

# HTML5 Offline Web Applications



**Jfokus 2011  
Stockholm  
Peter Lubbers,  
Kaazing**

# about:peterlubbers

peter lubbers  Search

peter lubbers

peter lubbers **html5** [I'm Feeling Lucky »](#)

peter lubbers kaazing

twitter peter lubbers

---

About 196,000 results (0.37 seconds) [Advanced search](#)

[Run Lake Tahoe](#) ☆

Feb 11, 2010 ... **Peter Lubbers'** blog about running and ultrarunning. Peter is the 2007 and 2009 ultrarunner.net series champion and three-time winner of the ...

[runlaketahoe.blogspot.com/](#) [Cached](#) [Similar](#)



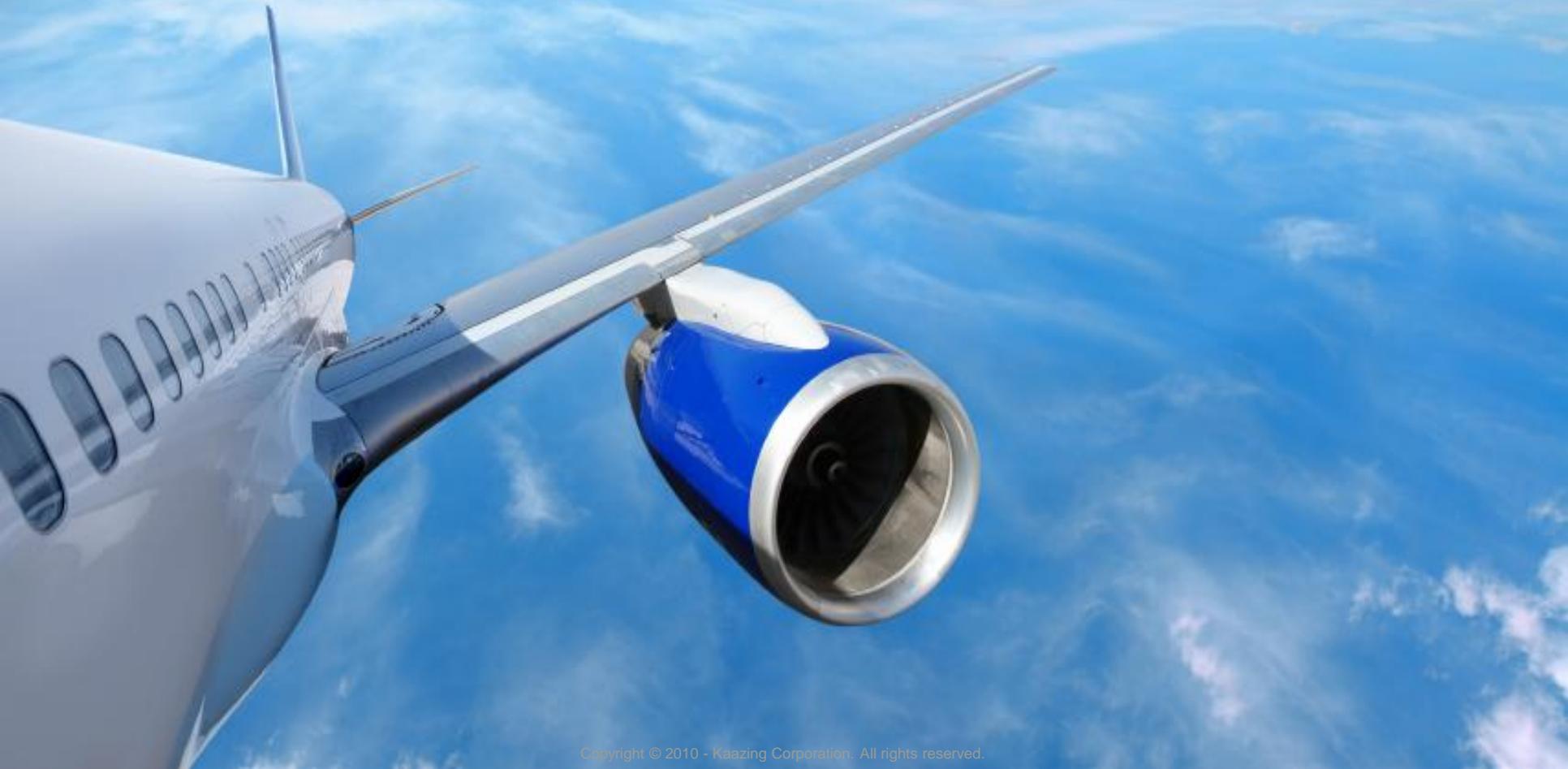


# Agenda

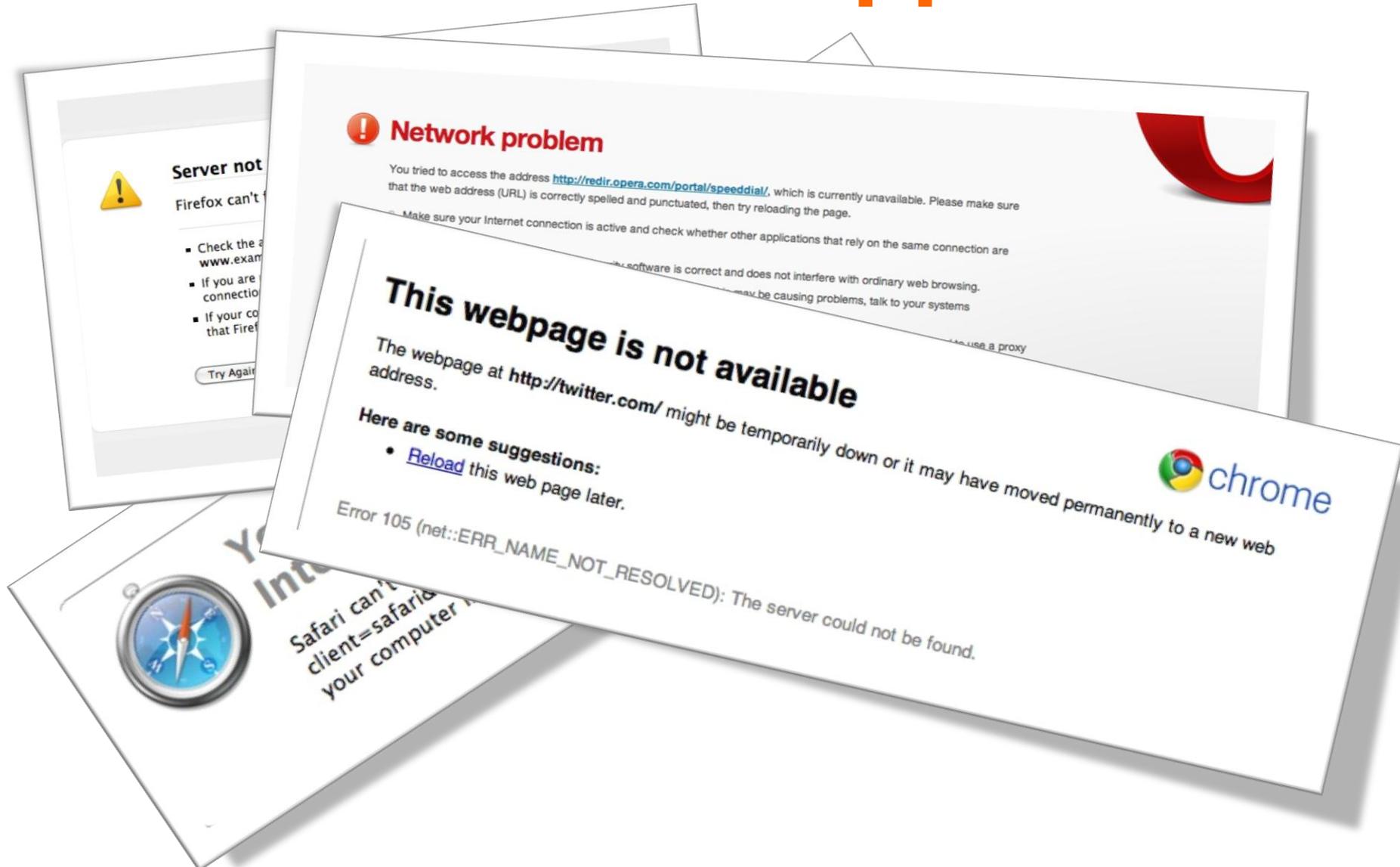
- About Offline Web Apps
- Creating Offline Web Apps
- Server Configuration
- Tips, Tricks, and Q&A
- Pay attention, win prizes!
- Tweet it: #jfokus @peterlubbers  
#html5 #appcache

# HTML5 Offline Web Apps

Allow you to keep using web apps and sites without a network connection (for example, on an airplane or in rural areas and subways)

