

Webプログラミング言語 (JavaScript処理系)

サイボウズ・ラボ株式会社

竹迫 良範

<takesako@shibuya.pl>

HTTP Meeting 2008 ~Webサービスの明日を見つけよう~ @ InternetWeek 2008

自己紹介

■ 氏名

- 竹迫 良範
- id:TAKESAKO
- (0x1F歳)



■ 所属

- サイボウズ・ラボ株式会社
- Microsoft MVPアワード2008 - Developer Security
- セキュリティ&プログラミングキャンプ2008講師
- 第29回 U-20プログラミングコンテスト審査委員
- Shibuya Perl Mongers 2代目リーダー
 - オライリー Perlクックブック第2版 監訳 など

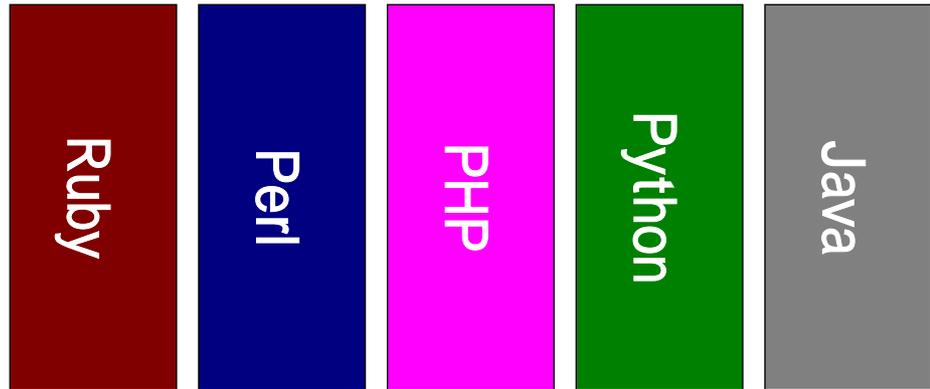
HTTP Meeting 2008 ~Webサービスの明日を見つけよう~ @ InternetWeek 2008

Webアプリケーションの開発環境

クライアント側
(ブラウザ)

(いま) JavaScript 必須

サーバ側
(Webサーバ)



HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008

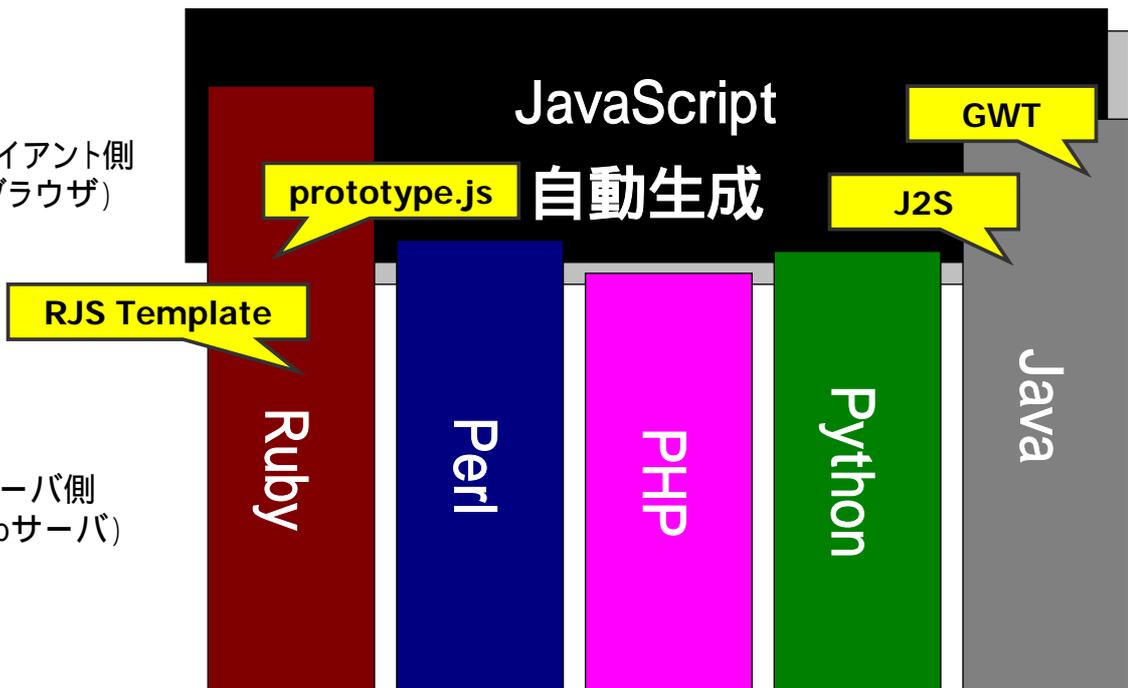
Web2.0・Ajaxブーム到来でカオスな状況に

クライアント側
(ブラウザ)

JavaScript

自動生成

サーバ側
(Webサーバ)



HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008

本当のWebアプリケーション開発言語は？

クライアント側
(ブラウザ)

言語大統一理論

サーバ側
(Webサーバ)

サーバ側も
JavaScript で
全部書く

HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008

コンピュータの進化とプログラミング言語

■ CPUのトレンド

■ マルチコア化

- それを生かすプログラミング言語処理系の研究

■ 高速化技術

- アウト・オブ・オーダー、投機的実行 ...
- スーパーパイプライン、スーパースカラー ...

■ Lightweight Language の登場

■ スクリプト言語でも高速に動作

- Perl, Ruby, Python, PHP, ActionScript ...

■ JavaScript 高速化バトル勃発

■ Chrome の v8 エンジンの公開(2008年)

HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008

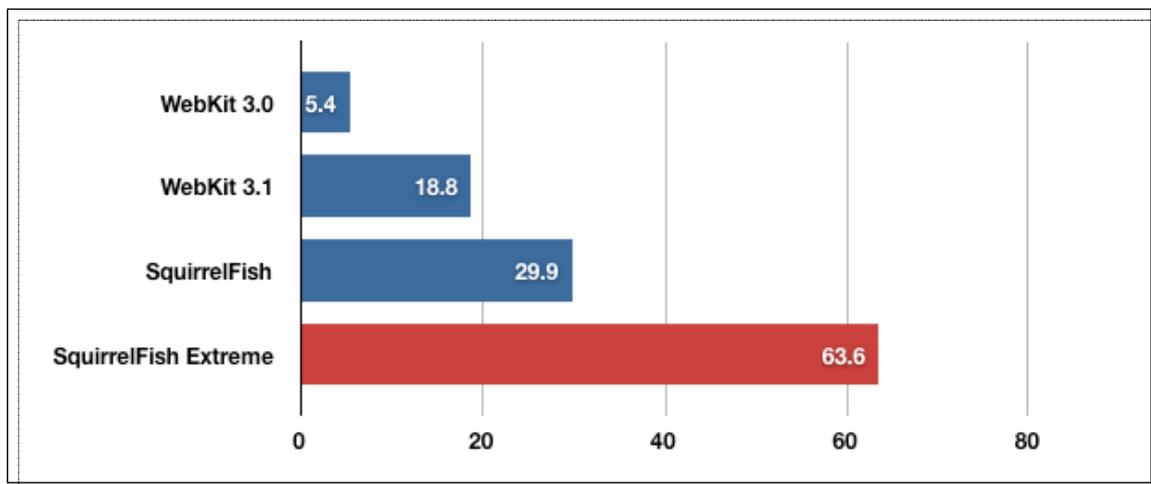
各ブラウザの JavaScript エンジン

- Internet Explorer
 - JScript
- Chrome
 - v8
- Firefox
 - SpiderMonkey
 - TraceMonkey
 - Tamarin
- Safari
 - WebKit
 - SquirrelFish
 - SFX (SquirrelFish Extreme)

HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008

JavaScript エンジンの速度比較

■ SunSpider benchmark の結果 (Safari)

