

# Front-End Modular & Automated Development

建立前端團隊開發環境

蔣定宇 / josephj / 啊嗚

<http://josephj.com> / [josephj6802@gmail.com](mailto:josephj6802@gmail.com)





現任職於 **D-Link**  
Building Networks for People

TELEVISION

INFORMATION WINDOW

DESKTOP



**muchiii**  
Powered by D-Link

**D-Link 網路服務處**  
在 D-Link 現有的網路設備上  
建立服務、強化與使用者的關聯性

因為超有趣的概念、決定進入 D-Link  
當時只想要發揮前端的專長、把產品做出來  
卻怎麼也沒想到...

因 D-Link 過去幾乎沒有開發大型網站的經驗，一進去什麼都沒有、得從零開始，我居然就開始去**建立開發環境與決定軟體架構**。

Software Engineer      Project Manager  
Service Engineer      System Architect      DBA  
Front-end Engineer      QA Engineer      QA Tester  
System Analyzer      Operation Engineer

而這些事情在前一家公司 (Y!) 是沒有機會碰的。因為它分工明確、高手多，且很多工具甚至有專門負責的團隊，加上各個服務大都已经有了解決方案了，能够真的大刀闊斧地改變現有環境、或注入自己想法的機會相對較少。

而在過程之中，我發現自己非常喜歡做前後端整合的工作。若只是單純 HTML/CSS/JavaScript 的撰寫以不能滿足我。結合其它技術讓前端能夠更好（開發速度、效能、模組化、維護性）是我現在熱血的方向。

Azure    Firmware  
Flash    I18N/L10N    PEAR  
CMS    Apache Module    Hardware  
Android    Win32    Objective C  
Embedded System  
Server-side Script

非常鼓勵前端工程師跨往不同的領域  
因只有前端工程師才知道前端工程師要的是什麼

# 本日大綱

- 模組化開發：網站的基本單元  
不管 HTML, CSS, JavaScript 都以模組開發
- 自動化開發：讓犯錯變得困難  
Nicholas Zakas, “Make it difficult to make mistakes.”

# DIV 模組化開發

Modular Development

# Modular Development

## 模組化是現今網站主流

不再以「頁」為開發邏輯

而是開發一個個可獨立運作的「DIV 模組」

它有獨立的 HTML、CSS、JS 檔案

## Photo Viewer

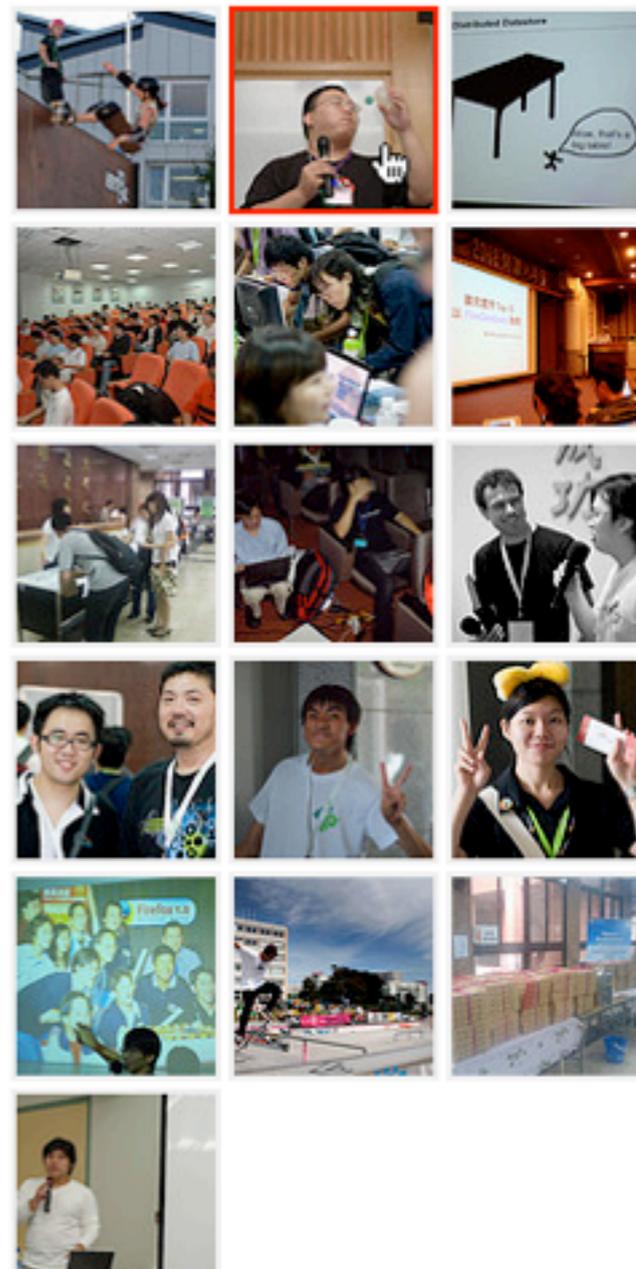


## Photo Filter

Search:

[Find Photos](#)

## Flickr Photo List



## About JavaScript Platform

### 問題

- JavaScript 需處理模組之間溝通的問題，透過全域變數或方法傳遞會沒辦法彼此獨立。

### 概念

- 請參考 Nicholas Zakas 的 [Scalable JavaScript Architecture](#)

### 好處

- 每個人開發邏輯一致、花點時間了解概念都可以快速上手。

## Photo Viewer

### 照片顯示模組

`_photo_view.php`  
`_photo_view.css`  
`_photo_view.js`

## Photo Filter

### 搜尋模組

Search:   
`_photo_filter.php`  
`_photo_filter.css`  
`_photo_filter.js`

## Flickr Photo List

### 列表模組

`_photo_list.php`  
`_photo_list.css`  
`_photo_list.js`

## About JavaScript Platform

### 問題

- JavaScript 需處理模組之間溝通的問題，透過全域變數或方法傳遞會沒辦法彼此獨立。

### 概念

- 請參考 Nicholas Zakas 的 [Scalable JavaScript Architecture](#)

### 好處

- 每個人開發邏輯一致、花點時間了解概念都可以快速上手。

# HTML - 由許多模組所構成

```
Aptana Web - sftp://josephj@josephj.com:443/var/www/modev/frontend/views/index.php - Aptana Studio - /Users/josephj/Documents/Ap...  
My Studio index.php  
1 <?php include_once VIEW_PATH . "_before_body.php"; ?>  
2 <div id="hd">  
3 <div class="yui3-d3">  
4 <?php include_once VIEW_PATH . "_masthead.php"; ?>  
5 </div><!-- .yui3-d3 (end) -->  
6 </div><!-- #hd (end) -->  
7 <div id="bd">  
8 <div class="yui3-d3">  
9 <div class="yui3-t6">  
10 <div class="yui3-main">  
11 <div class="yui3-b">  
12 <?php include_once VIEW_PATH . "_photo_viewer.php"; ?>  
13 <?php include_once VIEW_PATH . "_introduction.php"; ?>  
14 </div><!-- .yui3-b (end) -->  
15 </div><!-- .yui3-main (end) -->  
16 <div class="yui3-b">  
17 <?php include_once VIEW_PATH . "_photo_filter.php"; ?>  
18 <?php include_once VIEW_PATH . "_photo_list.php"; ?>  
19 </div><!-- .yui3-b (end) -->  
20 </div><!-- .yui3-t4 (end) -->  
21 </div><!-- .yui3-d3 (end) -->  
22 </div><!-- #bd (end) -->  
23 <div id="ft">  
24 <div class="yui3-d3">  
25 <?php include_once VIEW_PATH . "_mastfoot.php"; ?>  
26 </div><!-- .yui3-d3 (end) -->  
27 </div><!-- #ft (end) -->  
28 <?php include_once VIEW_PATH . "_after_body.php"; ?>  
29
```