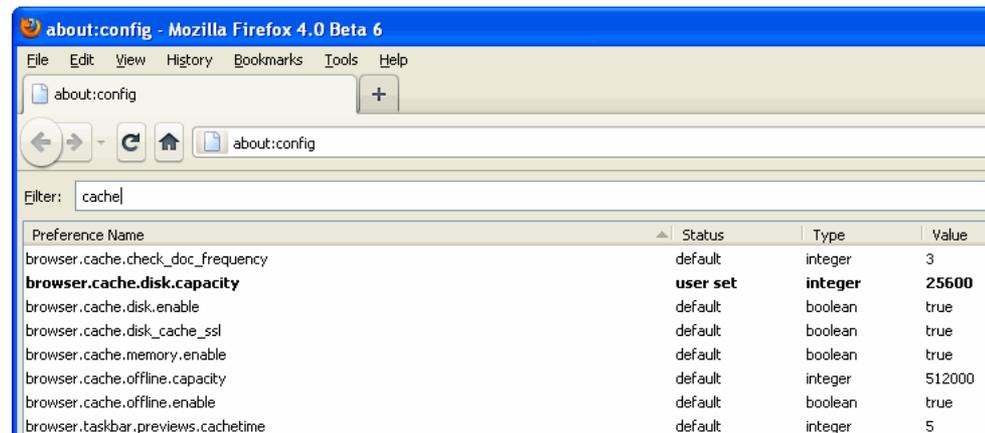
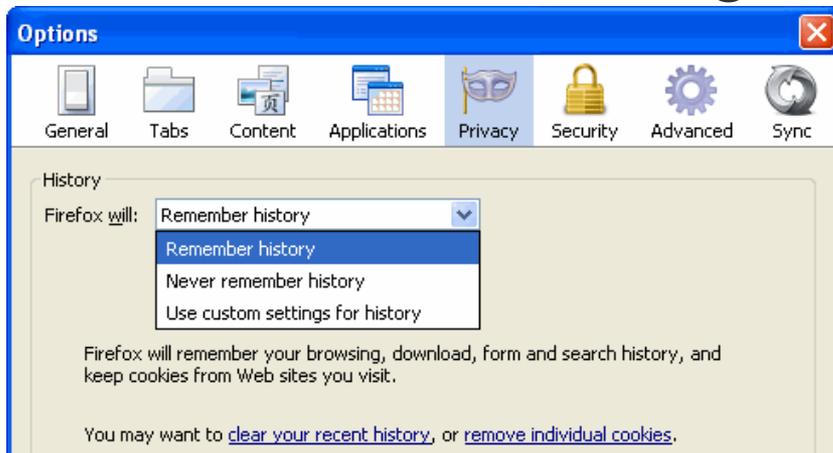


# Offline without AppCache



# Traditional Caching

- Cache duration can be controlled through Web Server's Cache HTTP headers
  - For example, in Apache's `.htaccess` file (see example on next slide)
- Not reliable for offline use
- Browser caching settings also apply



# Web Server Caching Settings

## .htaccess File

```
<IfModule mod_expires.c>
  Header set cache-control: public
  ExpiresActive on
# your document html
  ExpiresByType text/html          "access"
# rss feed
  ExpiresByType application/rss+xml "access plus 1 hour"
# favicon (cannot be renamed)
  ExpiresByType image/vnd.microsoft.icon "access plus 1 week"
# media: images, video, audio
  ExpiresByType image/png          "access plus 1 month"
  ExpiresByType image/jpg          "access plus 1 month"
  ExpiresByType image/jpeg         "access plus 1 month"
  ExpiresByType video/ogg          "access plus 1 month"
  ExpiresByType audio/ogg          "access plus 1 month"
  ExpiresByType video/mp4          "access plus 1 month"
# webfonts
  ExpiresByType font/ttf            "access plus 1 month"
  ExpiresByType font/woff           "access plus 1 month"
  ExpiresByType image/svg+xml      "access plus 1 month"
# css and javascript
  ExpiresByType text/css            "access plus 1 month"
  ExpiresByType application/javascript "access plus 1 month"
  ExpiresByType text/javascript    "access plus 1 month"
</IfModule>
```

**Source:** Paul Irish & Divya Manian <http://html5boilerplate.com/>

# Offline Web Applications

- New HTML5 Specification
- Also known as Application Cache (AppCache)
- WHATWG: <http://www.whatwg.org/specs/web-apps/current-work/multipage/offline.html#offline>
- W3C:  
<http://dev.w3.org/html5/spec/offline.html#offline>
- Spec is primarily aimed at browser developers to ensure interoperability

# Offline Web Applications

- Allow you to cache pages that have *not* been visited
- Browsers cache data in an *Application Cache*
- Once resources are cached, you can access them very quickly (without a network request)
- HTML5 also allows online and offline detection
- Using offline mechanism allows you to easily *prefetch* site resources (speeds up pages, but uses bandwidth)
- Pages served using TLS (SSL) can also be included to work offline

# Browser Support for Offline Web Applications



4.0+



3.5+



10.6+



4.0+



Some day...

Source: <http://caniuse.com/>  
(the best site for checking browser support for HTML5 features)

Hi-res browser logos: <http://paulirish.com/2010/high-res-browser-icons/>

# The idea is simple...

Add a manifest file. reference it.

Done.

But in reality...



# Developing Offline Web Applications

<html>

<head>

<title>

<stv>

# Example Manifest File

## Manifest File

```
CACHE MANIFEST
# manifest version 1.0.1
# Files to cache
index.html
cache.html
html5.css
image1.jpg
img/foo.gif
http://www.example.com/styles.css

# Use from network if available
NETWORK:
network.html

# Fallback content
FALLBACK:
/ fallback.html
```

# Using a Manifest File in HTML

- Reference the manifest file:
  - **<name>.manifest** – it must match the name of the manifest file
  - Add as attribute to HTML element

## HTML

```
<!DOCTYPE html>
<html manifest="offline.manifest">
  <head>
    <title>HTML5 Application Cache</title>
```

Files: <http://bit.ly/9pJ1Zq> or:  
[ch.kaazing.com/training/offline/peter-lubbers-html5-offline-web-apps-presentation-code.zip](http://ch.kaazing.com/training/offline/peter-lubbers-html5-offline-web-apps-presentation-code.zip)



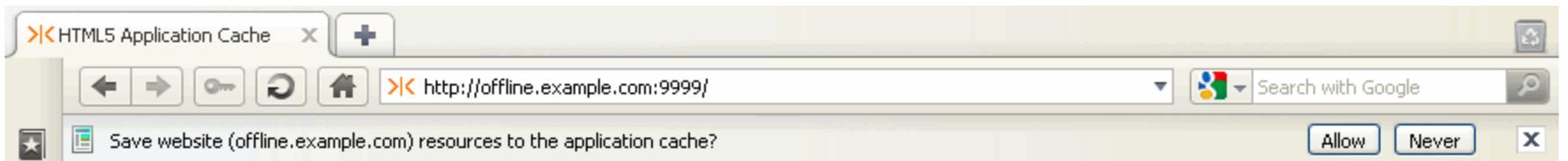
# Demo

# Initial Cache Sequence

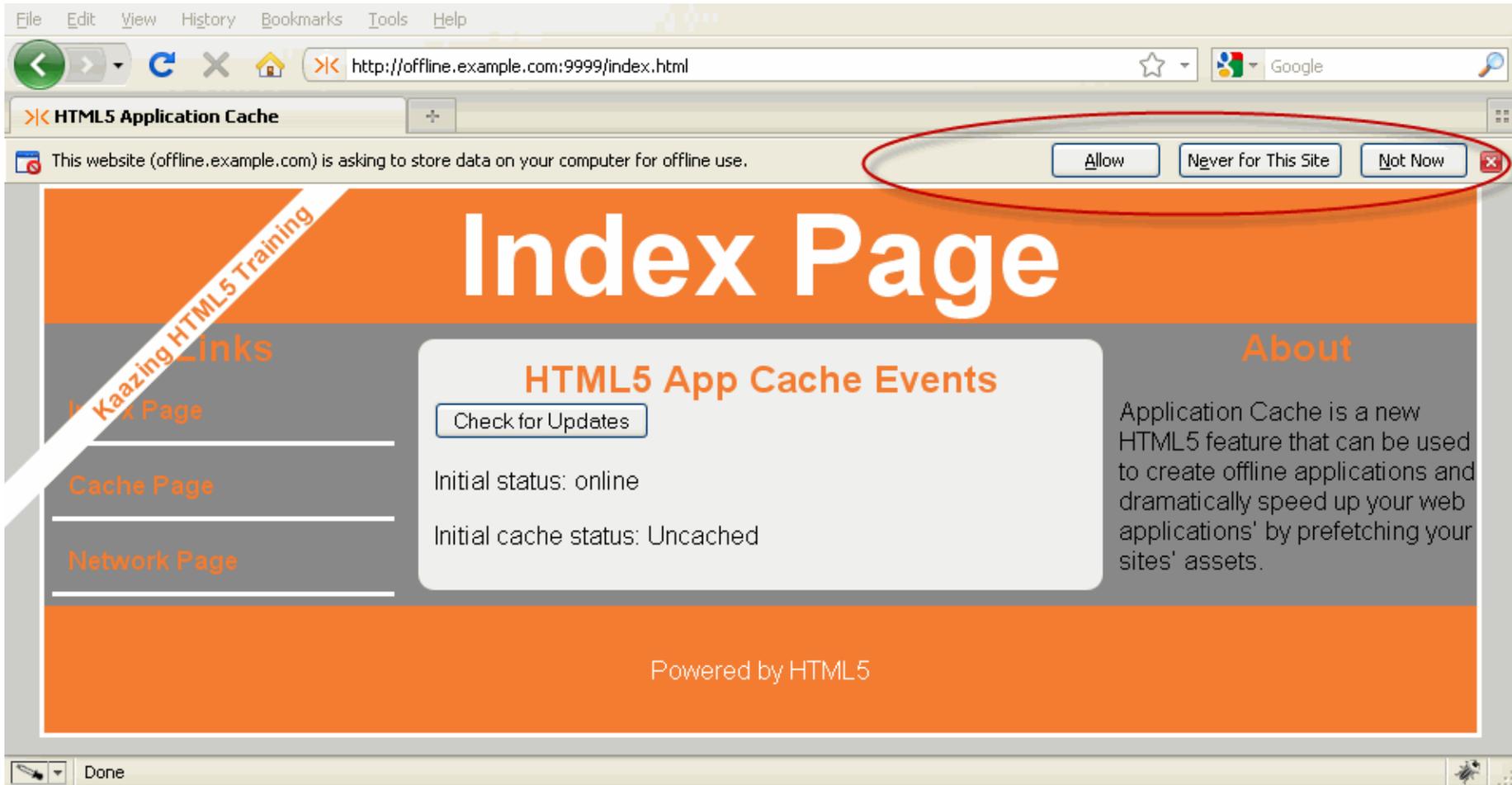
1. Access the page (with the manifest attribute)
2. Page is loaded and page's resources are loaded (from the server)
3. Manifest is encountered and parsed, all files flagged for caching are loaded in the background
4. Go offline (regular caching is also in effect, so watch for false positives)
5. Access a `CACHE` : resource (loads from cache)
6. Access a `NETWORK` resource (`FALLBACK` content is served, files will be available if you go back online)