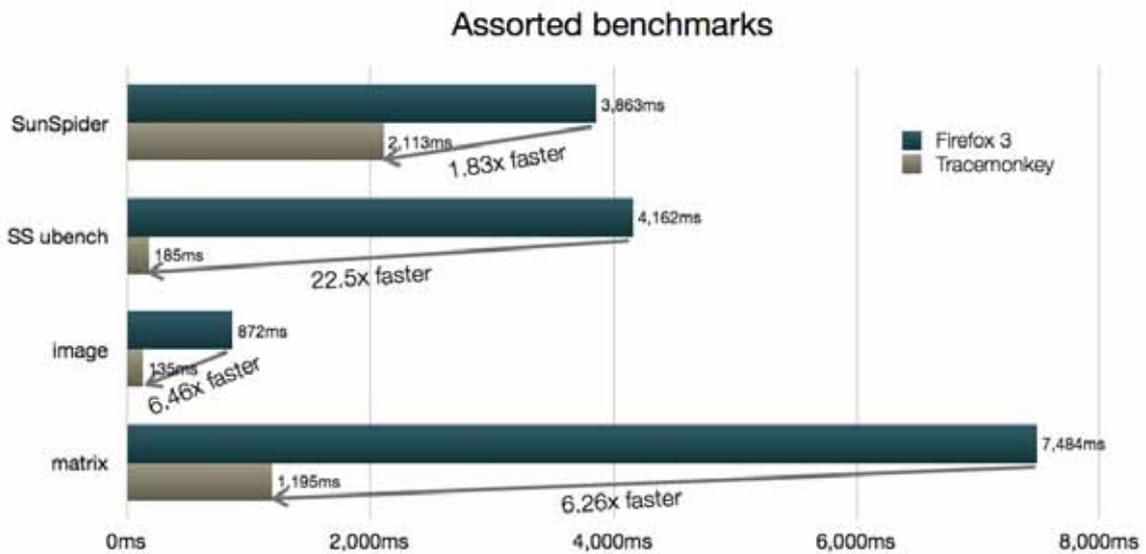


<http://trac.webkit.org/wiki/Introducing%20SquirrelFish%20Extreme.ja>

HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008

TraceMonkey: JavaScript Lightspeed

Brendan's Roadmap Updates

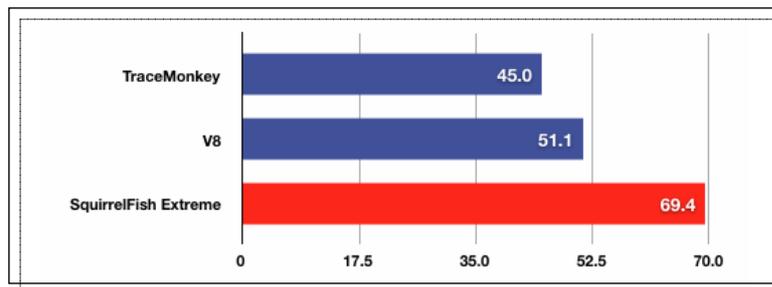


http://weblogs.mozillazine.org/roadmap/archives/2008/08/tracemonkey_javascript_lightsp.html

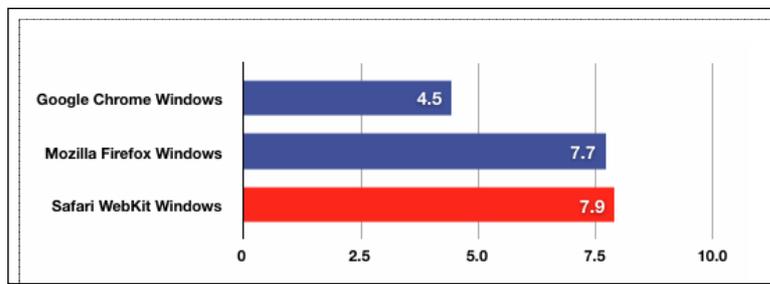
HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008

Safari vs. Firefox vs. Chrome

1. SunSpider benchmark (Mac OS X)



2. Mozilla's Dromaeo benchmark (Windows)



<http://www.satine.org/archives/2008/09/19/squirrelfish-extreme-fastest-javascript-engine-yet/>

HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008

量質転化の法則

- 量が質に転化するとき
 - 量を積み重ねることによって質的な変化が起こる瞬間
 - 頭の良くない奴(プログラム)でも、量をこなせば質に転化することがある
- コンピュータの処理速度が向上
 - 賢い枝狩りアルゴリズム
 - 昔はCPUの速度が遅く、メモリの量も少なかった
 - 単純な総当りアルゴリズム
 - 単位時間あたりに処理できる情報量が増えたため現在再注目されている

HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008

JavaScript = Web2.0時代のマシン語



HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008

Z80 エミュレーター

■ jsMSX

- The first MSX emulator 100% written in Javascript

jsMSX Copyright (c) 2006 Marcus Granado [mrc.gran@gmail.com]

jsMSX 0.9.2 on [canvas]:



```
Console:
Booting jsMSX
Starting RAM slots
Starting VDP
Starting PSG (No Sound)
interrupt=0,ticks=59666 cpu ticks/interrupt,
cpu clock=3.58 MHz
MSX ready to go. Load ROMs and hit [start].
```

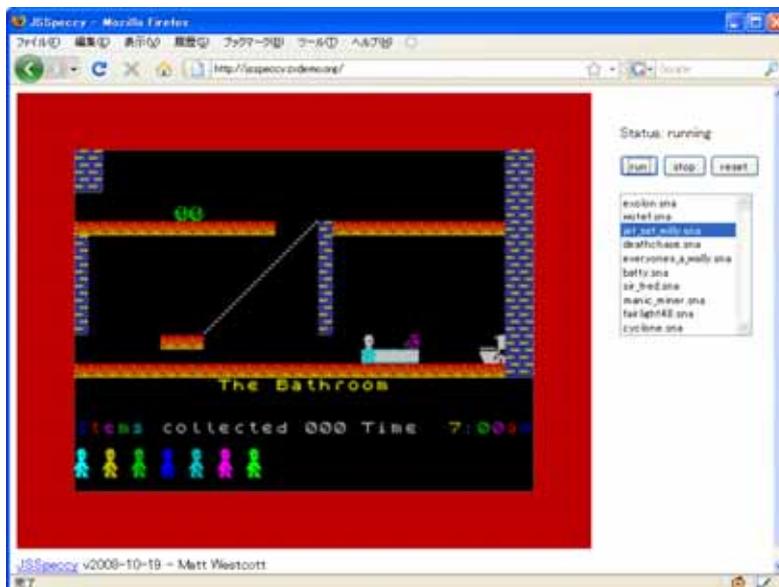
Just wait, it may be slow in your browser...
Actual speed should be around 60 interrupts/sec.

<http://jmsxdemo.googlepages.com/jmsx.html>

HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008

JSSpeccy

■ ZX Spectrum emulator in Javascript



ZX Spectrum
(3.50MHz Z80 / RAM 16KB)