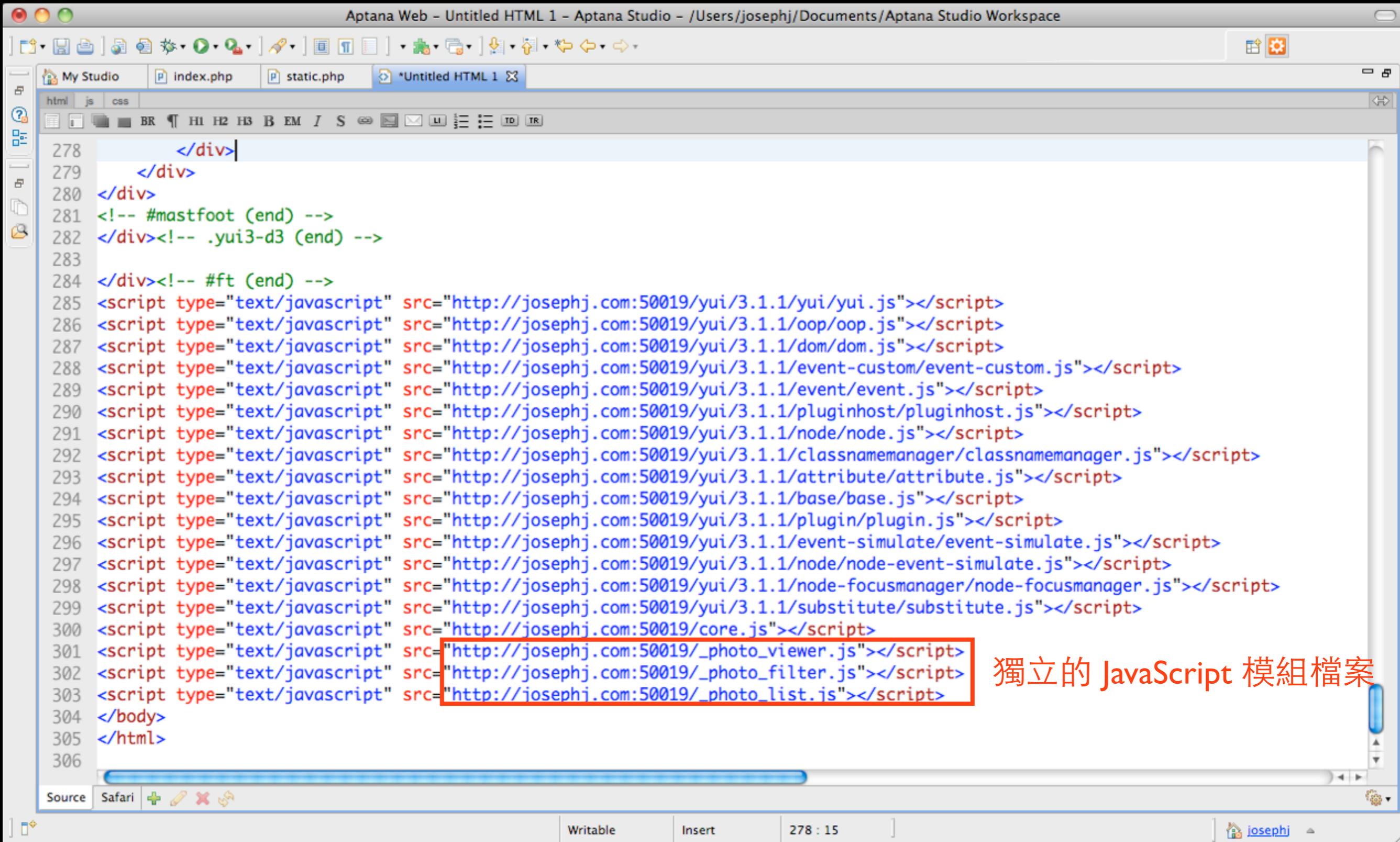
獨立的 HTML 模組檔案

# CSS - 由許多模組所構成

```
Aptana Web - Untitled HTML 1 - Aptana Studio - /Users/josephj/Documents/Aptana Studio Workspace  
My Studio | index.php | static.php | *Untitled HTML 1  
html | js | css  
BR | H1 | H2 | H3 | B | EM | I | S | LI | TD | TR  
1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">  
2 <html>  
3 <head>  
4 <meta http-equiv="content-type" content="text/html; charset=utf-8">  
5 <title>Modular Development (Raw) @ webrebuild.org | josephj.com</title>  
6 <link type="text/css" rel="stylesheet" href="http://josephj.com:50019/yui/3.1.1/cssreset/reset.css" media="all">  
7 <link type="text/css" rel="stylesheet" href="http://josephj.com:50019/yui/3.1.1/cssfonts/fonts.css" media="all">  
8 <link type="text/css" rel="stylesheet" href="http://josephj.com:50019/yui/3.1.1/cssgrids/grids.css" media="all">  
9 <link type="text/css" rel="stylesheet" href="http://josephj.com:50019/core.css" media="all">  
10 <link type="text/css" rel="stylesheet" href="http://josephj.com:50019/mod.css" media="all">  
11 <link type="text/css" rel="stylesheet" href="http://josephj.com:50019/_masthead.css" media="all">  
12 <link type="text/css" rel="stylesheet" href="http://josephj.com:50019/_photo_list.css" media="all">  
13 <link type="text/css" rel="stylesheet" href="http://josephj.com:50019/_photo_filter.css" media="all">  
14 <link type="text/css" rel="stylesheet" href="http://josephj.com:50019/_photo_viewer.css" media="all">  
15 <link type="text/css" rel="stylesheet" href="http://josephj.com:50019/_introduction.css" media="all">  
16 </head>  
17  
18 <body class="yui3-skin-sam">  
19 <h1>Modular Development (Raw) @ webrebuild.org</h1>  
20 <div id="hd">  
21 <div class="yui3-d3">  
22 <!-- #masthead (start) -->  
23 <div id="masthead">  
24 <div class="bd clearfix">  
25 <span class="title">  
26 <a name="top">  
27   
28 Modular Development  
29 </a>  
Source | Safari | Writable | Insert | 2 : 7 | josephj
```