# Browser Notification

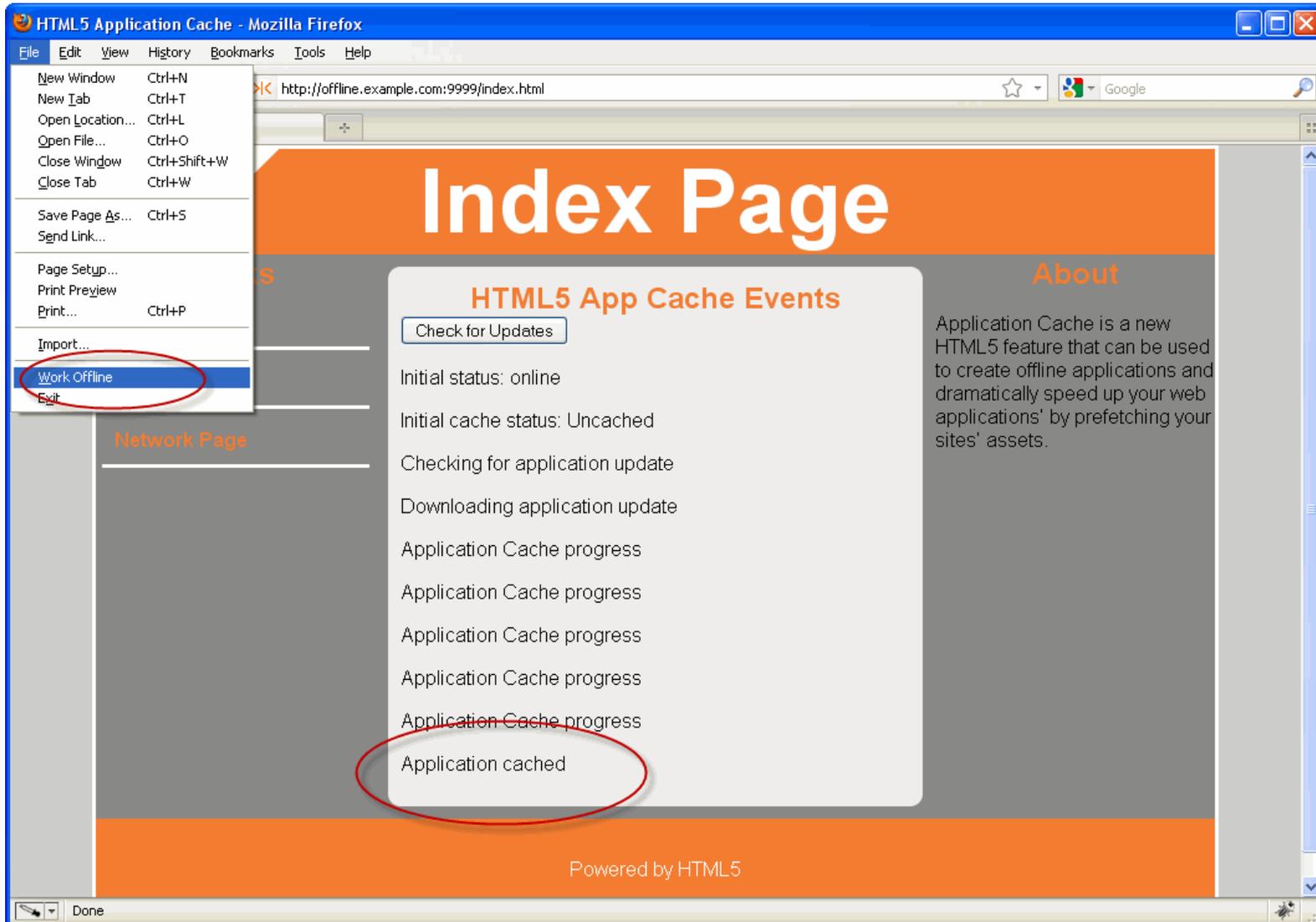
- Users may have to opt in (similar to Geolocation)
- Configurable
- Not default in all browsers



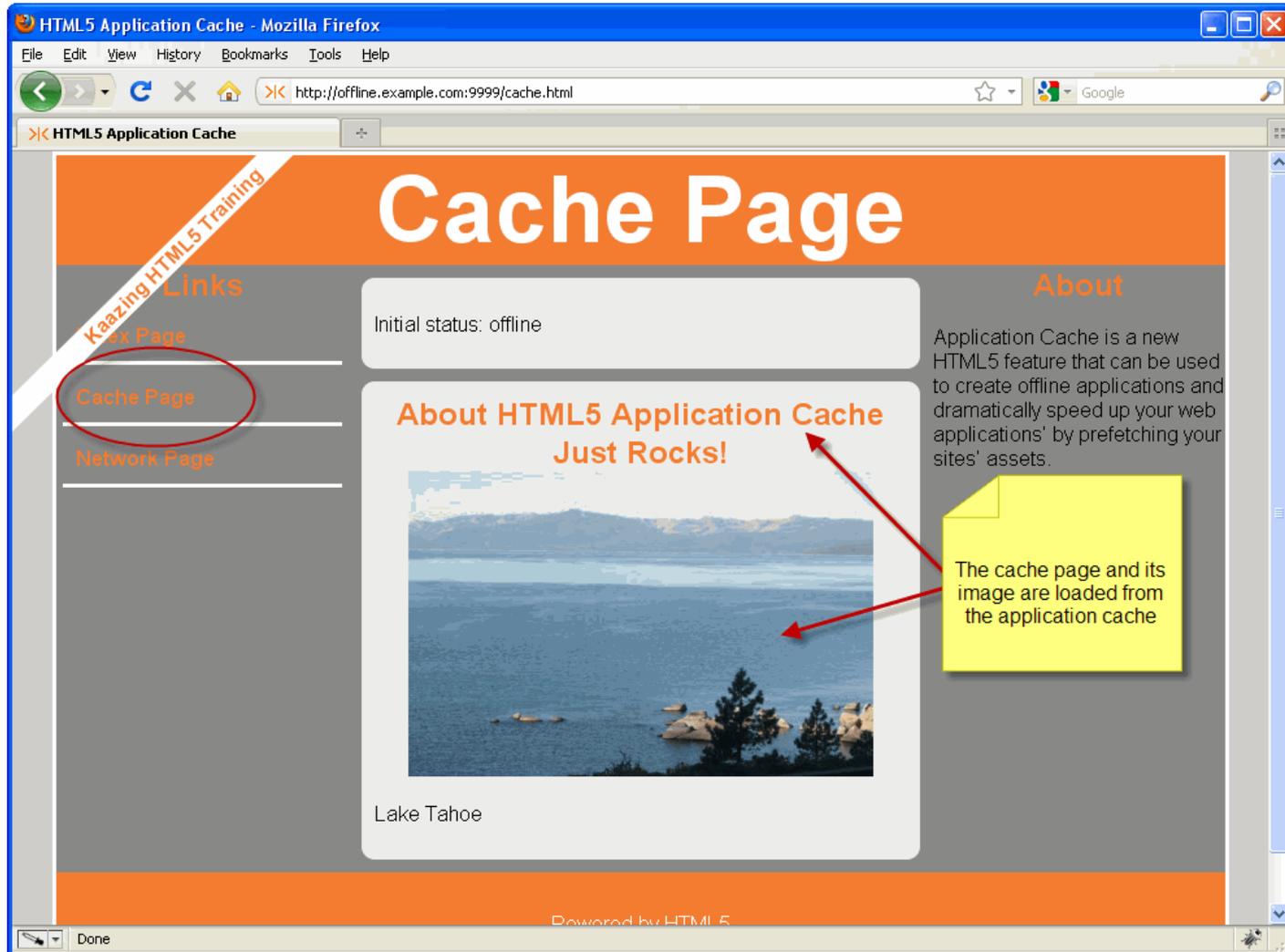
# Step 1: First Page Load



# Step 2: Going Offline



# Step 3: Offline Web Page



# Step 4: Network Page

HTML5 Application Cache - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://offline.example.com:9999/network.html

HTML5 Application Cache

**Fallback Page**

Kaazing HTML5 Training

Initial status: offline

**Fallback Content**

You must be online to use the page you requested...

Powered by HTML5

Done

Navigating to the Network Page while in offline mode loads the fallback content.

