

試しに

ginpei

a { color: red !important }

## [jQuery 1.8の更新内容をまとめたよ。](#)

85

投稿日時: [2012/08/14 04:54](#)

カテゴリー: [JavaScript](#), [Web](#)

タグ: [jQuery](#), [jQuery 1.8](#), [jQuery更新まとめ](#)

個人的に油断してたんですが、jQuery 1.8がリリースされましたね。

CSSのベンダープレフィックスを自動でほぼ補完してくれるようになったのが目玉機能でしょうか。作業量としては、セレクターエンジンSizzleとアニメーション関係を全面的に書き換えた事が大きそうです。なお諸々強化されているにも関わらず、ファイルサイズは減っています。

(ちょっとだけだね。) えらい！

- [jQuery Blog » jQuery 1.8 Released](#)
- [jQuery Blog » jQuery 1.8 Beta 1: See What's Coming \(and Going!\)](#)
- [jQuery Blog » The New Sizzle](#)
- [jQuery Blog » jQuery Core: Version 1.9 and Beyond](#)
- [jQuery Blog » jQuery 1.9 and 2.0 — TL;DR Edition](#)

なおjQuery UI (1.8.22)とjQuery Mobile (1.1.1)が正常に動作する事は確認しているようです。

### ダウンロード

- <http://code.jquery.com/jquery-1.8.0.min.js>
- <http://code.jquery.com/jquery-1.8.0.js>

ちなみにWordPressで利用するにはこちらの記事を。

- [WordPressでjQueryの最新版や任意のバージョンを使うには。](#)

### 主な変更点

#### CSSのベンダープレフィックスを自動付加

いままで `$elem.css('-webkit-transform', '...')` って書いてたのが

`$elem.css('transform', '...')` で済むようになりました。いやー、これ良いですね。ただし罣がありますのでご注意。

[→詳しく](#)

## モジュール化

[jQuery UI](#) のイメージですね。jQuery はたくさんの機能がありますが、必要なものだけを選んでカスタムした `jquery.js` を作成できるようになりました。

[→詳しく](#)

## アニメーションの処理を刷新

複雑になってきて管理が大変になって来た事と、何やら壮大な計画があるそうで、まるっきり書き換えたとの事。

使う側としては、基本的には今までどおりです。

[→詳しく](#)

## セレクター解析エンジン **Sizzle** を再構築

一から書き直したようです。（しかも二度！） これで過去のものよりも高速になったとか。どこかに比較とかあるのかな？

ただし高速化の影響があるのは複雑なセレクターです。単純なセレクター（ID やクラス、要素名だけのもの）については今回は高速化されていません。元々ほぼ限界まで高速化されていたから。

## XSS 対策

強化されたようです。あんまりよくわからんのですが、`$.parseHTML()` が用意されています。今後はこれを使いという方針のようです。

[ソースを表示](#)

[印刷SyntaxHighlighterについて](#)

```
1 $('<script>alert()</script>'); // => [<script>alert()</script>]
2 $.parseHTML('<script>alert()</script>'); // => []
3 $.parseHTML('<script>alert()</script>', null, true); // =>
   [<script>alert()</script>]
```

また今後 (1.9) ではさらに対策を強化する、とのアナウンスもあります。`$(html)` で生成できるのはごく単純なもの（例えば `$('<p />')` のようなもの）に限られ、また一文字目が `<` でないと駄目になるそうです。

まあ普通はそういう使い方だけ出来れば良いんじゃないかなと思います。属性付きのHTML要素を生成する場合は、`attr()` を使うか、`$(html, props)` という書き方ができます。

## [ソースを表示](#)

## [印刷SyntaxHighlighterについて](#)

```
1 var $elem = $('<element />', { attr1: val1, attr2: val2 });
2 // => [<element attr1="val1" attr2="val2" />]
```

## 削除

詳細	API	備考
<a href="#">#11787</a>	<code>\$.curCSS</code>	
<a href="#">#10589</a>	<code>\$elem.data("events")</code>	デバッグ用に <code>\$.data(\$elem, "events")</code> が利用可能。 。
<a href="#">#11738</a>	<code>\$elem.closest(Array)</code>	<code>.closest(selector)</code> は引き続き利用可能。
<a href="#">#11736</a>	<code>deferred.isResolved()</code>	<a href="#">deferred.stat()</a> で代用。
<a href="#">#11736</a>	<code>deferred.isRejected()</code>	<a href="#">deferred.stat()</a> で代用。

