

iPhoneでの
ウェブパフォーマンス最適化を考える



@hiloki



[EC studio](#) > [EC studio デザインブログ](#) > [EC studio デザインブログの記事一覧](#)

2010年02月22日 投稿者: 谷 拓樹

知らない人は損してる？コーディングが3倍速くなるZen-Codingを導入してみた

最近になって急に盛り上がってきているZen-Codingをみなさんご存知ですか？

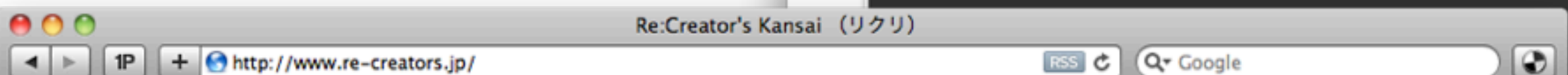
Zen-CodingはHTML/CSSをショートカット+スニペットで簡略化してコードを書くことができるライブラリです。

独立したテキストエディタソフトではなく、既存のHTML/CSSエディタにプラグイン的に導入するものです。

Zen-Coding

の動画を見て

※音が出ます

[Zen Coding](#)

Re:Creator's Kansai

Re:Creator's Kansai (リクリ) は関西のクリエイターの交流、勉強会・セミナーの開催を目的としたコミュニティです。

Re:Creator's Kansai

2010年9月24日 coyummy

● [11/20 \(土\) にUXのワークショップを開催します](#)

大変お久しぶりです。長い間潜伏してましたが、リクリ、一応生きてます。

11/20 (土) にユーザー エクスペリエンス (UX) のワークショップを開催することが決定しました！

講師は、[could](#)代表の長谷川恭久さん。

場所は、神戸ハーバーランドにある[兵庫県立大学神戸キャンパス](#)にて開催します。

詳細は決まり次第、当Blogにてお知らせしていきますので、
続報お待ちください！！

[セミナー](#)[ニュース](#)

Re:Creator's Kansai

[Re:Creator's Party Kansai \[mixi\]](#)

● [Re:Creator's Kansaiについて](#)