# Consecutive Load Sequence 1

1. Go back to online mode
2. Change the `cache.html` page on the server
3. Reload the `cache.html` page in the browser
4. The (old) page loads from the application cache  
(The changes do not appear!)
5. The browser checks to see if the referenced `manifest` file has been updated and does nothing since it has not been modified

# Consecutive Load Sequence 2

6. Update the manifest file (make a trivial change, like a version number comment update)
7. Reload the cache page in the browser
8. The (old) page loads from the application cache (This is always the first action from the browser, so the changes *still* don't appear!)
9. The browser checks to see if the referenced manifest has been updated and since it has, it downloads all the files flagged to be cached
10. The new files are now in the application cache, reload the page once more to see the latest changes

</Demo>

# The Manifest File

- Manifest file has three sections:
  - CACHE :
  - NETWORK :
  - FALLBACK :
- Multiple sections (of the same kind) are allowed
- First line *must* be `CACHE MANIFEST`
- Comments start with `#` (don't use inline comments)
- An application cache is created using the manifest's complete URL (you can have multiple manifest files in a site)

# CACHE: Section

- To cache files in the AppCache, include them in the `CACHE: section` or list files directly under `CACHE MANIFEST` (default is to cache files)
- Add one file per line (full name required)
- Files can contain path information or even be an absolute URL
- Application caches can't include fragment identifiers (`#`) or wildcards
- Case sensitive
- Files that reference the manifest file will automatically be cached

# NETWORK: Section

- Also called the online whitelist
- Files listed in this section listed will not be loaded from the cache, but retrieved over the network (from the server) if online
- You can specify “\*”
  - Sets the online whitelist wildcard flag to “open”
  - Access to resources on other origins will not be blocked

# FALLBACK: Section

- Provides a way to fall back if resources cannot be found
- Specify a fallback namespace and a fallback page for that namespace:  
`/ fallback.html`
- You can only list one fallback namespace

# Checking for Browser Support

## JavaScript

```
// Simple method
if(window.applicationCache) {
    // this browser supports offline web apps
}

//Or just use Modernizr (source: http://www.modernizr.com/)
if (Modernizr.applicationcache){
    // We have offline web app support! Continue
    operation,
    // indicating to the user that the app will sync up
    once they get back online
} else {
    // No offline support, show errors if the user goes
    offline
}
```

# Checking for Online and Offline Events

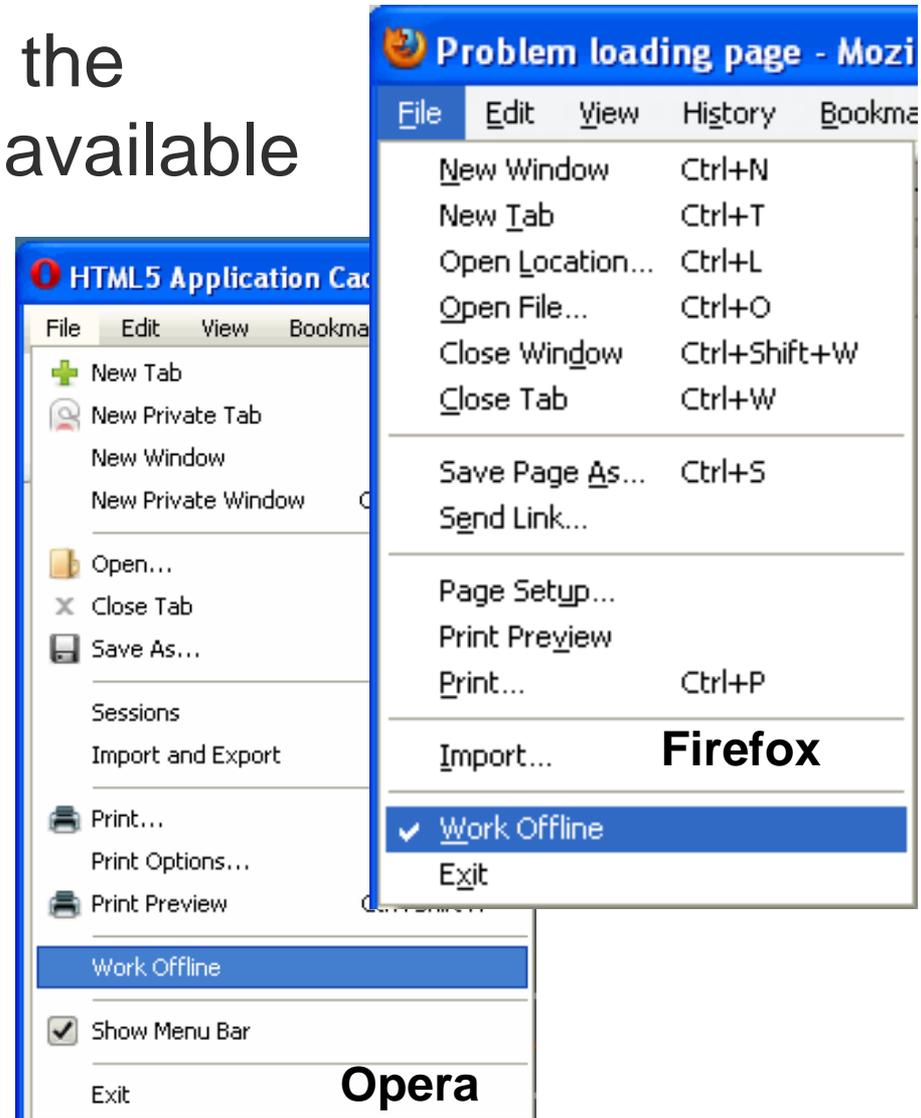
## JavaScript

```
window.addEventListener("online", function(e) {  
    log("Application is now online");  
}, true);
```

```
window.addEventListener("offline", function(e) {  
    log("Application is now offline");  
}, true);
```

# Using Web Pages Offline

- Test offline pages using the “Work Offline” feature if available (Not in Chrome and Safari)
- Disconnect your computer (does not work for localhost)
- Watch out for false positives (regular browser caching)



# Working Offline

Browser	Work Offline
Chrome	Disconnect from the network
Firefox	File > Work Offline
Safari	Disconnect from the network
Opera	File > Work Offline
Internet Explorer	N/A (Does not support App Cache)

# ApplicationCache Events

- The `window.applicationCache` object fires several events related to the state of the cache
- `window.applicationCache.status` is a numerical property indicating the state of the cache
  - 0 UNCACHED
  - 1 IDLE
  - 2 CHECKING
  - 3 DOWNLOADING
  - 4 UPDATEREADY
  - 5 OBSOLETE

# Event Callback Attributes

Callback Attribute	Event
onchecking	CHECKING
ondownloading	DOWNLOADING
onupdateready	UPDATEREADY
onobsolete	OBSOLETE
oncached	CACHED
onerror	ERROR
onnoupdateready	NOUPDATE
onprogress	PROGRESS

# Checking App Cache Status

## JavaScript

```
window.applicationCache.onchecking = function(e) {
    log("Checking for application update");
}
window.applicationCache.onnoupdate = function(e) {
    log("No application update found");
}
window.applicationCache.onupdateready = function(e) {
    log("Application update ready");
    window.applicationCache.swapCache();
}
window.applicationCache.onobsolete = function(e) {
    log("Application obsolete");
}
```

# Checking App Cache Status

## JavaScript

```
window.applicationCache.ondownloading = function(e) {  
    log("Downloading application update");  
}  
window.applicationCache.ondcached = function(e) {  
    log("Application cached");  
}  
window.applicationCache.onerror = function(e) {  
    log("Application cache error");  
}  
window.applicationCache.onprogress = function(e) {  
    log("Application Cache progress");  
}
```



# *Server Configuration*

# Serving the Manifest File

- Manifest files have the MIME type `text/cache-manifest`
- Most web servers need to be configured to serve the manifest files correctly
  - Served correctly by default by Apache on Ubuntu Linux and Kaazing WebSocket Gateway
  - For Python on Windows/Mac OS X or Apache, update the configuration files with the MIME type

# Apache Configuration

## `mime.types` File

```
# Apache mimetype configuration
# APACHE_HOME/conf/mime.types
text/cache-manifest manifest
```

Or:

## `.htaccess` File

```
# Apache mimetype configuration
AddType text/cache-manifest .manifest
```

# Prevent Manifest Caching

## .htaccess File

```
# Cache settings for the manifest file
<IfModule mod_expires.c>
  Header set cache-control: public
  ExpiresActive on
  .
  .
  .
# Prevent receiving a cached manifest
  ExpiresByType text/cache-manifest      "access plus 0 seconds"
  .
  .
  .
</IfModule>
```

**Source:** *Introducing HTML5*, Offline Chapter, Bruce Lawson and Remy Sharp

# Python Configuration

## mimetypes.py File

```
# Python SimpleHTTPServer mimetype Configuration
# python -m SimpleHTTPServer 9999)
'.manifest'      : 'text/cache-manifest',
```

### Windows:

*PYTHON\_HOME*/Lib/mimetypes.py, for example:

C:\Python26\Lib\mimetypes.py

### Mac:

*PYTHON\_HOME*/Lib/mimetypes.py, for example:

/System/Library/Frameworks/Python.framework/Versions/2.6/lib/  
python2.6/mimetypes.py

**Important:** If you do not have a `mimetypes.py` file, you can use `mimetypes.py` from the `offline/ mac-config-file example` folder. If you already have a compiled `mimetypes.pyc` file in the same directory, ensure that the permissions on this file are changed to read/write. When you start Python with the new file, Python compiles it and generates or overwrites the `mimetypes.pyc`.

# Microsoft IIS Configuration

The image shows a screenshot of the Microsoft IIS Manager interface. The main window is titled "Default Web Site Home" and displays a grid of configuration options. A yellow circle highlights the "MIME Types" icon, and a blue arrow points from it to a dialog box titled "Add MIME Type".