実は `$elem.data("events")` が何だかわかってません。

"\$elem.data("events")はjQueryが内部用に持ってる、要素ごとに設定されたイベントハンドラですね"との事です。

- [Twitter / kyo\\_ago: \\$elem.data\("events"\)はjQuer...](#)

## 削除アナウンス

将来 (1.9とか) 削除されるらしい。いつ消えるかは個別に調べてください。

詳細	API	備考
<a href="#">#9385</a>	<code>\$.browser</code>	<code>\$.support</code> で機能別に処理を。
<a href="#">blog</a>	<code>\$.attrFn</code>	
<a href="#">blog</a>	<code>\$.sub()</code>	
<a href="#">blog</a>	グローバルAjaxイベント	
<a href="#">#11721</a>	<code>\$.support.boxModel</code>	
<a href="#">#11013</a>	<code>\$.ajax()</code> で <code>{async:false}</code> とDeferredの併用	同期通信では必ずコールバック関数を利用する。
<a href="#">#10657</a>	<code>\$elem.size()</code>	<code>\$elem.length</code> で代用。
<a href="#">#11733</a>	<code>\$elem.load()</code>	<code>\$elem.on('load')</code> で代用。

<a href="#">#11733</a>	<code>\$elem.unload()</code>	<code>\$elem.on('unload')</code> で代用。
<a href="#">#11733</a>	<code>\$elem.error()</code>	<code>\$elem.on('error')</code> で代用。
<a href="#">#9800</a>	<code>\$elem.andSelf()</code>	<code>\$elem.addBack()</code> へ移行。
<a href="#">#11719</a>	<code>\$elem.bind('ready', fn)</code>	<code>\$(fn)</code> ないし <code>\$.ready(fn)</code> で代用。(※1)
<a href="#">#11786</a>	<code><a href="#">\$elem.toggle(handler1, handler2)</a></code>	<code><a href="#">\$elem.toggle([duration] [, callback])</a></code> は引き続き利用可能。(※2)
<a href="#">#9400</a>	フォーム項目疑似クラス ( <code>:text</code> や <code>:checkbox</code> など)	<code>[type=text]</code> 等で代用。
<a href="#">#11731</a>	hover イベント	
<a href="#">#11718</a>	<code>\$elem.data()</code> で発火するイベント	
<a href="#">#9399</a>	<code>\$.ajax()</code> が返す Promise の <code>.success()</code>	Promise の <code>.done()</code> へ移行。
<a href="#">#9399</a>	<code>\$.ajax()</code> が返す Promise の <code>.error()</code>	Promise の <code>.fail()</code> へ移行。
<a href="#">#9399</a>	<code>\$.ajax()</code> が返す Promise の <code>.complete()</code>	Promise の <code>.always()</code> へ移行。

- ※1 ... 厳密に言う则ちよつと違ひます。ready 発火前なら同じですが、ready 発火後に `.on('ready', fn)` しても `fn` は実行されません。 `$.ready(fn)` であれば、発火後に実行した場合 `fn` は即時実行されます。なお `ready` はjQuery独自のイベントです。
- ※2 ... `.toggle()` には1.0の頃から「`.show()` と `.hide()` を交互に実行する」という機能と、「クリックされるたびに二種類のコールバック関数を交互に実行する」という機能がありました。名前がかぶってるといふわけですね。紛らわしいので後者が削除になります。

いやはや、`$.browser` は前々から非推奨とされてましたが、いよいよ削除です。これも含め、だいたい互換プラグインへの移行が計画されている様子です。

## ファイルサイズ

ファイル名    ファイルサイズ

jquery-1.8.0.min.js 92,556 Bytes

jquery-1.7.2.min.js 94,840 Bytes

わずかですが、小さくなりました。減ったといふか、機能を追加しているのに増えなかった、という観点で見ると嬉しいかなと思います。

さて、では個別に見て行きましょうか。

## ベンダープレフィックス自動補完

CSSのプロパティを操作する際、名前は自動的にベンダープレフィックスが補完されるようになりました。例えば `$elem.css('transform', '...')` と書くと、自動的に `"-moz-transform"` なり `-webkit-transform"` なり、適切なものが利用されます。