<http://jsspeccy.zxdemo.org/>

<http://matt.west.co.tt/spectrum/jsspeccy/>

HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008

SproutCode

- RIA Webアプリケーション開発向け
- 100% JavaScript フレームワーク



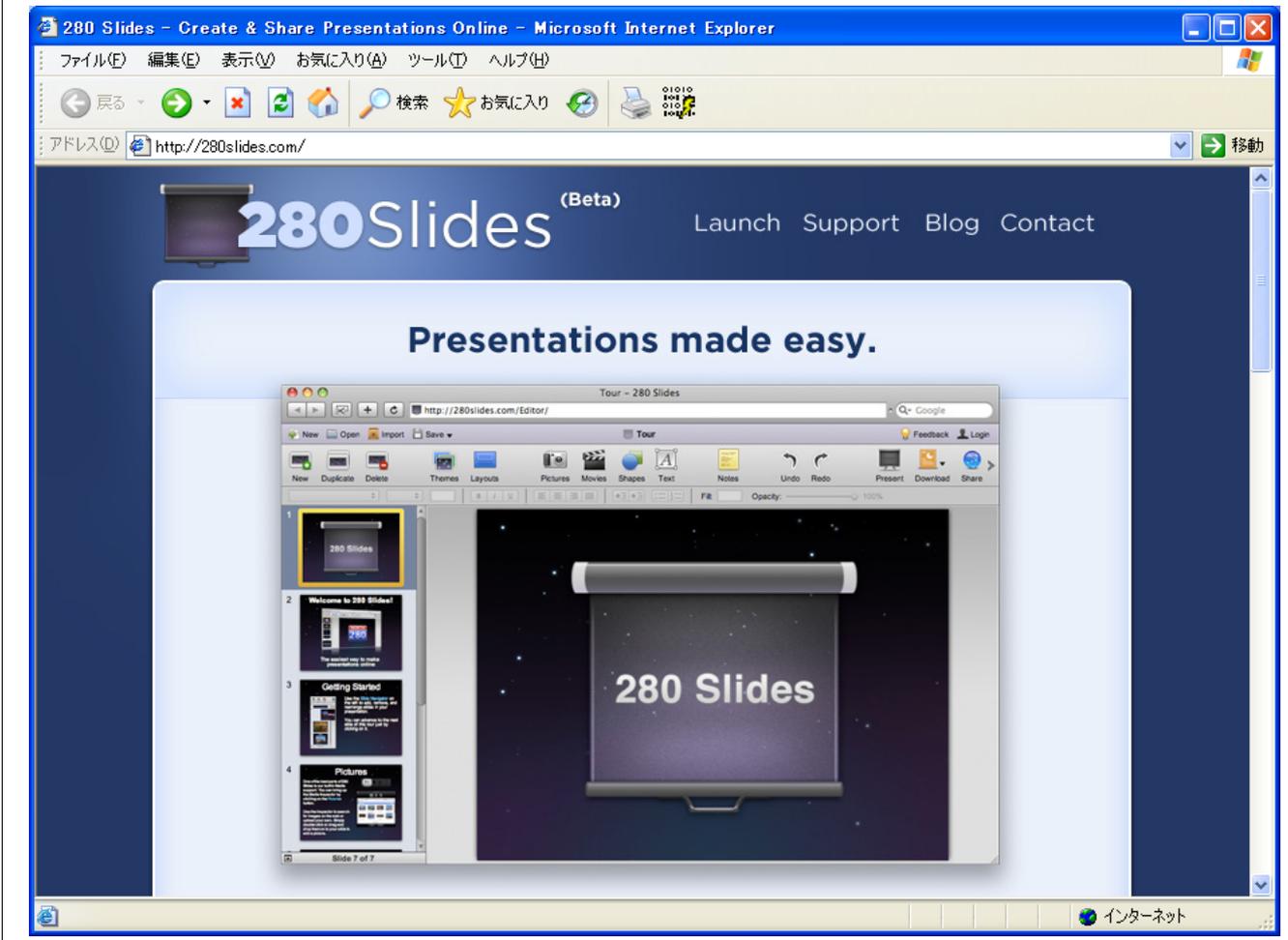
MobileMe の開発で採用
<http://www.sproutcore.com/>

HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008

Cappuccino & Objective-J

- Cocoa like
 - Cappuccino implements most of the familiar APIs from GNUstep and Apple's Cocoa frameworks.
- Objective-J
 - Cappuccino was implemented using a new programming language called Objective-J, which is modelled after Objective-C and built entirely on top of JavaScript.
- Cappuccino
 - When you program in Cappuccino, you don't need to concern yourself with the complexities of traditional web technologies like HTML, CSS, or even the DOM. The unpleasantness of building complex cross browser applications are abstracted away for you.

HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008



(1)

言語VM

JavaScript

“orto”

JavaVM on JavaScript

アクセラートジャパン
小林 悠 (yukoba) さん

Orto の仕組み

File Edit View History Bookmarks Window Help

http://orto.accelart.jp/docs/tutorial/Basic/Internal/Internal.html

Orto の仕組み

Orto は Web アプリケーションを作成するための言語です。Servlet, PHP を補完します。より理解を深めるために、その動作の仕組みを説明します。

ブラウザは1998年ごろから、従来の静的HTMLだけでなく、DynamicHTMLも表示できるようになりました。DynamicHTMLは3つの要素から成立しています。

```

graph TD
    A[DynamicHTML] --> B[静的HTML]
    A --> C[CSS]
    A --> D[JavaScript]
  
```

静的HTMLが文章の論理構造を扱い、スタイルシート(CSS)が色などの表示面を扱い、JavaScriptが動的な側面を扱うように分かれています。OrtoはこのDynamicHTMLを使っています。

```

graph LR
    A[Javaソースファイル] -- Javaコンパイラ --> B[Javaクラスファイル]
    B -- Orto --> C[DynamicHTML]
  
```

Java コンパイラが Java ソースファイルを Java クラスファイルに変換し、Orto がさらに、それを Dynamic HTML に変換します。Dynamic HTML なので、プラグインは必要ありません。Web アプリケーションの利用者はすぐに表示することが出来ます。