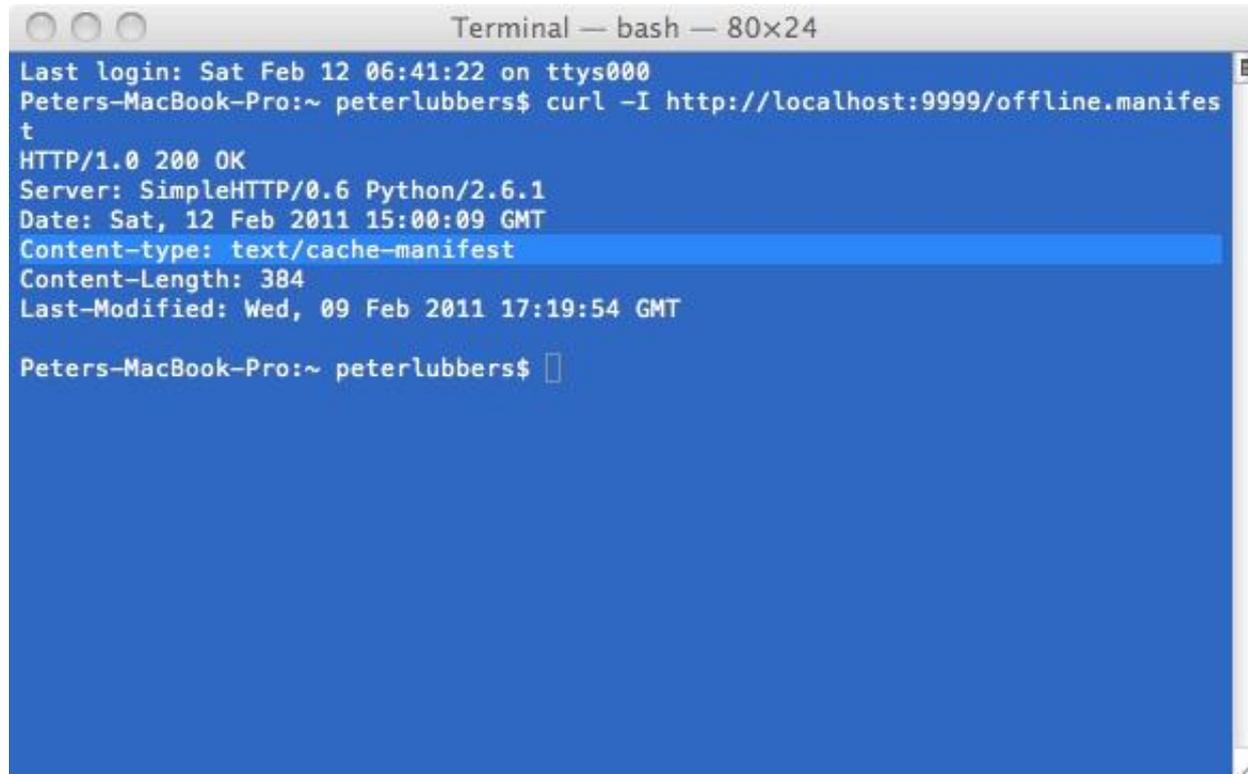
The "Add MIME Type" dialog box contains the following fields:

- File name extension:
- MIME type:

At the bottom of the dialog box, there are "OK" and "Cancel" buttons.

# Verify Mime Type

- Verify that the manifest file is served correctly:
  - `curl -I http://offline.example.com/offline.manifest | **`
  - Check Chrome Dev Tools console

A screenshot of a terminal window titled "Terminal — bash — 80x24". The terminal shows the output of a curl command: "curl -I http://localhost:9999/offline.manifest". The output is: "HTTP/1.0 200 OK", "Server: SimpleHTTP/0.6 Python/2.6.1", "Date: Sat, 12 Feb 2011 15:00:09 GMT", "Content-type: text/cache-manifest", "Content-Length: 384", and "Last-Modified: Wed, 09 Feb 2011 17:19:54 GMT". The prompt "Peters-MacBook-Pro:~ peterlubbers\$" is visible at the bottom.

```
Terminal — bash — 80x24
Last login: Sat Feb 12 06:41:22 on ttys000
Peters-MacBook-Pro:~ peterlubbers$ curl -I http://localhost:9999/offline.manifest
HTTP/1.0 200 OK
Server: SimpleHTTP/0.6 Python/2.6.1
Date: Sat, 12 Feb 2011 15:00:09 GMT
Content-type: text/cache-manifest
Content-Length: 384
Last-Modified: Wed, 09 Feb 2011 17:19:54 GMT

Peters-MacBook-Pro:~ peterlubbers$
```

**\*\*Source:** *Introducing HTML5*, Offline Chapter, Bruce Lawson and Remy Sharp

# Chrome Developer Tools

The screenshot displays the Chrome Developer Tools interface. The top navigation bar includes tabs for Elements, Resources, Scripts, Timeline, Profiles, Storage, Audits, and Console. The Storage tab is active, showing a search bar and a table of resources. The left sidebar is categorized into Databases, Local Storage, Session Storage, Cookies, and Application Cache. The Application Cache section is expanded, showing a list of resources for 'offline.example.com'. Below the table, there are refresh and close icons, and a status indicator 'UNCACHED'. The console at the bottom shows a sequence of events related to the application cache, including document loading, checking, downloading, and progress events for various resources.

Resource	Type	Size
http://kaazing.me/assets/image/kaazing-tab.png	Explicit	18.47KB
http://offline.example.com:9999/	Master	3.72KB
http://offline.example.com:9999/cache.html	Explicit	1.84KB
http://offline.example.com:9999/fallback.html	Fallback	1.91KB
http://offline.example.com:9999/html5.css	Explicit	1.91KB
http://offline.example.com:9999/image1.jpg	Explicit	16.22KB
http://offline.example.com:9999/index.html	Explicit	3.72KB
http://offline.example.com:9999/offline.manifest	Manifest	694B

Document was loaded from Application Cache with manifest <http://offline.example.com:9999/offline.manifest>

Application Cache Checking event

Application Cache Downloading event

Application Cache Progress event (0 of 6) <http://offline.example.com:9999/image1.jpg>

Application Cache Progress event (1 of 6) <http://offline.example.com:9999/html5.css>

Application Cache Progress event (2 of 6) <http://offline.example.com:9999/cache.html>

Application Cache Progress event (3 of 6) <http://offline.example.com:9999/index.html>

Application Cache Progress event (4 of 6) <http://offline.example.com:9999/>

Application Cache Progress event (5 of 6) <http://offline.example.com:9999/fallback.html>

Application Cache Progress event (6 of 6)

Application Cache UpdateReady event

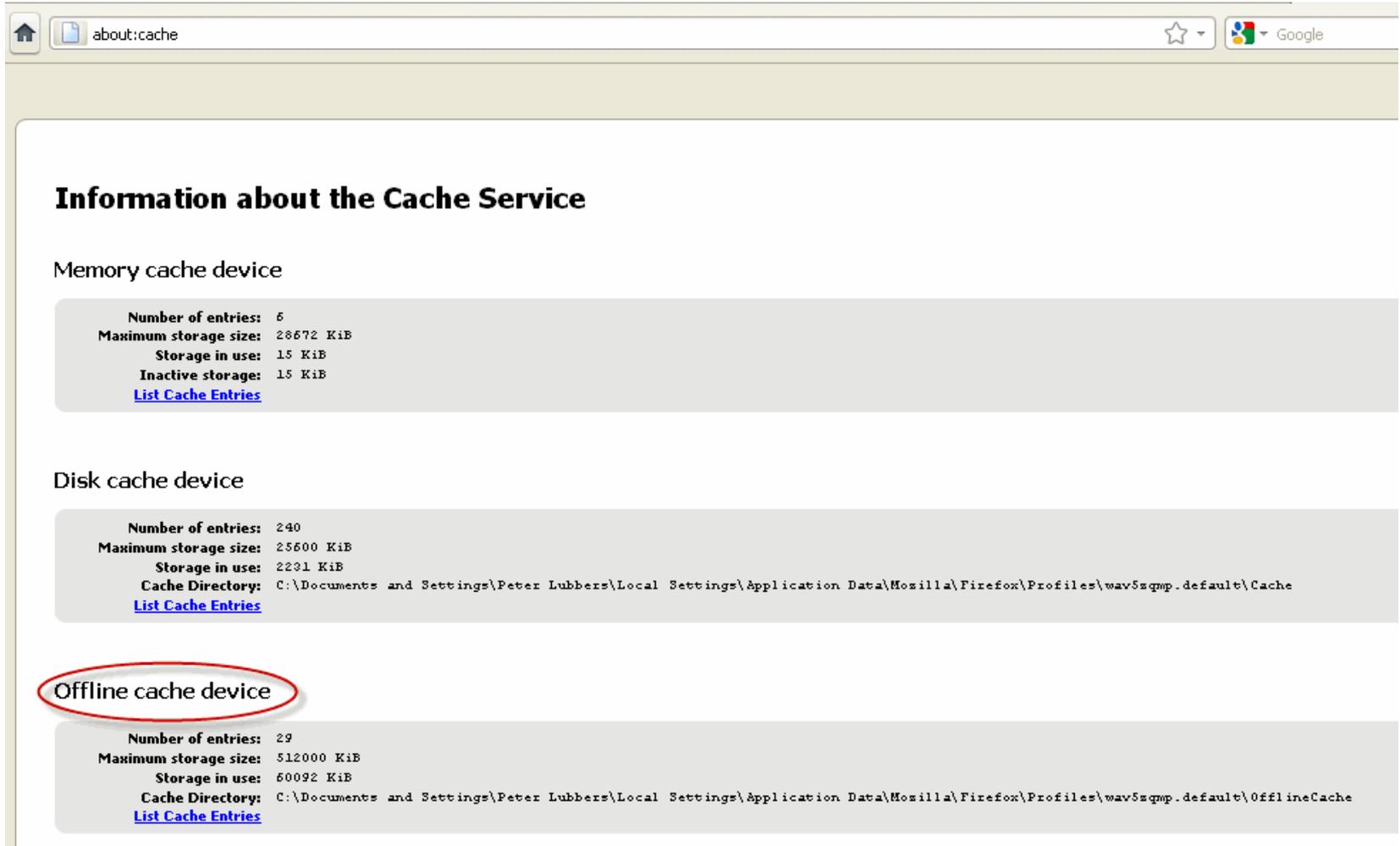
>

All Errors Warnings Logs

# Offline Web Applications Tips and Tricks



# Accessing the Offline Cache



The screenshot shows the Firefox browser's 'about:cache' page. The address bar contains 'about:cache'. The page title is 'Information about the Cache Service'. There are three sections: 'Memory cache device', 'Disk cache device', and 'Offline cache device'. The 'Offline cache device' section is circled in red. Each section lists statistics like 'Number of entries', 'Maximum storage size', 'Storage in use', and 'Inactive storage', along with a 'Cache Directory' path and a 'List Cache Entries' link.