ベンダープレフィックスの必要・不要を見分けるのにブラウザが持っているスタイル情報を利用しているので、常に最新のブラウザに対応できます。

### 値はベンダープレフィックスが補完されない

ただしひとつ注意点がありまして、プロパティ名の方には自動で付加してくれますが、値の方はそうはしてくれないみたいです。例えば `transition` で注意が必要です。

[ソースを表示](#)

[印刷SyntaxHighlighterについて](#)

```
1 // Webkitのベンダープレフィックスの場合
```

```
2 $elem.css('-webkit-transition', '-webkit-transform 1s'); // OK
```

```
3 $elem.css('transition', '-webkit-transform 1s'); // OK
```

```
4 $elem.css('transition', 'transform 1s'); // NG
```

`"transform"` ではなくベンダープレフィックスを付けて、`"-webkit-transform"` のようにしてやる必要があります。

## モジュール化

ただし[jQuery UIのビルドツール](#)のような、オンラインでカスタムしてダウンロード、というツールは（まだ？）ないようです。作るか。

以下のモジュールに分割されています。

- ajax
- css
- dimensions
- effects
- offset

なお今後もフル版を提供し続けるので、カスタマイズはやんなくていいよ安心してね、との事です。

## ビルドツールはgrunt

カスタムビルドにはgruntを利用します。

- [cowboy/grunt](#)
- [gruntで快適JS/CSSビルド生活 – Takazudo hamalog](#) (※古いです。)

jQuery 1.8のビルド方法はこちらに。

- [jquery/README.md at master · jquery/jquery](#)

このgruntの導入と利用は（UNIXに不慣れな方には）だいぶ大変かと思います。やっぱりオンラインビルダーが必要ですね。

## アニメーション

中身がこんがらがってきたので整理して、Deferredをより有効に使うようになったようです。高度なアニメーション処理をより円滑に行うために、`$.Animation` というのが用意されました。専用のプロパティを持ったPromiseを返します。（Promiseってのは[Deferredのアレ](#)です。）

### \$.Animation の簡単な使い方

まだドキュメントに掲載されていないようなので、試してみてさっくり使い方を示します。

[ソースを表示](#)

[印刷SyntaxHighlighterについて](#)

```
1 var anim = $.Animation($elem[0], { left: 100 }, { duration: 1000 });
```

注意点が二つあって、まず第一引数がjQueryオブジェクトではない事に注意してください。生のHTML要素オブジェクトです。

もうひとつは引数を省略できないということ。三つとも指定しないと `TypeError` になっちゃいます。全体的にエラー訂正は省略されている印象です。速度を求めた結果なのだと思います。

```
TypeError: Cannot read property 'duration' of undefined
```

戻り値 `anim` はPromiseで、`.progress()` や `.done()` といったメソッドを用いてコールバック関数を与える事ができます。

### \$.Animation は各種のアニメーションに利用できる

また第一引数は、実は任意のオブジェクトを与える事ができます。より抽象的な使い方を示します。

[ソースを表示](#)

[印刷SyntaxHighlighterについて](#)

```
1 var from = 0, to = 100;
2 var elem      = { value: from };
3 var properties = { value: to };
4 var options   = { duration: 1000 };
5 var anim = $.Animation(elem, properties, options);
6
7 // animはPromise
8 anim
9     .progress(function() { console.log('progress', arguments); })
10    .done(function() { console.log('done', arguments); });
```

こっちの使い方の方が \$.Animation を使う意義がありそう。実際、jQuery Blogでは[Canvasを操作する例](#)を挙げています。

このとき注意が必要なのが、 properties に与えたプロパティは、必ず elem にも存在しなければならないという事です。存在しない場合、 TypeError になります。

```
TypeError: Cannot use 'in' operator to search for 'xxx' in undefined
```

## jQueryの今後の展開

### 予定

#### jQuery 1.9 (2013前半)

1.8で「反対」とされた多数のインターフェイスを削除。一部については互換プラグインを提供。IE 6/7/8も引き続きサポート。

#### jQuery 1.9.x (2013年末)

引き続き不具合修正や新ブラウザへの対応など。

#### jQuery 2.0 (2013前半、1.9が出てすぐ)

このバージョンでは1.9と同じAPIを提供。ただしIE 6/7/8のサポートを打ち切り、再設計して軽量化、高速化。

### 古いIEのサポート

継続されます。

1.9と2.0は（古いIEのサポート以外は）同じ機能を提供するようになるみたいです。不安なら全てカバーした1.9を使えば良いし、不要なら軽量の2.0を使えばよろしい。