獨立的 CSS 模組檔案

# JavaScript – 由許多模組所構成



The screenshot shows the Aptana Studio interface with an HTML file open. The code is as follows:

```
278     </div>|
279   </div>
280 </div>
281 <!-- #mastfoot (end) -->
282 </div><!-- .yui3-d3 (end) -->
283
284 </div><!-- #ft (end) -->
285 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/yui/yui.js"></script>
286 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/oop/oop.js"></script>
287 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/dom/dom.js"></script>
288 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/event-custom/event-custom.js"></script>
289 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/event/event.js"></script>
290 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/pluginhost/pluginhost.js"></script>
291 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/node/node.js"></script>
292 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/classnamemanager/classnamemanager.js"></script>
293 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/attribute/attribute.js"></script>
294 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/base/base.js"></script>
295 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/plugin/plugin.js"></script>
296 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/event-simulate/event-simulate.js"></script>
297 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/node/node-event-simulate.js"></script>
298 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/node-focusmanager/node-focusmanager.js"></script>
299 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/substitute/substitute.js"></script>
300 <script type="text/javascript" src="http://josephj.com:50019/core.js"></script>
301 <script type="text/javascript" src="http://josephj.com:50019/_photo_viewer.js"></script>
302 <script type="text/javascript" src="http://josephj.com:50019/_photo_filter.js"></script>
303 <script type="text/javascript" src="http://josephj.com:50019/_photo_list.js"></script>
304 </body>
305 </html>
306
```

A red box highlights the last three script tags (lines 301-303). A red text annotation "獨立的 JavaScript 模組檔案" (Independent JavaScript Module Files) is placed to the right of the box, pointing to these three lines.

# 模組化的困難

- HTML：很好解決，只要用 include 即可。效能影響不大。
- CSS：以 ID Selector 作為 CSS 命名空間可以做到。但若每個模組有一個檔案，一頁 30 個模組就會有 30 個檔案，Request 對性能影響很大。
- JavaScript：跟 CSS 一樣有檔案數量的問題。另外該如何有效率的做跨模組的溝通？

# **Request 數量的問題**

# static.php - 每個頁面要放置哪些 JS/CSS 的組態檔

```
1 <?php
2 /*
3  * Static files configuration
4  * The proxy between tool and web pages
5  */
6 $static = array();
7
8 $static["raw"][] = array(
9     "type"      => "css",
10    "href"       => STATIC_URL . "yui/3.1.1/cssreset/reset.css",
11    "is_top"     => TRUE,
12 );
13 $static["raw"][] = array(
14     "type"      => "css",
15     "href"       => STATIC_URL . "yui/3.1.1/cssfonts/fonts.css",
16     "is_top"     => TRUE,
17 );
18 $static["raw"][] = array(
19     "type"      => "css",
20     "href"       => STATIC_URL . "yui/3.1.1/cssgrids/grids.css",
21     "is_top"     => TRUE,
22 );
23 $static["raw"][] = array(
24     "type"      => "css",
25     "href"       => STATIC_URL . "core.css",
26     "is_top"     => TRUE,
27 );
28 $static["raw"][] = array(
29     "type"      => "css",
30     "href"       => STATIC_URL . "mod.css",
31     "is_top"     => TRUE,
32 );
```