**Information about the Cache Service**

Memory cache device

- Number of entries: 6
- Maximum storage size: 28672 KiB
- Storage in use: 15 KiB
- Inactive storage: 15 KiB
- [List Cache Entries](#)

Disk cache device

- Number of entries: 240
- Maximum storage size: 25600 KiB
- Storage in use: 2201 KiB
- Cache Directory: C:\Documents and Settings\Peter Lubbers\Local Settings\Application Data\Mozilla\Firefox\Profiles\wav5sqmp.default\Cache
- [List Cache Entries](#)

**Offline cache device**

- Number of entries: 29
- Maximum storage size: 512000 KiB
- Storage in use: 60092 KiB
- Cache Directory: C:\Documents and Settings\Peter Lubbers\Local Settings\Application Data\Mozilla\Firefox\Profiles\wav5sqmp.default\OfflineCache
- [List Cache Entries](#)

Firefox: **about:cache**

# Accessing the Offline Cache

The screenshot shows the SQLite Manager interface with the following details:

- Database:** C:\Documents and Settings\Peter Lubbers\Local Settings\Application Data\Mozilla\Firefox\Profiles\wav5zqmp.default\OfflineCache\index.sqlite
- Table:** moz\_cache
- Table Structure:**

rowid	ClientID	Key	MetaData
1	http%3A//localhost%3A9999/tracker.manifest 0000001279856699 0	http://localhost:9999/tracker.manifest	BLOB (Size: 283)
2	http%3A//localhost%3A9999/tracker.manifest 0000001279856699 0	http://localhost:9999/offline.js	BLOB (Size: 233)
3	http%3A//localhost%3A9999/tracker.manifest 0000001279856699 0	http://localhost:9999/log.js	BLOB (Size: 232)
4	http%3A//localhost%3A9999/tracker.manifest 0000001279856699 0	http://localhost:9999/html5.css	BLOB (Size: 217)
5	http%3A//localhost%3A9999/tracker.manifest 0000001279856699 0	http://localhost:9999/tracker.html	BLOB (Size: 236)
66	http%3A//main.example.com%3A9999/offline.manifest 0000001284640985 16	http://main.example.com:9999/offline.manifest	BLOB (Size: 274)
67	http%3A//main.example.com%3A9999/offline.manifest 0000001284640985 16	http://main.example.com:9999/index.html	BLOB (Size: 218)
68	http%3A//main.example.com%3A9999/offline.manifest 0000001284640985 16	http://main.example.com:9999/cache.html	BLOB (Size: 218)
69	http%3A//main.example.com%3A9999/offline.manifest 0000001284640985 16	http://main.example.com:9999/html5.css	BLOB (Size: 217)
70	http%3A//main.example.com%3A9999/offline.manifest 0000001284640985 16	http://main.example.com:9999/image1.jpg	BLOB (Size: 220)
71	http%3A//main.example.com%3A9999/offline.manifest 0000001284640985 16	http://kaazing.me/assets/Image/kaazing-tab.png	BLOB (Size: 268)
72	http%3A//main.example.com%3A9999/offline.manifest 0000001284640985 16	http://main.example.com:9999/Intermission-Walk-in_512kb.mp4	BLOB (Size: 222)
73	http%3A//main.example.com%3A9999/offline.manifest 0000001284640985 16	http://main.example.com:9999/Intermission-Walk-in.ogv	BLOB (Size: 237)
74	http%3A//main.example.com%3A9999/offline.manifest 0000001284640985 16	http://main.example.com:9999/fallback.html	BLOB (Size: 218)
75	http%3A//localhost%3A9999/offline/offline.manifest 0000001285006715 2	http://localhost:9999/offline/offline.manifest	BLOB (Size: 274)
76	http%3A//localhost%3A9999/offline/offline.manifest 0000001285006715 2	http://localhost:9999/offline/index.html	BLOB (Size: 232)
77	http%3A//localhost%3A9999/offline/offline.manifest 0000001285006715 2	http://localhost:9999/offline/cache.html	BLOB (Size: 232)
78	http%3A//localhost%3A9999/offline/offline.manifest 0000001285006715 2	http://localhost:9999/offline/html5.css	BLOB (Size: 217)
79	http%3A//localhost%3A9999/offline/offline.manifest 0000001285006715 2	http://localhost:9999/offline/image1.jpg	BLOB (Size: 220)
80	http%3A//localhost%3A9999/offline/offline.manifest 0000001285006715 2	http://localhost:9999/offline/Intermission-Walk-in_512kb.mp4	BLOB (Size: 222)
81	http%3A//localhost%3A9999/offline/offline.manifest 0000001285006715 2	http://localhost:9999/offline/Intermission-Walk-in.ogv	BLOB (Size: 237)
82	http%3A//localhost%3A9999/offline/offline.manifest 0000001285006715 2	http://localhost:9999/offline/fallback.html	BLOB (Size: 218)
83	http%3A//localhost%3A9999/offline/offline.manifest 0000001285006715 2	http://localhost:9999/offline/	BLOB (Size: 218)
105	http%3A//offline.example.com%3A9999/offline.manifest 0000001285242704 12	http://offline.example.com:9999/offline.manifest	BLOB (Size: 274)
106	http%3A//offline.example.com%3A9999/offline.manifest 0000001285242704 12	http://offline.example.com:9999/index.html	BLOB (Size: 254)
107	http%3A//offline.example.com%3A9999/offline.manifest 0000001285242704 12	http://offline.example.com:9999/cache.html	BLOB (Size: 232)
108	http%3A//offline.example.com%3A9999/offline.manifest 0000001285242704 12	http://offline.example.com:9999/html5.css	BLOB (Size: 217)
109	http%3A//offline.example.com%3A9999/offline.manifest 0000001285242704 12	http://offline.example.com:9999/image1.jpg	BLOB (Size: 220)
110	http%3A//offline.example.com%3A9999/offline.manifest 0000001285242704 12	http://offline.example.com:9999/fallback.html	BLOB (Size: 218)
- Navigation:** 1 to 29 of 29
- Status:** SQLite 3.7.1, Gecko 2.0b6, 0.6.2, Exclusive, Number of files in selected directory: 14

Firefox SQLite Manager Add-on: <https://addons.mozilla.org/en-US/firefox/addon/5817/>

# Accessing the Offline Cache

Address  C:\Documents and Settings\Peter Lubbers\Local Settings\Application Data\Mozilla\Firefox\Profiles\wav5zqmp.default\OfflineCache

Folders	Name	Size	Type	Date Modified
 Mozilla	0		File Folder	9/20/2010 11:17 AM
 Firefox	2		File Folder	9/16/2010 5:42 AM
 Mozilla Firefox	3		File Folder	9/15/2010 5:26 PM
 Mozilla Firefox 4.0 Beta 1	4		File Folder	9/20/2010 11:18 AM
 Profiles	5		File Folder	9/15/2010 5:23 PM
 7jec8430.KAAZINGME	6		File Folder	9/20/2010 11:17 AM
 wav5zqmp.default	7		File Folder	9/20/2010 11:18 AM
 Cache	8		File Folder	9/16/2010 5:42 AM
 OfflineCache	9		File Folder	9/16/2010 5:43 AM
 0	A		File Folder	9/20/2010 11:17 AM
 B	B		File Folder	9/16/2010 5:43 AM
 2	C		File Folder	7/22/2010 2:04 PM
 3	D		File Folder	9/20/2010 11:17 AM
 4	E		File Folder	9/20/2010 11:17 AM
 5	F		File Folder	9/16/2010 5:43 AM
 6	index.sqlite	53 KB	SQLITE File	9/28/2010 4:36 PM
 7				
 8				
 9				
 A				
 B				
 C				
 D				
 E				
 F				
 startupCache				
 Opera				
 Opera				
 application_cache				
 c432bf417049036fb93552a64d46425c				
 g_0000				
 pstorage				
 00				
 OA				
 sesn				
 mcache				
 sesn				
 bt_metadata				

