	1.53KB
http://192.168.1.3/~scott/m/drag/gesture.html	Explicit	1.12KB
http://192.168.1.3/~scott/m/drag/touch.html	Explicit	1.06KB
http://192.168.1.3/~scott/m/m.appcache	Manifest	800B
http://192.168.1.3/~scott/m/news-before.html	Explicit	1.40KB
http://192.168.1.3/~scott/m/news.html	Explicit	2.40KB
http://192.168.1.3/~scott/m/orient.html	Explicit	896B



**Local Storage  
(data)**

**Application Cache  
(resources)**

# **HTML5 in Your Pocket:**

## *Application Cache and Local Storage*

**Scott Davis**

ThirstyHead.com





ThirstyHead.com  
training done right.

scott@thirstyhead.com



Scott Davis  
@scottdavis99

Questions?  
Thanks for your time.

# Image Credits

<http://www.flickr.com/photos/rutty/503238148/>

<http://www.flickr.com/photos/daniello/422213306/>

<http://www.flickr.com/photos/f-l-e-x/3096005116/>

<http://www.flickr.com/photos/photohannah/512202109/>

<http://www.flickr.com/photos/jesper/346483297/>

<http://www.flickr.com/photos/stinajonsson/3932774410/>