key - 頁面名稱

is\_top: 放在 </head> 前或 </body> 前

另外有 is\_noscript, media 等屬性可以設定

# static.php 把模組的 CSS/JS 檔用一致的方法設定 但 Request 量過多，勢必會造成性能問題

```
Aptana Web - Untitled HTML 1 - Aptana Studio - /Users/josephj/Documents/Aptana Studio Workspace
My Studio | index.php | static.php | *Untitled HTML 1
html | js | css
BR | H1 | H2 | H3 | B | EM | I | S | U | TD | TR
1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
2 <html>
3 <head>
4 <meta http-equiv="content-type" content="text/html; charset=utf-8">
5 <title>Modular Development (Raw) @ webrebuild.org | josephj.com</title>
6 <link type="text/css" rel="stylesheet" href="http://josephj.com:50019/yui/3.1.1/cssreset/reset.css" media="all">
7 <link type="text/css" rel="stylesheet" href="http://josephj.com:50019/yui/3.1.1/cssfonts/fonts.css" media="all">
8 <link type="text/css" rel="stylesheet" href="http://josephj.com:50019/yui/3.1.1/cssgrids/grids.css" media="all">
9 <link type="text/css" rel="stylesheet" href="http://josephj.com:50019/core.css" media="all">
10 <link type="text/css" rel="stylesheet" href="http://josephj.com:50019/mod.css" media="all">
11 <link type="text/css" rel="stylesheet" href="http://josephj.com:50019/_masthead.css" media="all">
12 <link type="text/css" rel="stylesheet" href="http://josephj.com:50019/_photo_list.css" media="all">
13 <link type="text/css" rel="stylesheet" href="http://josephj.com:50019/_photo_filter.css" media="all">
14 <link type="text/css" rel="stylesheet" href="http://josephj.com:50019/_photo_viewer.css" media="all">
15 <link type="text/css" rel="stylesheet" href="http://josephj.com:50019/_introduction.css" media="all">
16 </head>
17
18 <body class="yui3-skin-sam">
19 <h1>Modular Development (Raw) @ webrebuild.org</h1>
20 <div id="hd">
21 <div class="yui3-d3">
22 <!-- #masthead (start) -->
23 <div id="masthead">
24 <div class="bd clearfix">
25 <span class="title">
26 <a name="top">
27 
30 </div>
31 </div>
32 </div>
33 <!-- #ft (end) -->
34 </div><!-- .yui3-d3 (end) -->
35 </div>
36 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/yui/yui.js"></script>
37 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/ooop/ooop.js"></script>
38 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/dom/dom.js"></script>
39 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/event-custom/event-custom.js"></script>
40 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/event/event.js"></script>
41 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/pluginhost/pluginhost.js"></script>
42 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/node/node.js"></script>
43 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/classnamemanager/classnamemanager.js"></script>
44 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/attribute/attribute.js"></script>
45 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/base/base.js"></script>
46 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/plugin/plugin.js"></script>
47 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/event-simulate/event-simulate.js"></script>
48 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/node/node-event-simulate.js"></script>
49 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/node-focusmanager/node-focusmanager.js"></script>
50 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/substitute/substitute.js"></script>
51 <script type="text/javascript" src="http://josephj.com:50019/core.js"></script>
52 <script type="text/javascript" src="http://josephj.com:50019/_photo_viewer.js"></script>
53 <script type="text/javascript" src="http://josephj.com:50019/_photo_filter.js"></script>
54 <script type="text/javascript" src="http://josephj.com:50019/_photo_list.js"></script>
55 </body>
56 </html>
```

10 個 CSS Request

19 個 JavaScript Request

```
Aptana Web - Untitled HTML 1 - Aptana Studio - /Users/josephj/Documents/Aptana Studio Workspace
My Studio | index.php | static.php | *Untitled HTML 1
html | js | css
BR | H1 | H2 | H3 | B | EM | I | S | U | TD | TR
278 </div>
279 </div>
280 </div>
281 <!-- #mastfoot (end) -->
282 </div><!-- .yui3-d3 (end) -->
283
284 </div><!-- #ft (end) -->
285 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/yui/yui.js"></script>
286 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/ooop/ooop.js"></script>
287 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/dom/dom.js"></script>
288 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/event-custom/event-custom.js"></script>
289 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/event/event.js"></script>
290 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/pluginhost/pluginhost.js"></script>
291 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/node/node.js"></script>
292 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/classnamemanager/classnamemanager.js"></script>
293 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/attribute/attribute.js"></script>
294 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/base/base.js"></script>
295 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/plugin/plugin.js"></script>
296 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/event-simulate/event-simulate.js"></script>
297 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/node/node-event-simulate.js"></script>
298 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/node-focusmanager/node-focusmanager.js"></script>
299 <script type="text/javascript" src="http://josephj.com:50019/yui/3.1.1/substitute/substitute.js"></script>
300 <script type="text/javascript" src="http://josephj.com:50019/core.js"></script>
301 <script type="text/javascript" src="http://josephj.com:50019/_photo_viewer.js"></script>
302 <script type="text/javascript" src="http://josephj.com:50019/_photo_filter.js"></script>
303 <script type="text/javascript" src="http://josephj.com:50019/_photo_list.js"></script>
304 </body>
305 </html>
306
```