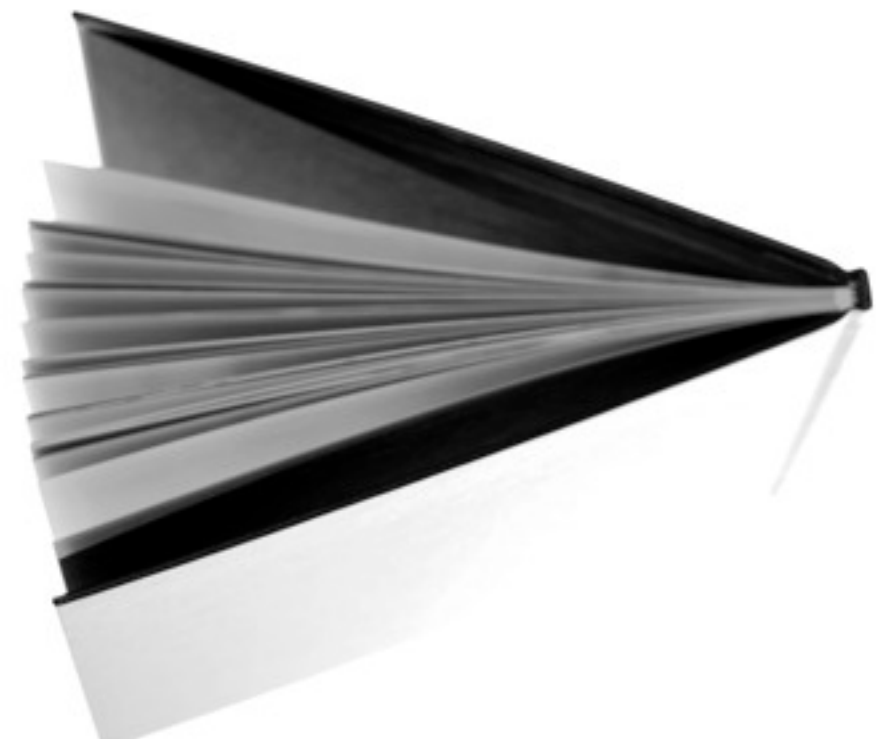
最近のブログエントリ

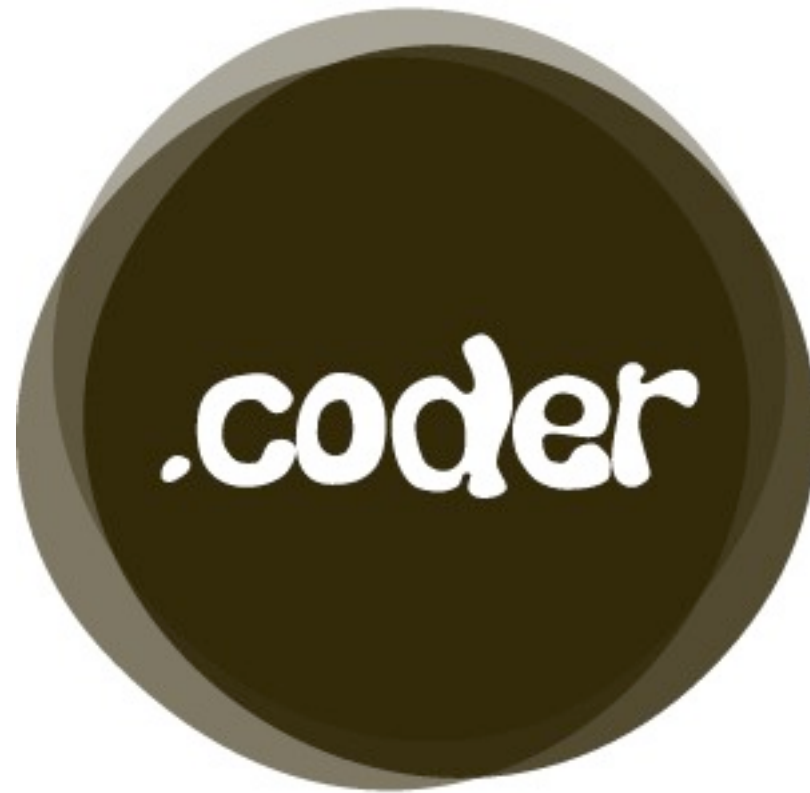
● [11/20 \(土\) にUXのワークショップを開催します](#)

● [\[イベント情報\] 4/12 \(水\) に京都にて「Dreamweaver Town」を開催します](#)

”HTML5とCSS3で魅せる iPhoneサイト”（仮）

っていう本出します。
（たぶん11月）





iPhoneでの
ウェブパフォーマンス最適化を考える

まとめ

デスクトップ向けのノウハウと
だいたい一緒。

本当はいろんな環境での検証テストをやりたかったんですよーデスクトップ向けのノウハウでもいまいち数字だけじゃわからないのも多いし、こりゃ実際に体感しないといけない&体感してもらうのが一番わかりやすいだろうとか思ったりして、特にそれがよくわかるのがモバイルなんじゃね？ってことで今回のスマートフォン（iPhone）にフォーカスしてやってみたら面白い結果が出るかもとおもって意気揚々と今回のセッションに名乗りでてみたわけなんだけどそしていざ日が近づいてきたらあれ全然時間ないやんっていうか検証がすげー難しいんじゃないのってことに気づいた。

iPhone (Mobile Safari) の キャッシュのお話


```
this.initMotion(el, attributes, duration, method);  
},  
YAHOO.util.Motion.prototype = new YAHOO.util.Anim();  
YAHOO.util.Motion.prototype.initMotion = function(el, attributes, duration, method) {  
    YAHOO.util.Anim.call(this, el, attributes, duration, method);  
};
```

YAHOO! USER INTERFACE BLOG

News and Articles about Designing and Developing with Yahoo! Libraries.

[Blog](#)[About](#)[YUI Jobs](#)[Contribute](#)

Mobile Browser Cache Limits, Revisited

July 12, 2010 at 8:45 am by Ryan Grove | In [Development](#), [Performance](#) | [4 Comments](#)

In [Mobile Browser Cache Limits: Android, iOS, and webOS](#), I shared the results of my attempts to determine browser cache limits on Android, iOS, and webOS devices. At the end of the article, I wrote:

Use these results as a starting point, but verify them yourself before you make major decisions based on assumptions about mobile cache limitations. The mobile browser world changes at a lightning pace, so this research will have a very short shelf life.

As it turns out, that was good advice: the day after the article was posted, [Steve Souders](#) commented that he had run tests using a different methodology that was more representative of a real-world web workflow and had gotten different results.

New Methodology

My original methodology involved navigating directly to a randomly generated page of a certain size, served via a proxy, and then checking the cache. The results of these tests were reliably reproduced. However, I had assumed that the limits for direct navigation to an HTML resource were the same as the limits for external CSS and JavaScript. My assumption that the limits for direct navigation to an HTML resource were the same as the limits for external CSS and JavaScript was incorrect, so even though the results of my tests were valid, they weren't widely applicable.