http://orto.accelart.jp/samples/Xetris/Xetris-ja.js

File Edit View History Bookmarks Window Help

http://orto.accelart.jp/samples/Xetris/Xetris-ja.js

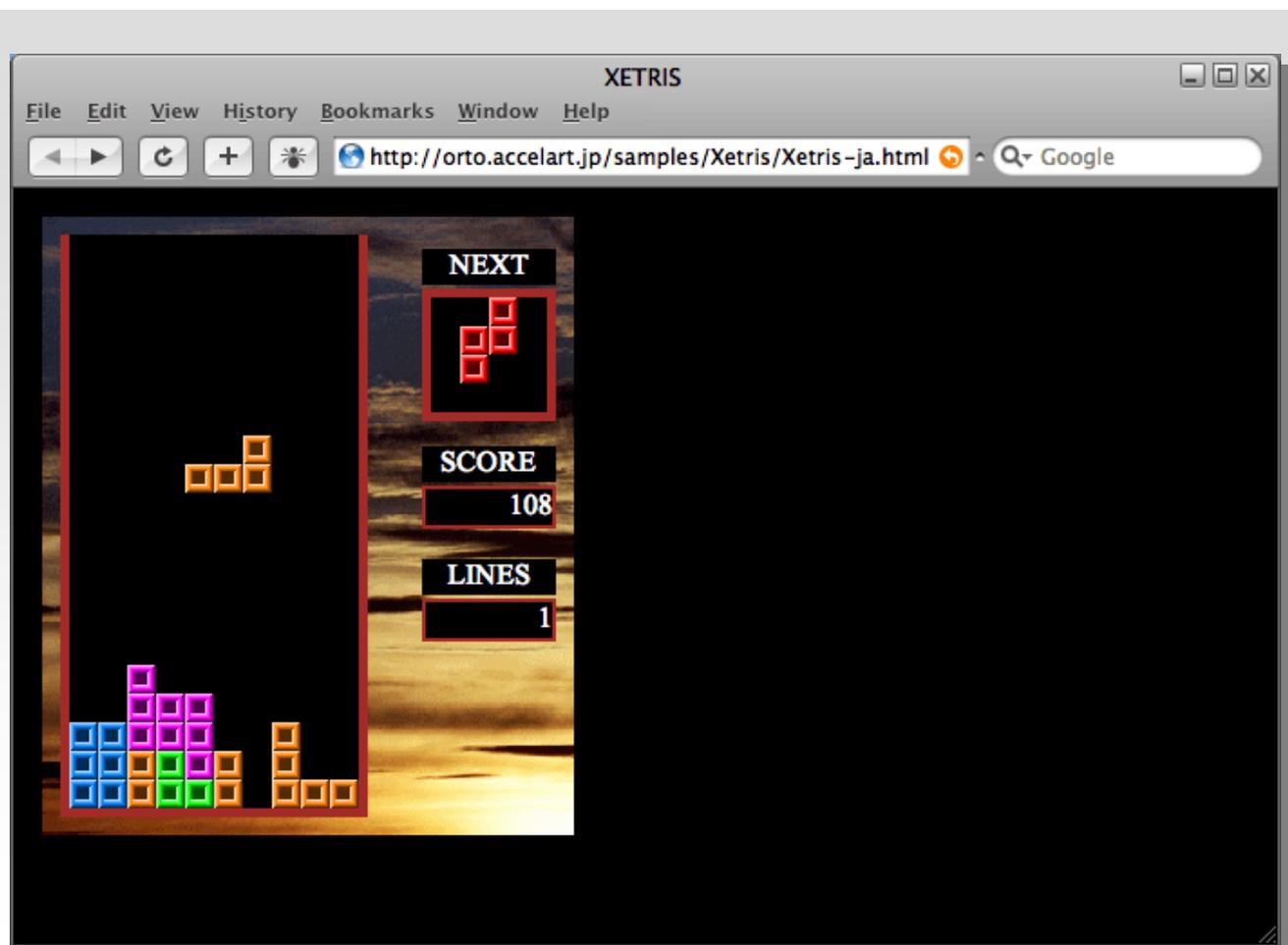
```

805360701,0,3,,,,,1,[],"<init>()V",1818100338,-1,[],[[]],[[169,8,[77],,,,],[[8840,-
1901636148,3,3,,,,,[[19,34,37],[37,42,37]],"componentAction(Lorto/ui/Component;I
1343810157,-1,8)],[[1151949384,0,,1,,8,"val$FrameLorto/ui/IFrame;"],[-
1914930104,0,,1,,8,"this$0Lorto/ui/PanelCreateableContainer;"]],[[170,97,[],1,,1],
805360701,0,3,,,,,1,[],"<init>()V",1818100338,-
1,],[8894,943196877,1,3,,,,,1,[],"<init>(Ljava/lang/String;)V",1818100338,-
1,10],[[]],[[171,25,[],1,],],[[8903,-
805360701,0,3,,,,,1,[],"<init>()V",1818100338,-
1,],[8911,803651488,1,3,,,,,1,[],"<init>(I)V",1818100338,-
1,4],[8950,943196877,1,3,,,,,1,[],"<init>(Ljava/lang/String;)V",1818100338,-
1,10],[[]];
mainClassNo=[33];
.....
ortoResourcePos={};
ortoResourceLen={};
var binData=[];
binData[0]=[177,42,43,166,0,7,4,167,0,4,3,172,42,3,182,14,139,21,35,2,177,42,1,183,
0,9,0,3,2,177,42,183,0,8,0,0,1,42,8,181,150,134,187,37,184,0,0,0,9,0,0,0,77,42,44,1
82,233,154,86,14,1,181,150,134,187,37,42,43,181,130,63,168,143,42,42,180,150,134,18
7,37,182,127,249,219,84,2,177,27,16,10,163,0,8,27,4,162,0,14,187,0,85,89,183,0,85,0
,0,1,191,42,27,181,150,134,187,37,177,42,180,150,134,187,37,172,42,183,0,8,0,0,1,17
7,42,183,0,8,0,0,1,43,89,77,194,42,43,183,0,10,0,2,2,44,195,167,0,8,78,44,195,45,19
1,177,42,183,0,8,0,0,1,177,42,183,0,9,0,2,1,42,182,78,123,227,211,1,177,42,183,0,53
,0,0,1,177,42,43,183,0,53,0,1,2,177,42,180,206,229,156,242,172,43,193,0,18,0,153,0,
29,42,180,206,229,156,242,43,192,0,18,0,182,188,181,9,199,1,160,0,7,4,167,0,4,3,172
,3,172,42,183,0,34,0,0,1,177,42,43,183,0,34,0,1,2,177,42,183,0,34,0,0,1,177,42,43,1
83,0,34,0,1,2,177,42,183,0,27,0,0,1,177,42,43,182,15,216,129,55,2,192,0,111,0,176,4
2,183,0,21,0,0,1,177,42,183,0,21,0,0,1,177,42,18,0,184,0,0,0,27,0,4,0,182,181,151,7

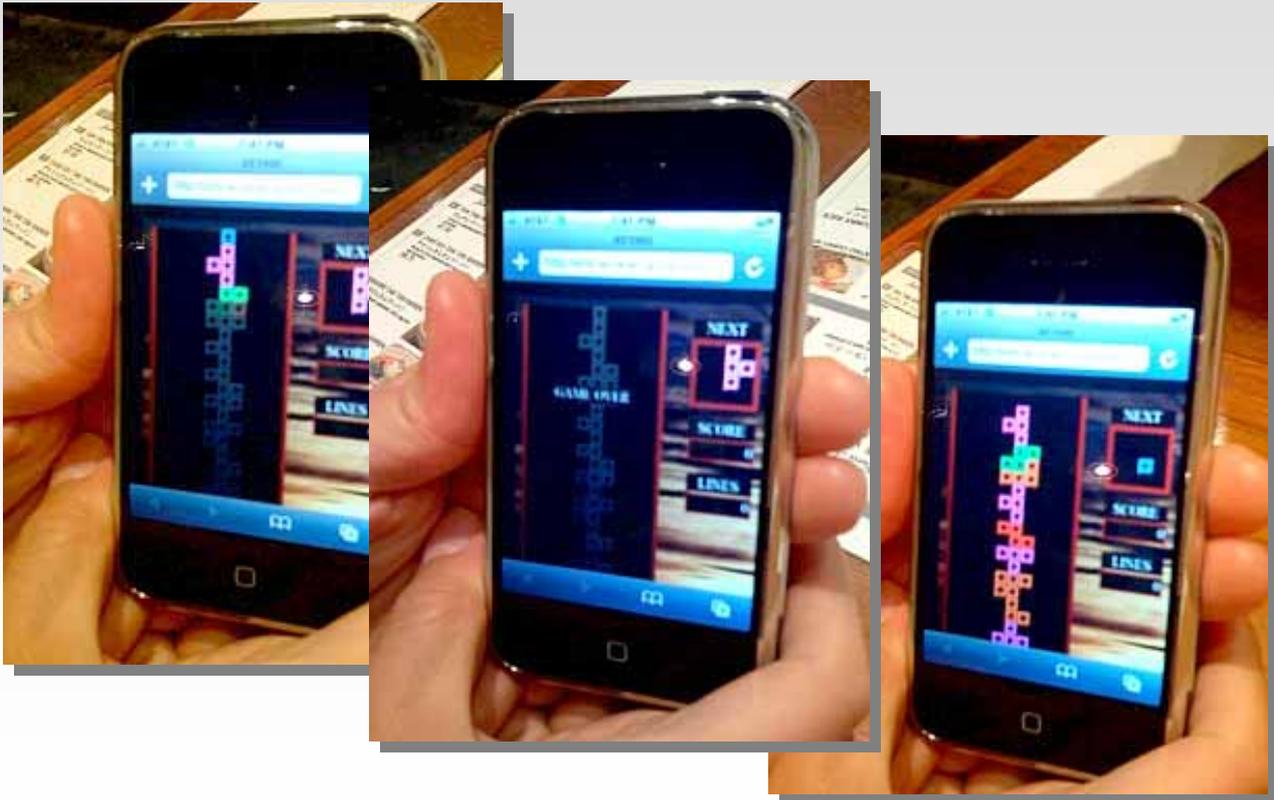
```

JavaScript + DHTML による JavaVM の実装

Java バイトコード (J2ME)



It works on iPhone!