またHTMLをこんな風にする事で、適切なバージョンのjQueryを読み込む事ができます。

[ソースを表示](#)

[印刷SyntaxHighlighterについて](#)

```
1 <!-- [if lt IE 9] >
2     <script src="jquery-1.9.0.js"></script>
3 <![endif] -->
4 <!-- [if gte IE 9] ><!-->
5     <script src="jquery-2.0.0.js"></script>
6 <!--<![endif] -->
```

将来 (2.1) では古いIEのサポートはどうなるか

「（占いの）水晶玉でも持ってきやがれ」との事です。

ちなみにWindows XPのサポートは2014年の四月まで。そんなに気にしなくて良いんじゃないかな。

## おしまい

間違ってるところあればコメント欄で教えてくださいまし。

gruntの使い方と、あと（.querySelectorAll()があるのに）Sizzleが何故必要なのか、てな事を近いうちに記事できたらなー思っています。乞うご期待。（夏休みで出かけちゃうけどね。）

とっぴんぱらりのぷう。

## 追記

削除、削除アナウンスの情報を更新 (2012-08-18)

- 詳細ページ（主にチケット）へのリンクを追加。
- 代用等の備考を追加。 .toggle() とか紛らわしいしね。
- 削除アナウンスに、 \$.ajax() の戻り値Promiseの .success(), .error(), .complete() を追加。
- 削除アナウンスの、「\$.ajax() の同期オプション { async: false }」を「\$.ajax() が同期{ async: false } のときのDeferred併用」に修正。

## “jQuery 1.8の更新内容をまとめたよ。”への6件のフィードバック

1. [jQuery 1.8の更新内容をまとめたよ。 | Ginpen.com | ちゅどん道中記](#) より:  
[2012年8月14日 10:36 AM](#)

[...] 1.8の更新内容をまとめたよ。 | Ginpen.com Posted 2012年8月14日 Check jQuery 1.8の更新内容をまとめたよ。 | Ginpen.com個人的に油断してたんですが、jQuery 1.8がリリースされましたね。 [...]

[返信](#)

2. [8月14日の注目記事 | Javable.Jp](#) より:  
[2012年8月15日 12:32 AM](#)

[...] jQuery 1.8の更新内容をまとめたよ。 | Ginpen.com [...]

[返信](#)

3. [#TODO | CYY](#) より:  
[2012年8月16日 4:12 AM](#)

[...] <http://ginpen.com/2012/08/14/jquery-1-8/> This entry was posted in Uncategorized. Bookmark the permalink. ← Hadoop [...]

[返信](#)

4.  Syn より:

[2012年8月16日 11:02 AM](#)

> 削除アナウンス

> \$.ajax の同期オプション {async:false}

As of jQuery 1.8, the use of `async: false` with `jqXHR ($.Deferred)` is deprecated; you must use the `complete/success/error` callbacks.

とあるので、`async: false` 時に `jqXHR` 形式で値を返すのをやめる (Deferred 形式のメソッドチェーンができなくなる) 予定なだけで、コールバック形式で処理するなら大丈夫ではないかと思うのですが...  
読み違いでしたらすみません。

あと、

Deprecation Notice:

The `jqXHR.success()`, `jqXHR.error()`, and `jqXHR.complete()` callbacks will be deprecated in jQuery 1.8.

To prepare your code for their eventual removal, use `jqXHR.done()`, `jqXHR.fail()`, and `jqXHR.always()` instead.

ということで、非同期の場合でも `success()`, `error()`, `complete()` は削除予定のようですね。

[返信](#)

○ *ginpei* より:

[2012年8月18日 10:37 PM](#)

おお、ありがとうございます！ チケットの議論をすっ飛ばして見ていたので、subjectが変更になっていたのを見落としていました。よくよく見てみるとコード付きでコメント付いてましたね。

[返信](#)

5. [jQuery1.8.0リリースされました。 : ホームページ制作会社 アイデアシーディー スタッフブログ](#) より:

[2012年8月28日 11:04 AM](#)

[...] 参考URL : 横浜ペンギンWebに行く <http://ginpen.com/2012/08/14/jquery-1-8/> [...]

[返信](#)

## コメントを残す

名前 (必須)

メールアドレス (非公開) (必須)

ウェブサイト

[← WordPressでjQueryの最新版や任意のバージョンを使うには。](#)