YUI LIBRARY ON TWITTER

[YUILibrary Twitter feed](#)

[YUI Devs on Twitter](#)

CROCKFORD ON JAVASCRIPT VIDEO SERIES



TRANSLATE



<http://www.yuiblog.com/blog/2010/07/12/mobile-browser-cache-limits-revisited/>

All Posts

[+ MY YAHOO!](#) [RSS](#)



WebOS Goodies へようこそ！ WebOS はインターネットの未来形。あらゆる Web サイトが繋がり、共有し、協力して創り上げる、ひとつの巨大な情報システムです。そこでは、あらゆる情報がネットワーク上に蓄積され、我々はいつでも、どこからでも、多彩なデバイスを使ってそれらにアクセスできます。WebOS Goodies は、さまざまな情報提供やツール開発を通して、そんな世界の実現に少しでも貢献するべく活動していきます。

Subscribe



続・iPhone, Android, webOS モバイルブラウザキャッシュの制限

Tweet This

いいね！

1人が「いいね！」と言っています。

以前、[iPhone, Android, webOS モバイルブラウザキャッシュの制限](#)という記事で、iPhoneをはじめとするモバイルブラウザのキャッシュに関する調査結果を公開した Ryan Grove 氏の記事をご紹介しました。その後いろいろと多忙になってフォローできずにいたのですが、実は情報元の YUI Blog では追加情報を掲載した[補足記事](#)が公開されています。そこで、本日はその内容をご紹介します。

前回同様、全訳する時間はないので、要点のみをまとめました。前回の記事では、とくに iOS 3.x についてはキャッシュの制限がかなり厳しいという結果が出ていましたが、「実はそうでもなかった」というのがわかっています。モバイルサイトをより閲覧しやすいものにするために、今回もぜひご覧ください。

前回のデータの計測はランダムに生成した HTML ページ (Content-Type は text/html) にダイレクトにナビゲートするという方法で行われていました。つまり、ユーザーが直接アクセスする HTML ページに対するキャッシュのみを計測し、間接的に読み込まれる CSS や JavaScript についてはノータッチだったのです。High Performance Web Sites の著者として有名な Steve Souders 氏がそれを指摘し、Ryan 氏と協力してテストを改善しました。新しいテストでは、ランダムに生成した CSS や JavaScript を読み込む HTML を表示し、さらに同じ CSS / JavaScript ファイルを読み込む別の HTML に遷移してキャッシュが働いているかどうかを確認しています。その結果、間接的に読み込まれる CSS や JavaScript のキャッシュも有効化されていることが確認されています (元ネタのサイトのスクリーンショット参照)。

http://webos-goodies.jp/archives/mobile_browser_cache_limits_revisited.html

Browser/OS/Device	Single Component Limit	Survives Power Cycle	Categories
Android 2.2 (Nexus One)	2MB	Yes	一歩進んだ Web アプリケーション活用術

Browser/OS/ Device	Single Component Limit	Page Cache Size Limit	Survives Power Cycle
Mobile Safari, iOS 3.1.3 (1st-gen iPhone)	4MB+※1	∞ ※2	No
Mobile Safari, iOS 3.2 (iPad)	4MB+※1	25.6KB (26,214b)	No
Mobile Safari, iOS 4.0 (iPhone 3GS)	4MB+※1	∞ ※2	No
Mobile Safari, iOS 4.0 (iPhone 4)	4MB+※1	∞ ※2	No
Android 2.2 (Nexus One)	2MB	∞ ※2	Yes