<とおる。>: But I can't control because iphone has no keyboard;(

Logo on JavaScript (yukobaさん)

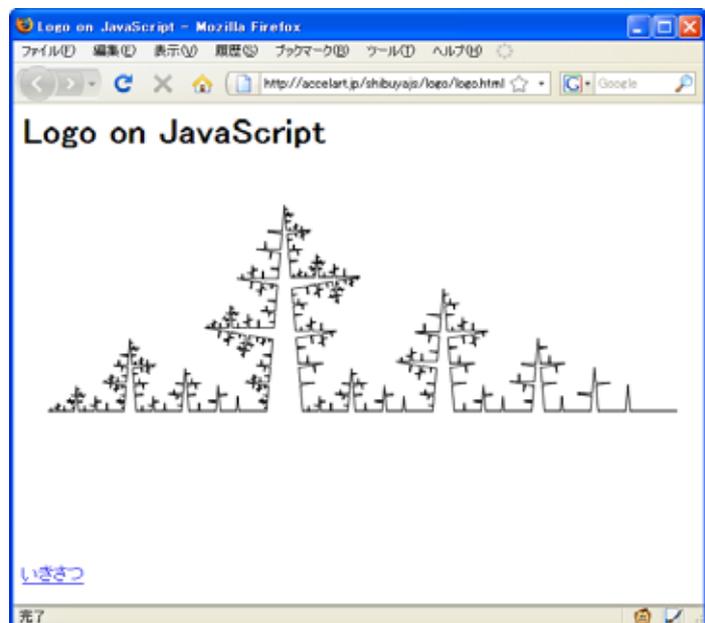
■ <http://accelart.jp/shibuyajs/logo/logo.html>

```
<script type="text/logo">
手順は 森の一部を描け :何回目 :長さ
もし :何回目 = :緻密さ [
  前へ :長さ
  止まれ
]
森の一部を描け :何回目 + 1 :長さ * 0.6
右へ 85
森の一部を描け :何回目 + 1 :長さ / 3
左へ 170
森の一部を描け :何回目 + 1 :長さ / 3
右へ 85
森の一部を描け :何回目 + 1 :長さ * 0.35
終わり

手順は 森を描け :緻密さ
ペンを上げる
位置は [ 270 -20 ]
左へ 90
ペンを下ろせ

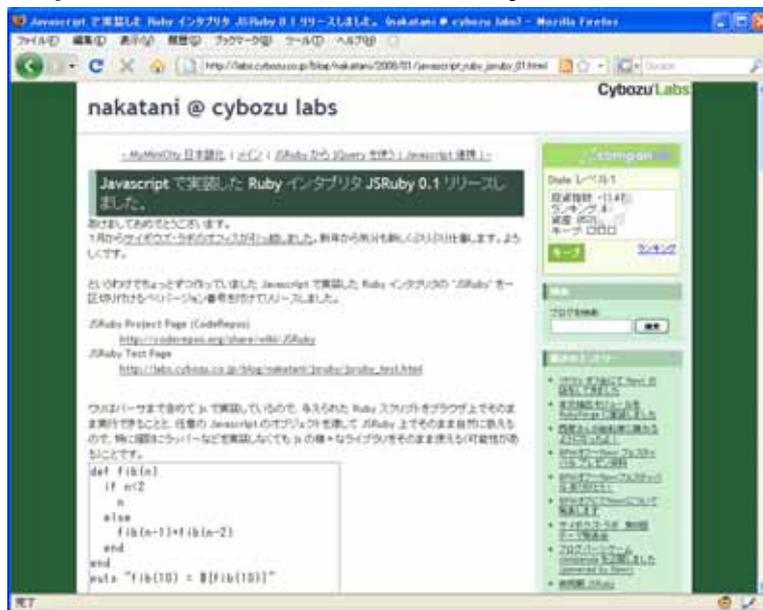
森の一部を描け 0 550
終わり

森を描け 5
</script>
```



JsRuby (中谷さん)

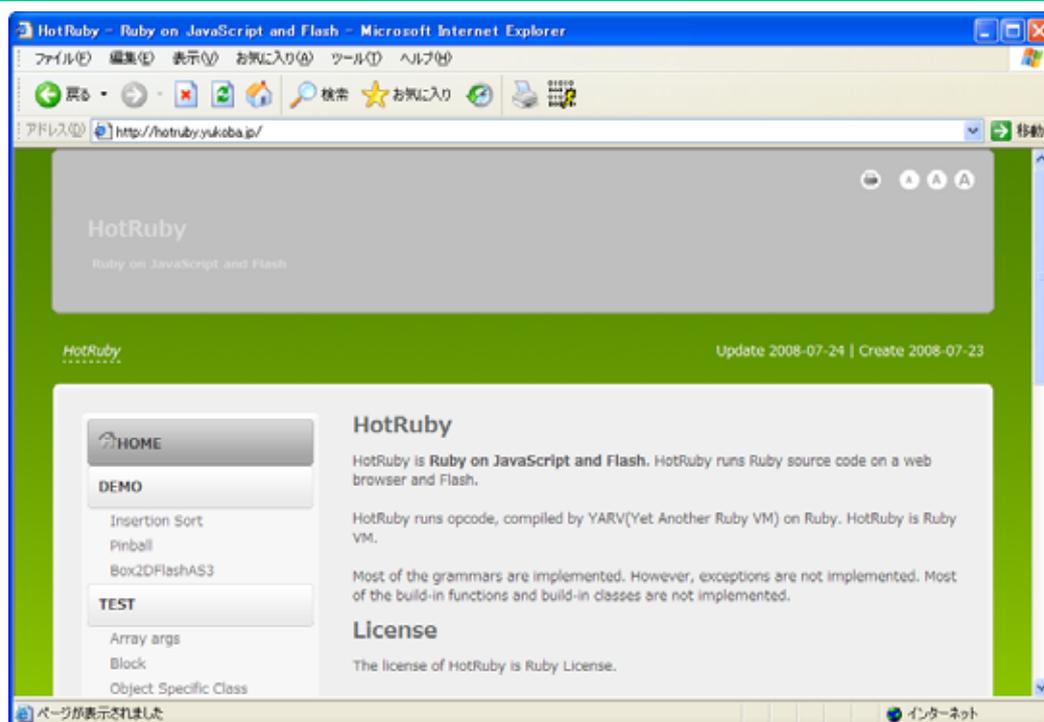
■ JavaScript で実装した Ruby インタプリタ



http://labs.cybozu.co.jp/blog/nakatani/2008/01/_jsruby.html

HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008

HotRuby - Ruby on JavaScript and Flash



<http://hotruby.yukoba.jp/>

HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008

早速 AIR で Flash Player を作ってみた

■ yossyさん @ Shibuya.abc



HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008

Flash C Compiler (Scott Petersen@Adobe)

- FlaCC is a research project that compiles C code to ActionScript using llvm-gcc with a custom flash code generator.



<http://llvm.org/devmtg/2008-08/>

HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008

第1回 LLVM 勉強会 (2008/08/23)

1. JIT のやりかた by syoyo
2. Partial evaluation のやりかた by syoyo
3. LLVM First Steps by nyaxt
4. An LLVM benchmark by shinh
5. オレ言語の LLVM フロントエンド
コンパイラの作り方 by syoyo
 - (MUDA, rsl2llvm, py2llvm あたりの経験を元に)
 - py2llvm
 - rsl2llvm
6. バイナリだけが出力じゃない by omo

(2)

Server Side JavaScript

サーバサイド JavaScript の有名どころ

- Microsoft IIS
 - ASP (Active Server Page)
 - VBScript / PerlScript / JScript ...
- LiveWare JavaScript から発展
 - Netscape Enterprise Server
 - iPlanet
 - Sun Java System Web server
- BroadVision
 - One-To-One Enterprise



いろいろ



Ajaja とは？

- サーバサイド JavaScript のオープンソース実装
 - IIS/ASP の記法がそのまま使えるように配慮
 - 過去の資産が継承できる
- 処理系
 - SpiderMonkey を採用 (C言語)
 - Mozilla Firefox での豊富な実績
- CGI で提供
 - /usr/bin/asp_js コマンド
- SQLite3標準サポート
 - `use('SQLite');`
 - `db = new SQLite('sample.db');`

Ajaja の名前の由来

■ Ajaja

- Asynchronous JavaScript and JavaScript/ASP

- Asynchronous JavaScript and XML Ajax

- Ajaxのもじり どう見ても後付です。本当にありがとうございます。

■ 最初の動機

- ブラウザ上で動く JavaScript と、ウェブサーバで動く JavaScript とを相互に通信させることで、JavaScript のみによる Ajax なシステムを作れるようにしてみよう

■ 開発者

- 西田圭介さん (+ アルファギークな仲間達)