# Mini Tool



<http://www.flickr.com/photos/prettypony/2644225789/>

將代碼**縮小**  
多個 CSS/JS 檔案**合併**的  
開發環境工具

模組檔再多也不需擔心效能的問題

# Mini 的 XML 設定檔，把剛剛的設定移往這吧

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <mini xmlns:xi="http://www.w3.org/2001/XInclude">
3
4 <path type="css">$DEV_ROOT/static</path>
5 <path type="js">$DEV_ROOT/static</path>
6
7 <module id="yui-css">
8   <file type="css" src="yui/3.1.1/cssreset/reset.css" />
9   <file type="css" src="yui/3.1.1/cssfonts/fonts.css" />
10  <file type="css" src="yui/3.1.1/cssgrids/grids.css" />
11 </module>
12
13 <module id="framework">
31
32 <module id="index">
33   <require module="yui-css" />
34   <require module="framework" />
35   <file type="css" src="core.css" />
36   <file type="css" src="mod.css" />
37   <file type="css" src="_masthead.css" />
38   <file type="css" src="_photo_list.css" />
39   <file type="css" src="_photo_filter.css" />
40   <file type="css" src="_photo_viewer.css" />
41   <file type="css" src="_introduction.css" />
42   <file type="js" src="_photo_viewer.js" />