File System Access

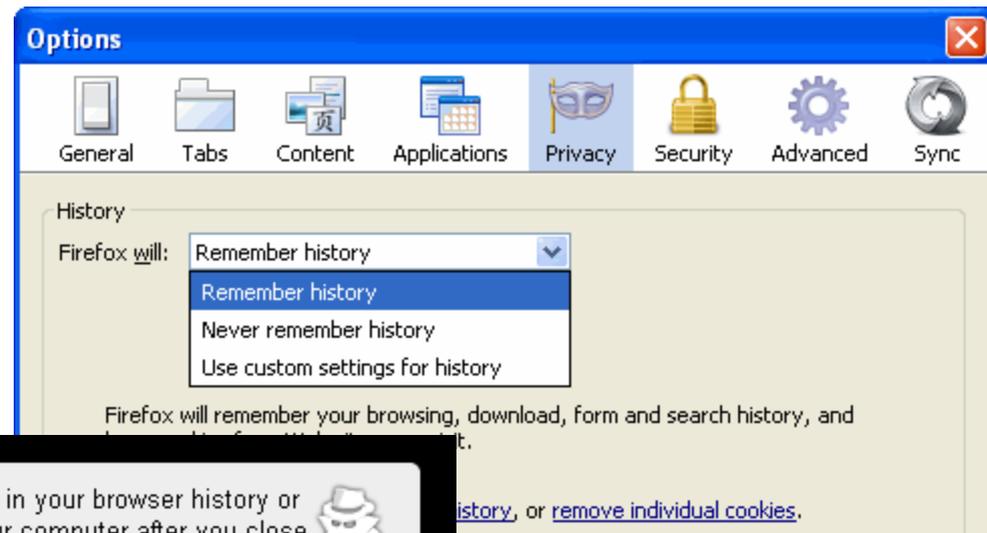
# Security Considerations

- Others browsing the same site (on the same machine in the same browser) can potentially access your cached data (data is cached based on the manifest file URL)
- Do not store sensitive, personal data in the application cache



# Private Browsing

- Most private browsing modes prevent writing to application cache
  - For example, Safari's Private Browsing mode, and Chrome's Incognito Mode

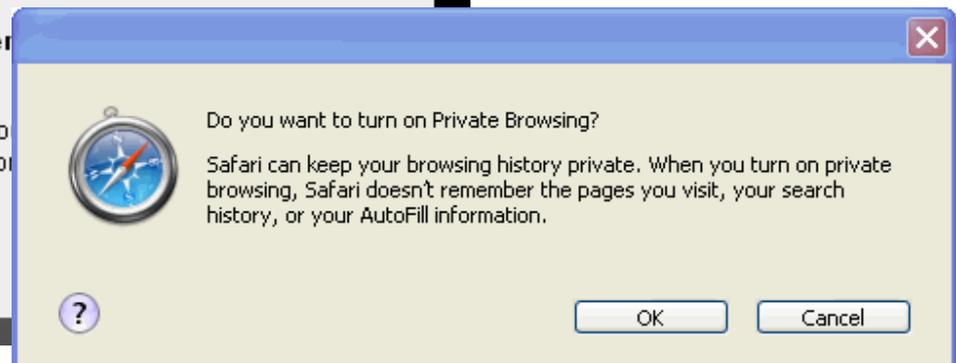


**You've gone incognito.** Pages you view in this window won't appear in your browser history or search history, and they won't leave other traces, like cookies, on your computer after you close the incognito window. Any files you download or bookmarks you create will be preserved, however.

**Going incognito doesn't affect the behavior of other people, services, or devices.**

- Websites that collect or share information about you
- Internet service providers or employers that track the pages you visit
- Malicious software that tracks your keystrokes in exchange for free services
- Surveillance by secret agents
- People standing behind you

[Learn more](#) about incognito browsing.



# Best Practices

- Manifest errors are fatal (case sensitive entries)
- If you are adding and removing (lots of files) files, remember to update the manifest file
  - Use a predeployment script
  - Use a version Comment in the manifest file
- Host your site on different domain names
  - You can do this on your local machine by hacking the hosts file (see example on the next slide)
  - Windows: `\WINDOWS\system32\drivers\etc\hosts`
  - UNIX: `/etc/hosts`
- To see if files are requested, watch the server log

# Checking Server Access

```
C:\WINDOWS\system32\cmd.exe - python -m SimpleHTTPServer 9999
portal.example.com -- [29/Sep/2010 19:58:47] "GET /image2.png HTTP/1.1" 200 -
portal.example.com -- [29/Sep/2010 20:15:01] "GET /offline.manifest HTTP/1.1" 200 -
portal.example.com -- [29/Sep/2010 20:15:01] "GET /index.html HTTP/1.1" 200 -
portal.example.com -- [29/Sep/2010 20:15:01] "GET /cache.html HTTP/1.1" 200 -
portal.example.com -- [29/Sep/2010 20:15:01] "GET /html5.css HTTP/1.1" 200 -
portal.example.com -- [29/Sep/2010 20:15:01] "GET /image1.jpg HTTP/1.1" 200 -
portal.example.com -- [29/Sep/2010 20:15:01] "GET /fallback.html HTTP/1.1" 200
-
portal.example.com -- [29/Sep/2010 20:15:01] "GET / HTTP/1.1" 200 -
portal.example.com -- [29/Sep/2010 20:15:01] "GET /offline.manifest HTTP/1.1" 200 -
portal.example.com -- [29/Sep/2010 20:16:41] "GET / HTTP/1.1" 200 -
portal.example.com -- [29/Sep/2010 20:16:41] "GET /html5.css HTTP/1.1" 200 -
portal.example.com -- [29/Sep/2010 20:16:41] "GET /offline.manifest HTTP/1.1" 200
portal.example.com -- [29/Sep/2010 20:16:41] "GET /index.html HTTP/1.1" 200 -
portal.example.com -- [29/Sep/2010 20:16:41] "GET /cache.html HTTP/1.1" 200 -
portal.example.com -- [29/Sep/2010 20:16:41] "GET /html5.css HTTP/1.1" 200 -
portal.example.com -- [29/Sep/2010 20:16:41] "GET /image1.jpg HTTP/1.1" 200 -
portal.example.com -- [29/Sep/2010 20:16:41] "GET /fallback.html HTTP/1.1" 200
-
portal.example.com -- [29/Sep/2010 20:16:41] "GET /offline.manifest HTTP/1.1" 200
portal.example.com -- [29/Sep/2010 20:16:41] "GET /offline.manifest HTTP/1.1" 200 -
```

# hosts File Hack for Local Testing

hosts file

```
# For example  
127.0.0.1 localhost  
127.0.0.1 offline0.example.com  
127.0.0.1 offline1.example.com
```

# Disk Quota

- Don't assume success and check for errors
  - Example in Chrome:  
`Application Cache Error event: Failed to commit new cache to storage, would exceed quota`
- In the future, browsers will hopefully have graceful quota upgrade mechanisms like Opera's for Local Storage



# Works Well With HTML5 Web Storage...

## JavaScript

```
if (navigator.onLine) {  
    //Send updates to server  
} else {  
    window.localStorage.myLocalKey = 'Some Data';  
}
```



# Cache-As-You-Go

- If you add the manifest attribute, files will be added to the cache implicitly
- Subresources and dependencies may not be loaded properly (CSS, JS, etc.)

## Manifest File

```
CACHE MANIFEST
```

```
FALLBACK:
```

```
/ /offline.html
```

```
NETWORK:
```

```
*
```

# Offline Emulation

- HTML5 Gears Project by Brad Neuberg, Google:  
<http://code.google.com/p/html5-gears/>





**QUESTIONS**

**ANSWERS**

# THANKS!

**E-mail:** [peter.lubbers@kaazing.com](mailto:peter.lubbers@kaazing.com)

**Twitter:** [@peterlubbers](https://twitter.com/peterlubbers)

**LinkedIn:** Peter Lubbers

**Apress HTML5 Book:** <http://prohtml5.com>

**Book Coupon:**

50% off Pro HTML5 e-book: **JFOKUSVTRU**



# Resources

- **WHATWG Offline Web Apps spec:**  
<http://www.whatwg.org/specs/web-apps/current-work/multipage/offline.html#offline>
- **W3C Offline Web Apps spec:**  
<http://dev.w3.org/html5/spec/offline.html#offline>
- **Offline example source files:** <http://bit.ly/9pJ1Zq>
- **Pro HTML5 Programming**, Offline chapter, Peter Lubbers, Brian Albers, and Frank Salim
- **Introducing HTML5**, Offline chapter, Bruce Lawson and Remy Sharp