- OpenCOBOLの開発者、Gonzuiの共同開発者

- 現在、某商用VPNソフトウェアを開発中

■ ライセンス

- MPL, GPL, LGPL のトリプルライセンス

HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008

Ajaja サンプルプログラム

■ hello.asp

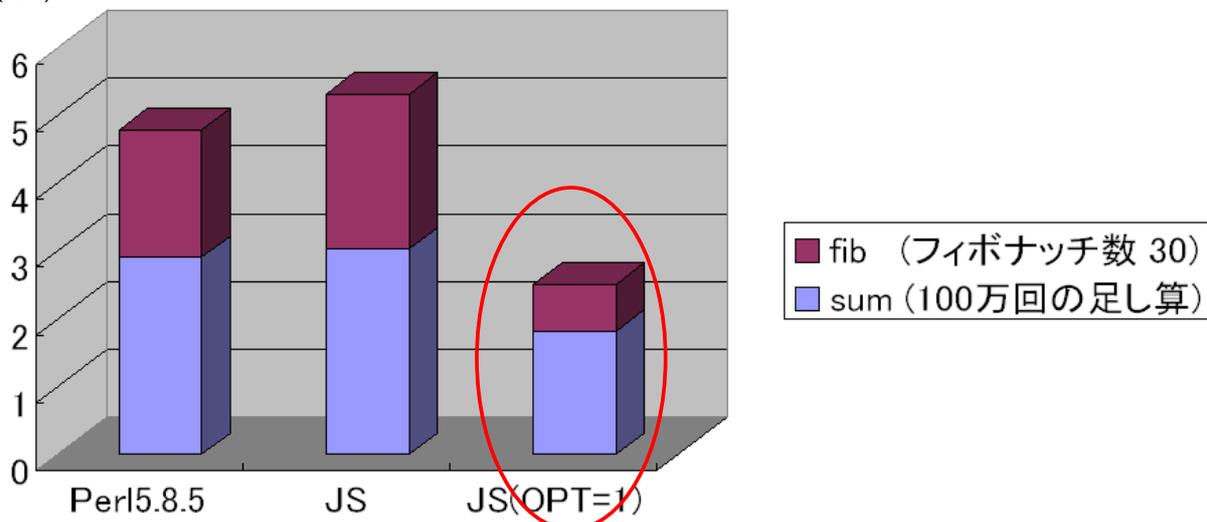
```
<%@ Language=JavaScript %>
<%
    var hello = "Hello world";
%>
<html>
<head>
    <title><%= hello %></title>
</head>
<body>
    <h1><%= hello %></h1>
</body>
</html>
```

HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008

実行処理系のベンチマーク

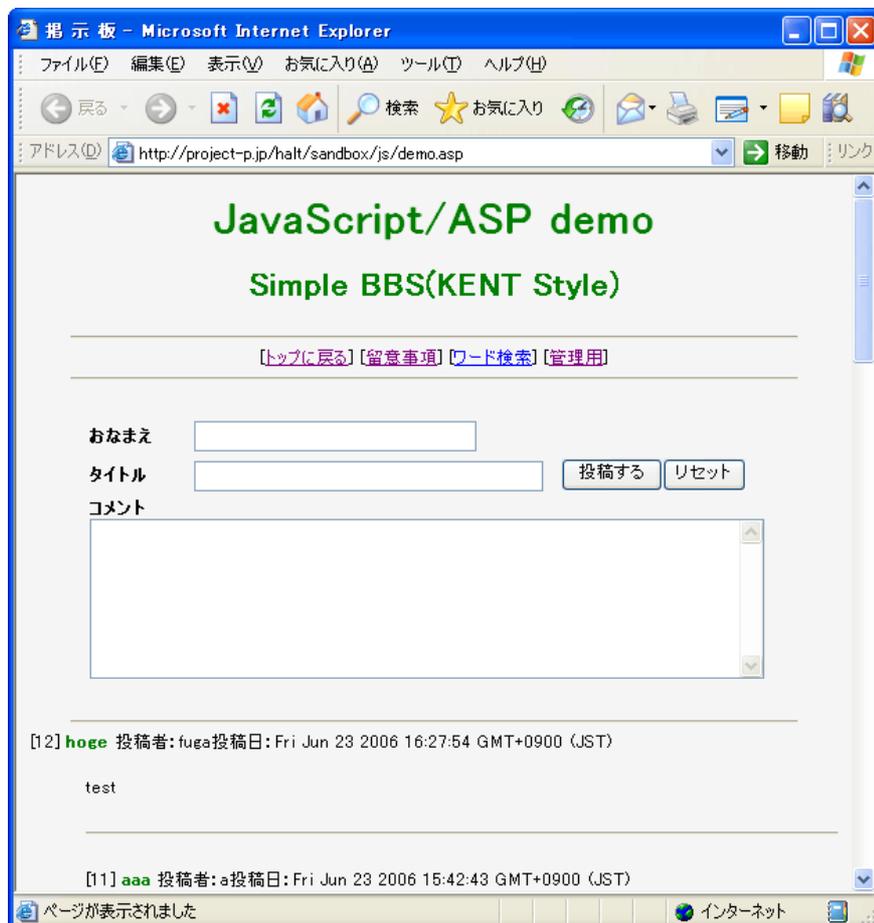
CPU: Pentium 4 3GHz
OS: RHEL4 Update 3 (Linux 2.6.9-34)

(秒)



JS は SpiderMonkey を DEBUG モードでビルド
JS(OPT=1)は SpiderMonkey を BUILD_OPT=1 でビルド

HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008



[ソース]

(3)

PHP

JavaScript

JSmarty

(PHP Template on JS)

bbs.tpl (JSmarty - Smarty互換)

■ HTMLテンプレート(ソースから一部抜粋)

```
{foreach from=$rows item='row'}  
<d1>  
<dt>  
[{$row[0]}]&nbsp;  
<strong style='color:green;'>{$row[1]}</strong>&nbsp;  
投稿者 : {$row[2]}  
投稿日 : {$row[3]}  
</dt>  
<dd><p>{$row[4]}</p></dd>  
</d1>  
<hr>  
{/foreach}
```

JSmarty - ブラウザと Ajaja の両方に対応

■ Smarty

- PHPの代表的なテンプレートエンジン

■ JSmarty

- JavaScriptで動くテンプレートエンジン

■ Smartyクローン

- できるだけSmarty互換(少し独自仕様あり)
- 利用するメソッドやプロパティととかは大体そのまま
- Smartyのテンプレートがそのまま利用できる
- プラグインとかも対応
 - プラグインロードには JSAN.js 利用

HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008

サーバ/クライアントで .js を共有

■ JavaScriptライブラリをサーバサイドでも

- prototype.js

- jQuery

- どこでもJavaScriptライブラリ

■ (例) Validationロジックの共通化

- フォームの入力値のチェック

- 数値、文字列の長さ

- クライアント側のチェック サーバ送信

- DRYの原則 (Don't Repeat Yourself)