※1 キャッシュされるサイズ（ファイル単体の制限は10MB）

※2 RAM容量の制限のみ？

✓ 遠い未来を指す expiration ヘッダを使用する

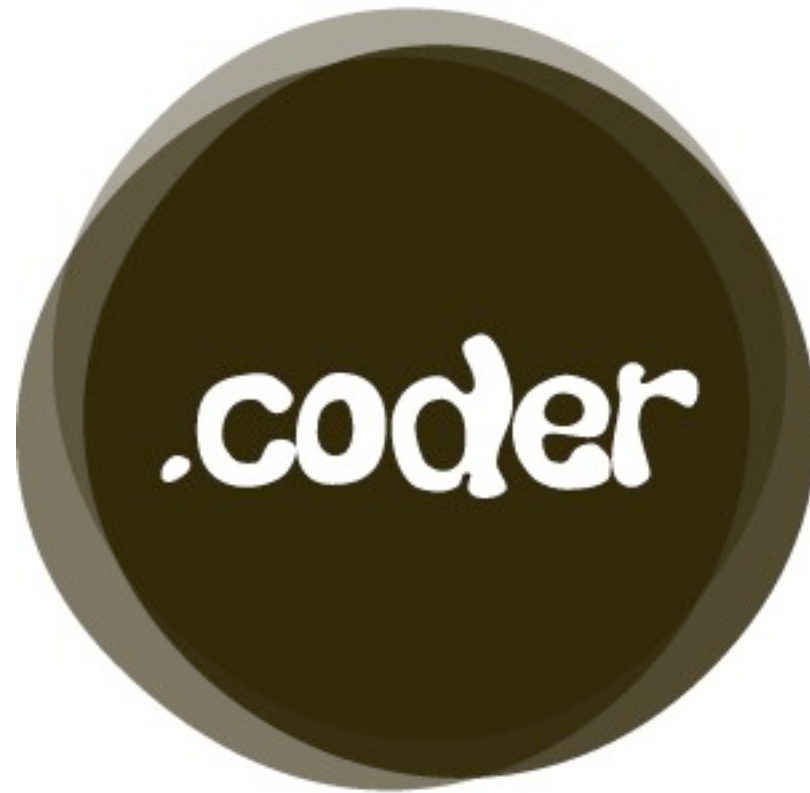
✓ HTML ページの容量を 25.6KB 以下にする

✓ CSS ,JS コンポーネント_{※3}を 2MB 以下にする

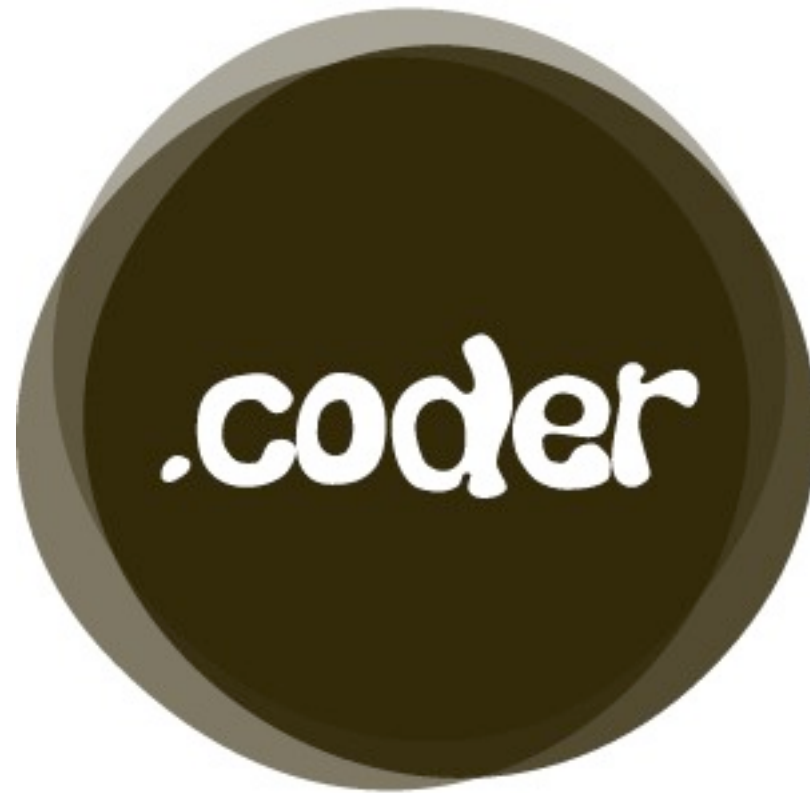
✓ HTML5 アプリケーションキャッシュの利用を検討する

✓ 自分自身でテストする

※3 imageは対象外のようにです（すくなくとも2MB上の画像でも読み込める）



iPhoneでの
ウェブパフォーマンス最適化を考える



iPhoneでの
ウェブパフォーマンス最適化を考える

ここからデモタイム

http://bit.ly/dc4_iphone

- ✓ 画像（PNG,GIF,TIFF）は2メガピクセル以下に抑える
- ✓ SVGはサイズ次第で使える（制限の詳細は不明）
- ✓ data URI scheme は 25.6KB /HTML、 2MB / CSS , JS
以内であれば使える http://bit.ly/dc4_iphone
- ✓ アニメーションはCSS3 を使う
- ✓ CSS3でデザインをする

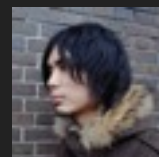
ほんとの まとめ

- ✓ WPOの施策方針はデスクトップと一緒に
- ✓ ただしデスクトップ以上にファイルサイズに気を使う

http://bit.ly/dc4_safari_webcontent_guideline

ご声援ありがとうございました

※次回作にご期待ください



<http://twitter.com/hiloki>