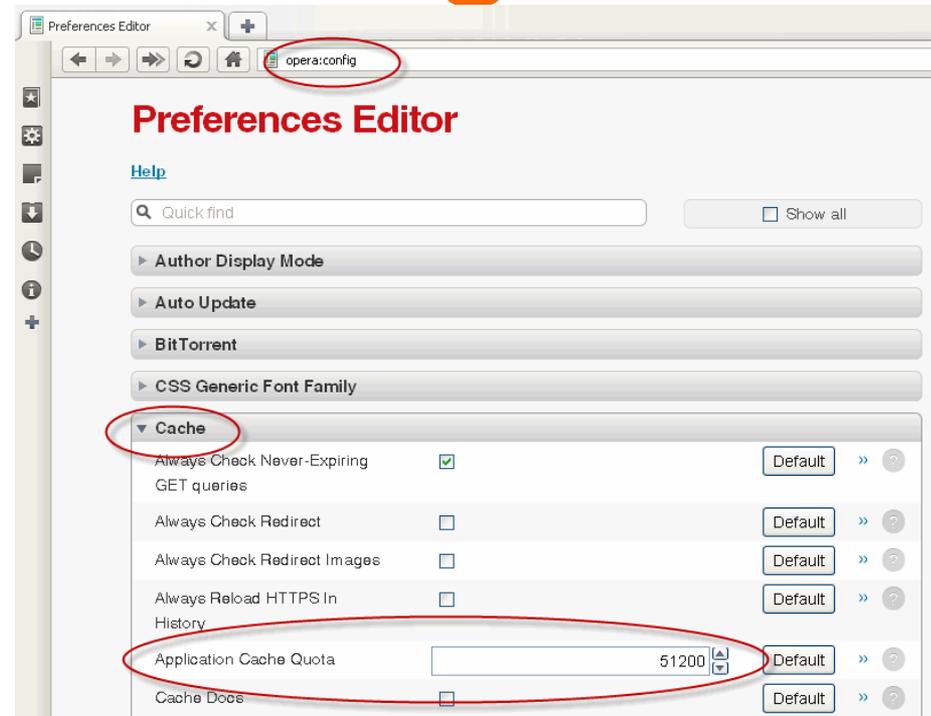
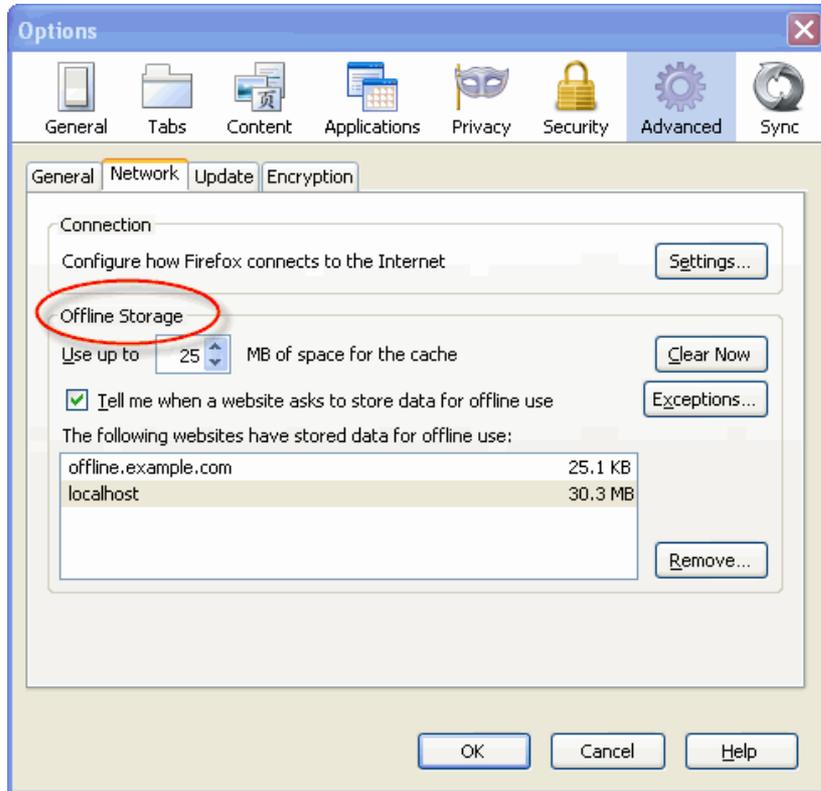
Copyright © 2010 Kaazing Corporation, All rights reserved.

All materials, including labs and other handouts are property of Kaazing Corporation. Except when expressly permitted by Kaazing Corporation, you may not copy, reproduce, publish, or display any part of this training material, in any form, or by any means.

# Appendix: Browser Offline Cache Settings and Clearing the Cache



# Offline Cache Settings



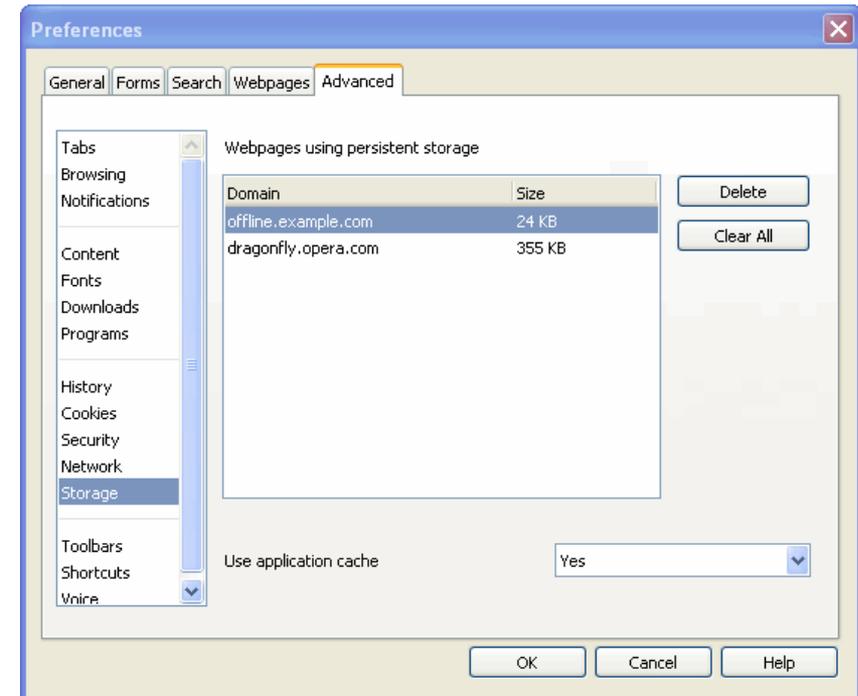
**Firefox: Tools > Options  
(Preferences on Mac OS X) >  
Advanced > Network**

**Opera: opera:config**

# Clearing the Cache

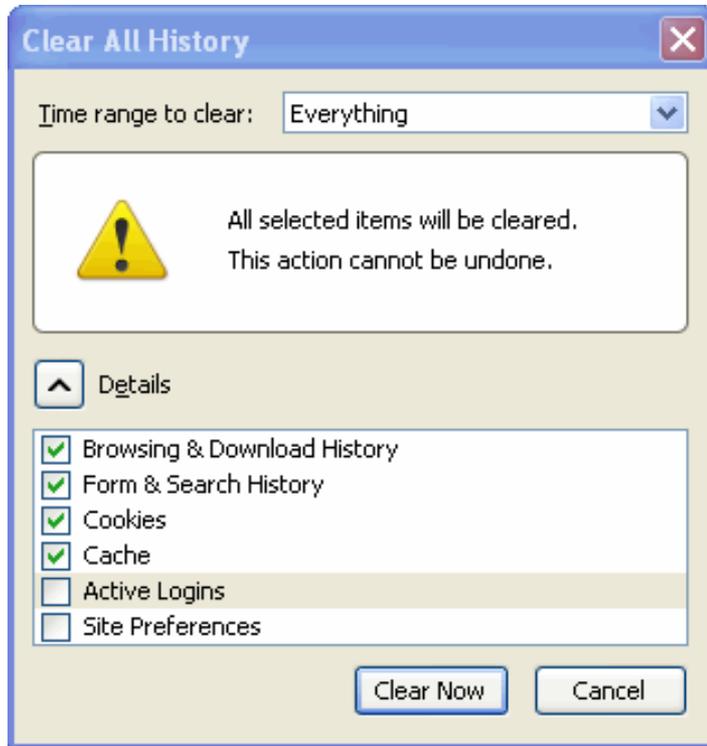


**Chrome:**  
**Settings Menu > Tools >**  
**Clear Browsing Data**

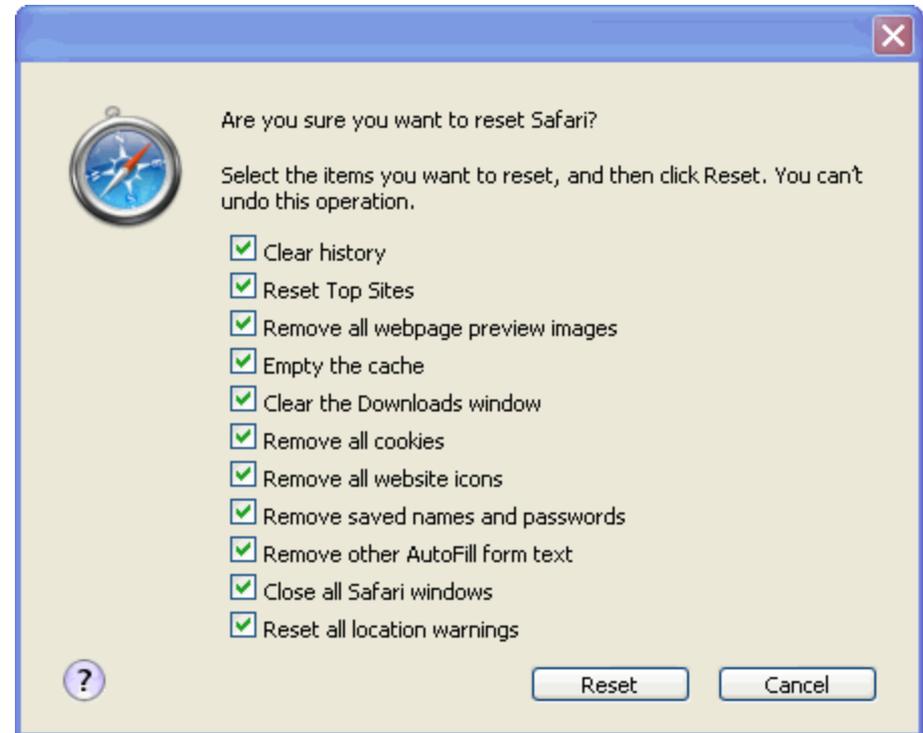


**Opera:**  
**Tools > Preferences > Storage**

# Clearing the Cache



**Firefox:**  
**Tools > Clear Recent History**



**Safari:**  
**Settings Menu > Reset Safari**

# Clearing the Cache

Browser	Steps to Clear the Cache
Chrome	Settings Menu > Tools > Clear Browsing Data
Firefox	Tools > Clear Recent History (and Tools > Options (Preferences on Mac OS X) > Advanced > Network > Remove for app cache)
Safari	Settings Menu > Reset Safari
Opera	Tools > Preferences > Storage (+ Tools > Clear Private Data)
Internet Explorer	Coming soon...

**Note:** Close any offline pages before you do this to avoid problems



TM

**KAAZING**