- Ruby on Rails の設計指針

- サーバ・クライアント両方でソースを共有

HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008

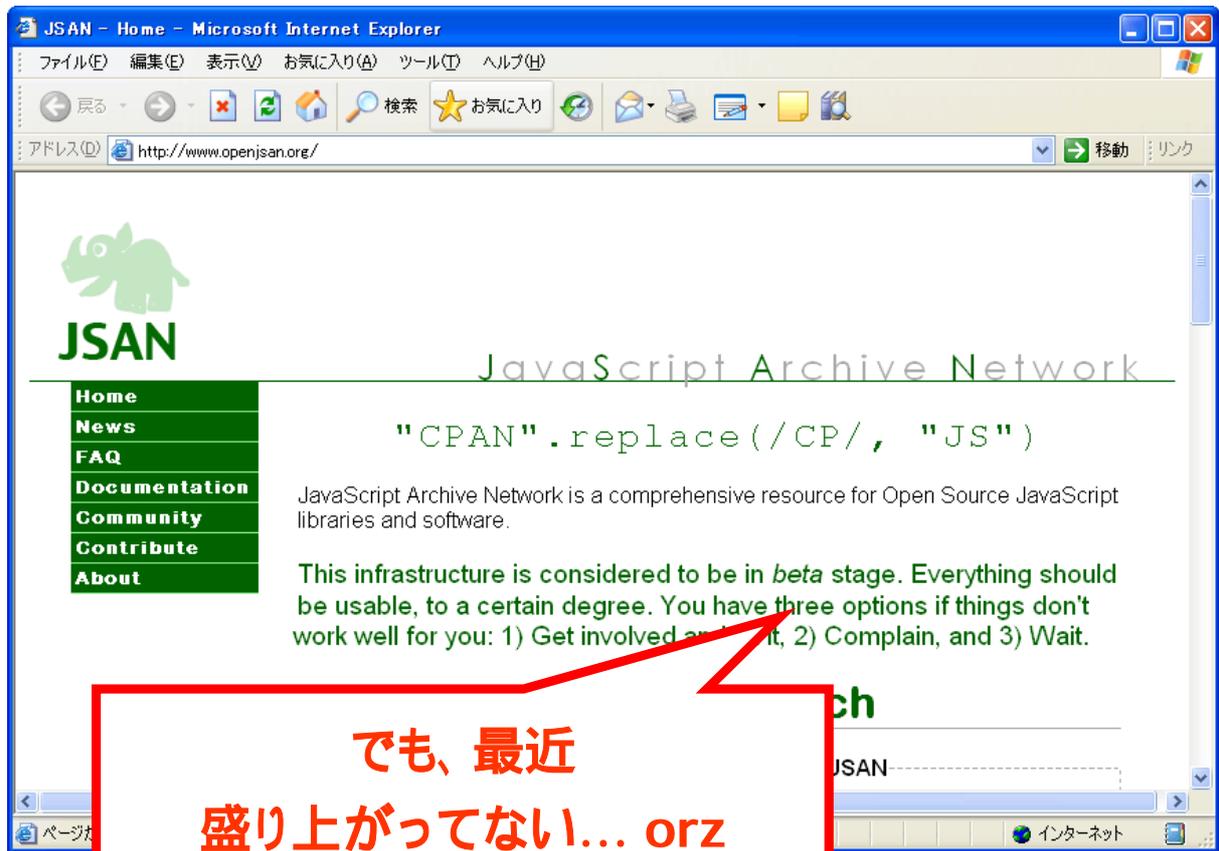
Wiki.Formatter in JavaScript (kazuhoさん)

```
var formatter = new Wiki.Formatter();
document.getElementById('wiki').innerHTML = formatter.format(srcText);
```

```
var formatter = new Wiki.Formatter({
  autoLink: false, // http:// 等の文字列をリンク化しない
  foldLines: true, // 改行を <br /> に変換する
  nameToLink: function (name) { // WikiName => リンクへの変換
    return 'http://example.com/wiki/' + name;
  },
  initHandlers: function () { // 記法定義をオーバーライド
    this.initHandlers();
    // 順序なしリストのレベル 1-3 として · を登録
    this.addTaggedStatement('.', '<li>', 'ul', 1);
    this.addTaggedStatement('...', '<li>', 'ul', 2);
    this.addTaggedStatement('...', '<li>', 'ul', 3);
    // &fontweight(フォントの重さ) { ... } という記法を追加
    this.addExpression(
      function (stack) {
        stack[1].out +=
          '<span style="font-weight: " + stack[0].params + ">'
          + stack[0].out
          + '</span>';
      },
      '&fontweight',
      '¥}',
      '&fontweight¥¥¥(¥¥¥s*¥¥¥d+¥¥¥s*)¥¥¥s*¥¥¥{' );
    }
  });
```

HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008

OpenJSAN



(4)

Perl

JavaScript

PMconnect



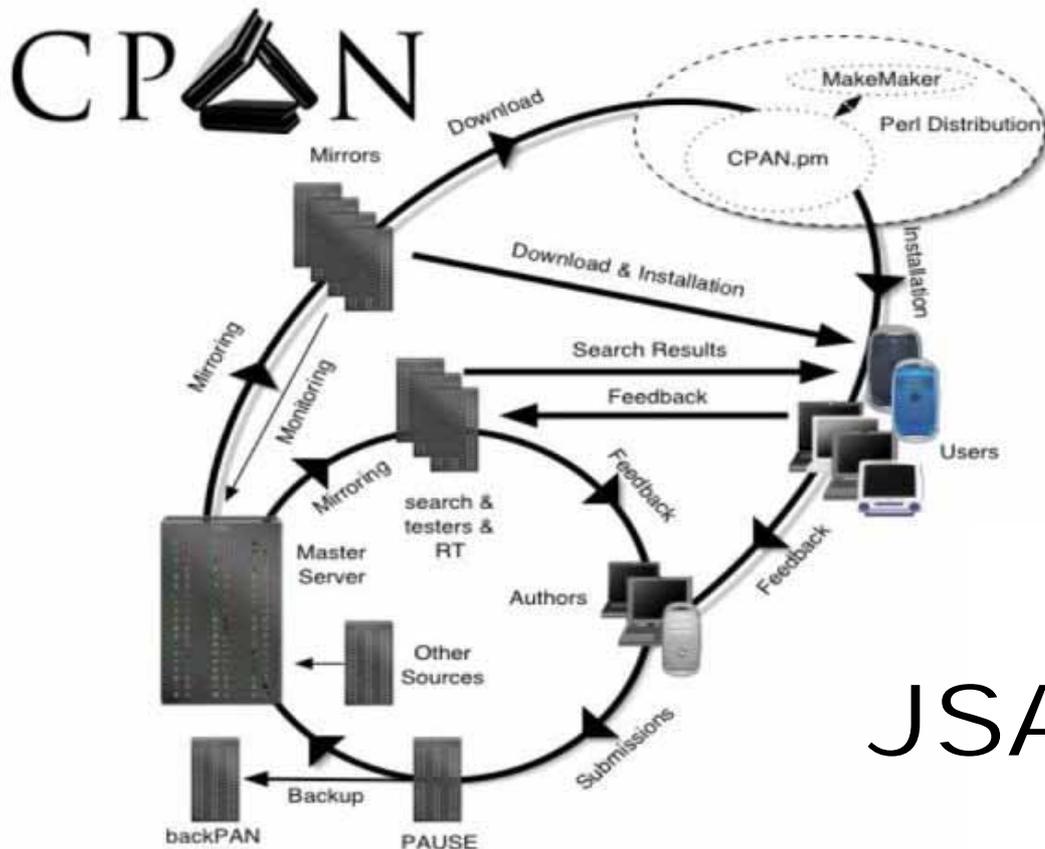
(JS + Perl CPAN)

■ SpiderMonkeyとPerlのバインディング

```
/* GD グラフィックスライブラリ */
Perl.Use("GD");
var im = new Perl("GD::Image",100,100);
var white = im.colorAllocate(255,255,255);
var black = im.colorAllocate(0,0,0);
var red = im.colorAllocate(255,0,0);
var blue = im.colorAllocate(0,0,255);
im.transparent(white);
im.interlaced('true');
im.rectangle(0,0,99,99,black);
im.arc(50,50,95,75,0,360,blue);
im.fill(50,50,red);
var s = im.png();

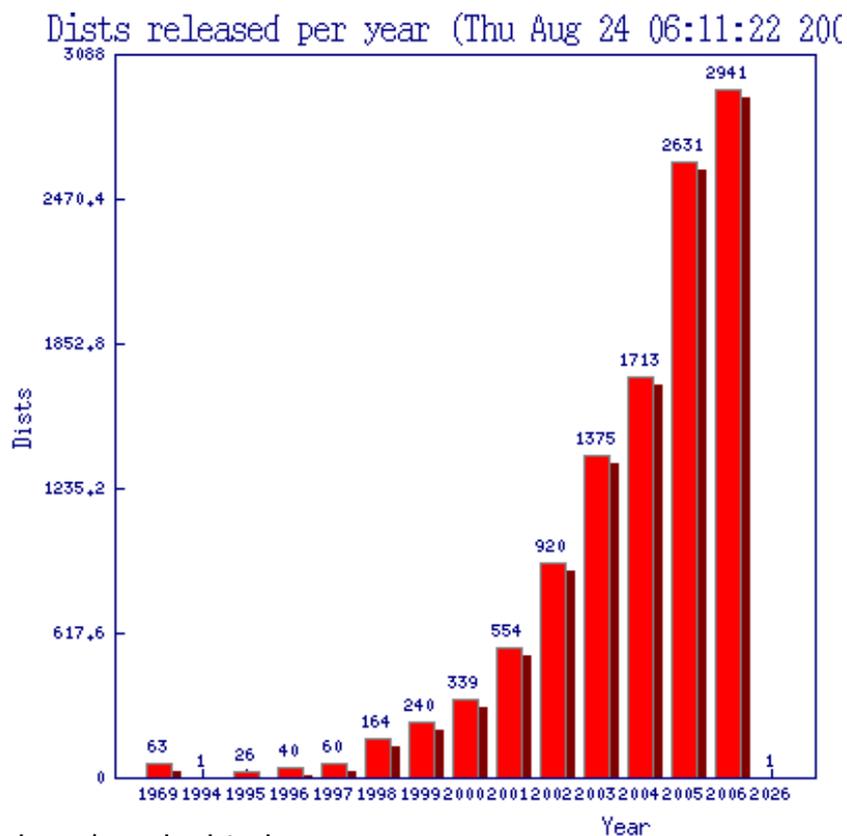
Perl.Use("FileHandle");
var fh = new Perl("FileHandle", "gdout.png", "w");
fh.print(s);
fh.close();
```

変数に \$ を
つけなくても
Perlのライブラリ
が使える！！



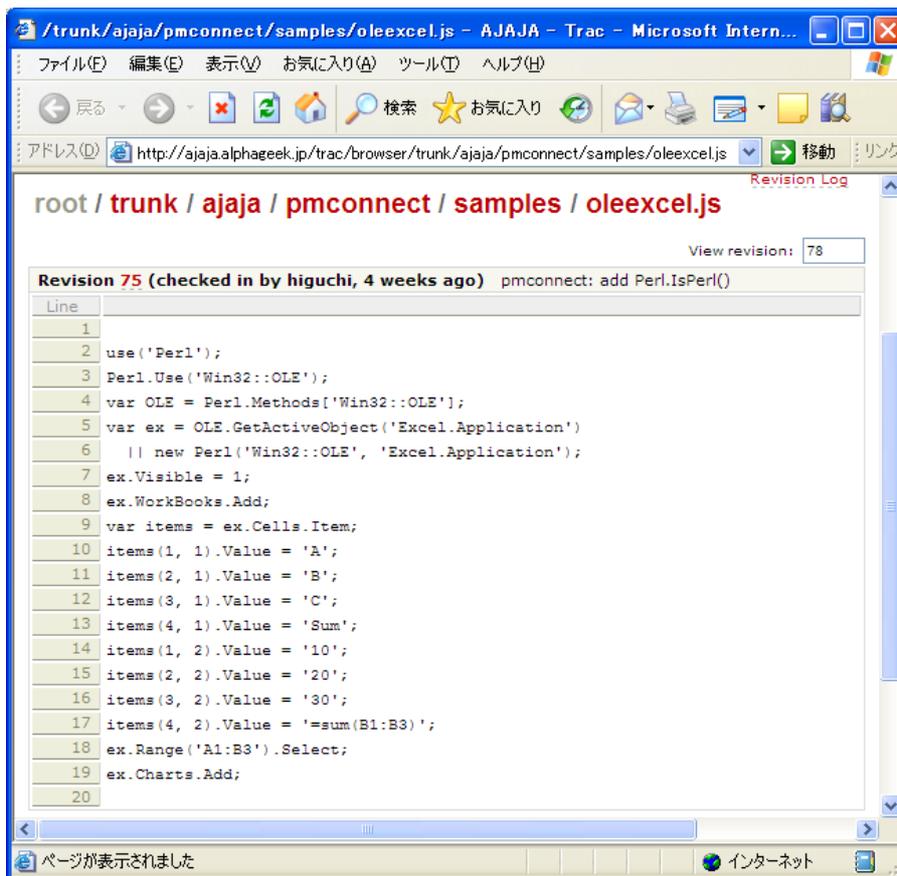
JSAN

CPAN には使えるPerlライブラリがいっぱい存在



<http://cpants.perl.org/graphs.html>

pmconnect - samples



JavaScript から Perl の XML パーサーを使う

```
$ jsp1 dom.js
```

```
use("Perl");
Perl.Use("XML::DOM");
var parser = new Perl("XML::DOM::Parser");
var doc = parser.parsefile("test.xml");
var nodes = doc.getElementsByTagName("foo");
var n = nodes.getLength();
for (var i = 0; i < n; ++i) {
    var node = nodes.item(i);
    var txelm = node.getFirstChild();
    print(txelm.getNodeValue());
}
print(doc);
doc.dispose();
```

test.xml

```
<?xml version="1.0"
encoding="UTF-8"
standalone="no"?>
<data>
  <foo>foo1</foo>
  <bar>bar1</bar>
  <foo>foo2</foo>
  <bar>bar2</bar>
</data>
```

Ajajaの戦略

■ 新しいプログラミング言語は作らない

■ SpiderMonkeyを採用

- 最も普及しているJavaScriptの処理系

■ IIS/ASPの記法をサポート

- VBプログラマ層の取り込み

■ JSmartyも使えるよ

- PHPテンプレートエンジンSmartyのJavaScript実装

■ JavaScriptのライブラリは少ない

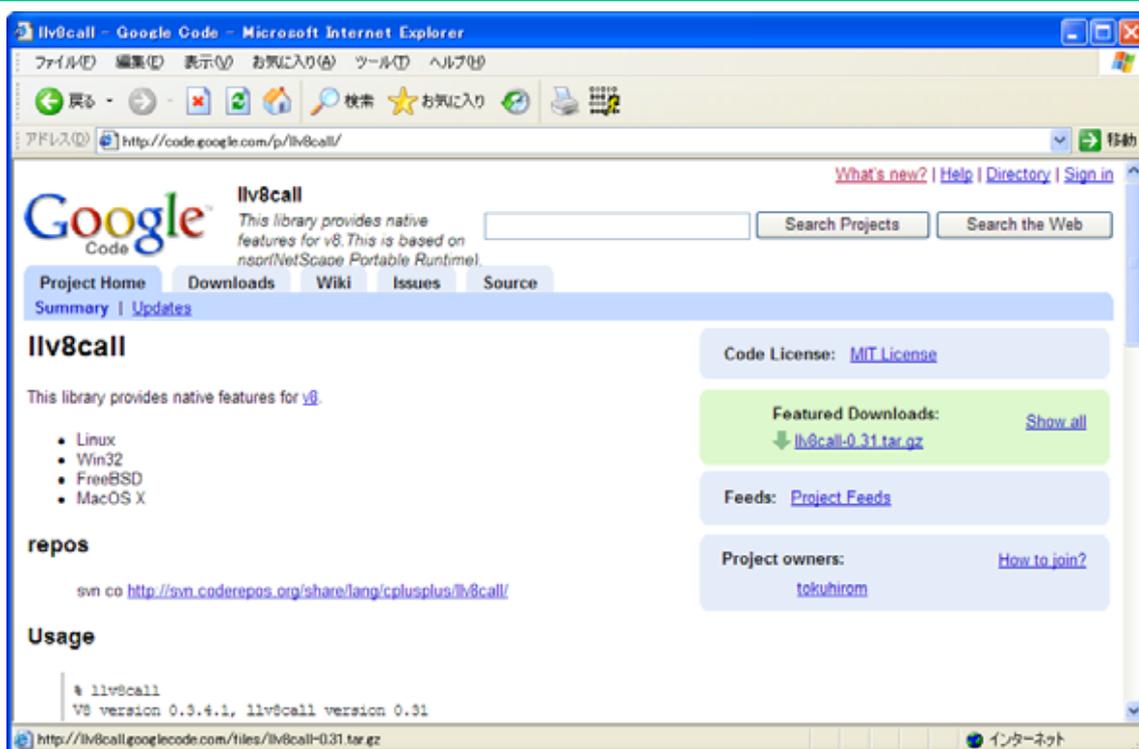
■ PMconnect

- PerlのライブラリをJavaScriptの構文で利用できる
- 既にある豊富なPerlのCPANモジュールを活用

■ OpenJSAN

HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008

Llv8call (tokuhiromさん)



HTTP Meeting 2008 ~ Webサービスの明日を見つけよう ~ @ InternetWeek 2008

最後に