43   <file type="js" src="_photo_filter.js" />
44   <file type="js" src="_photo_list.js" />
45 </module>
46
47 </mini>
```

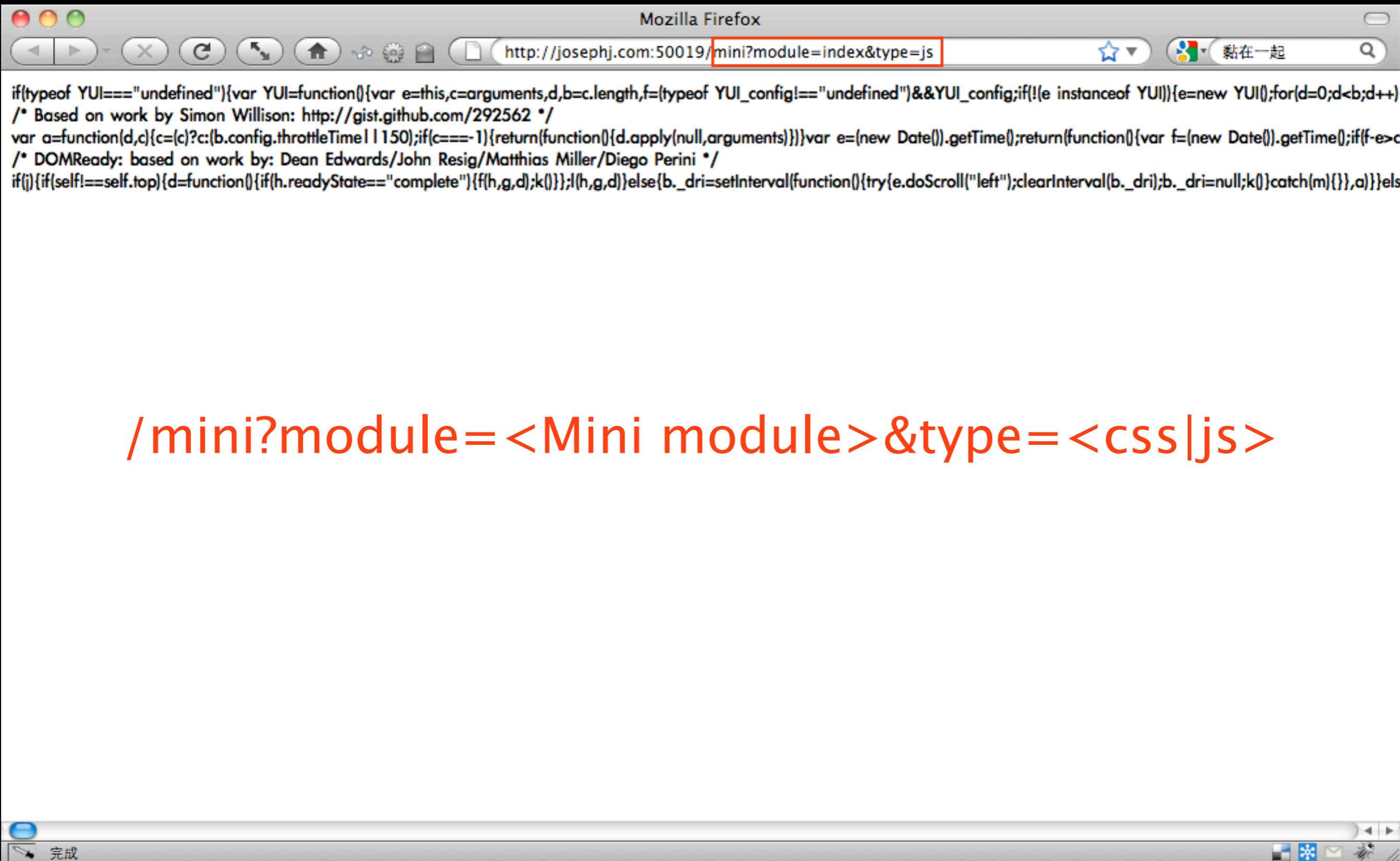
尋找檔案的路徑、可定義多個。  
\$DEV\_ROOT 是在設定 Apache VirtualHost 的環境變數

每個 Mini module 有自己的 CSS 與 JS 檔

module 可以去加入其他的 module，也可以 exclude 某個檔案

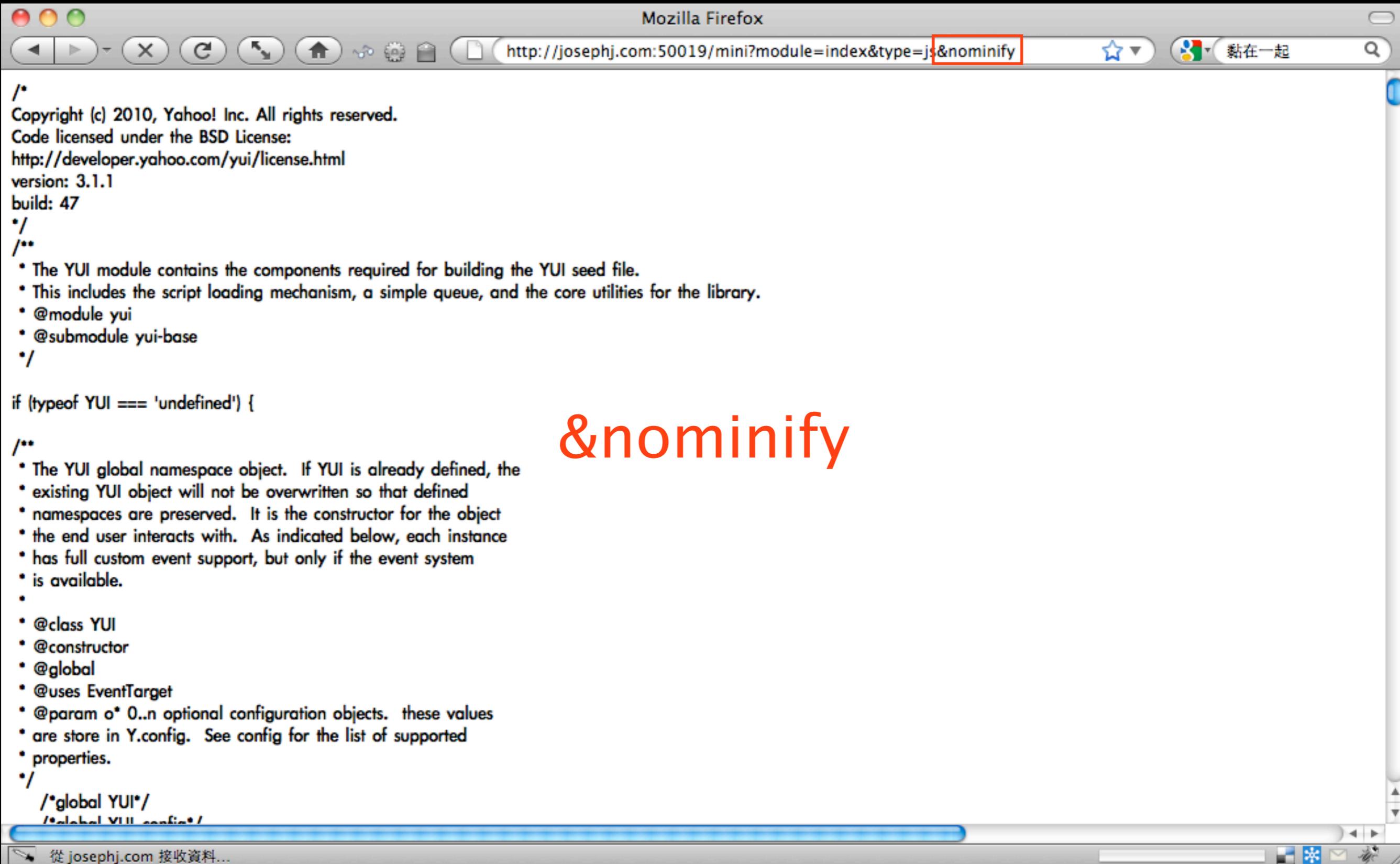
index 的 CSS/JS 模組檔案

# 经过 Mini 合并及压缩的内容：跟線上一致



`/mini?module=<Mini module>&type=<css|js>`

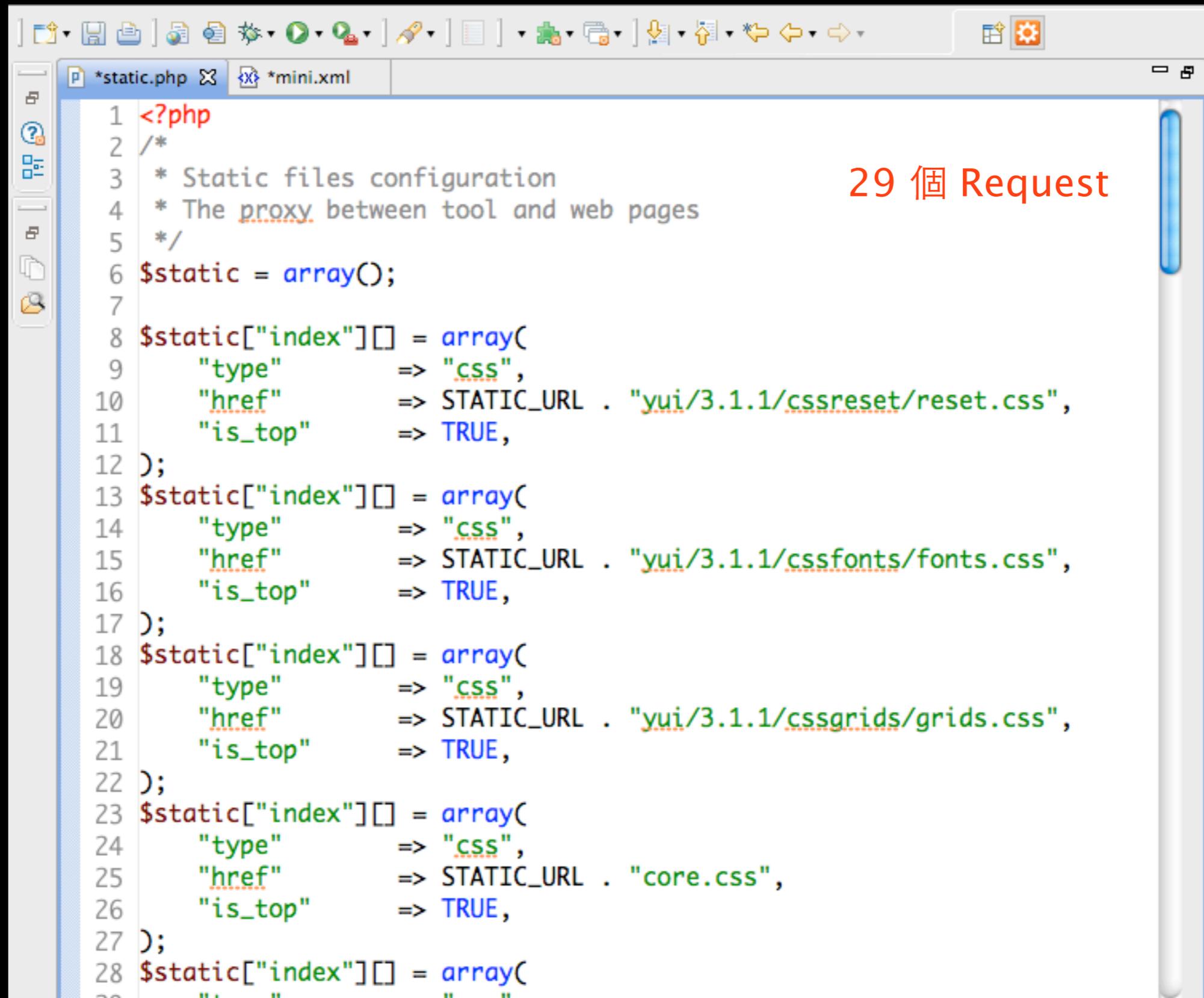
# 也可以只觀看合併但未壓縮的內容



The screenshot shows a Mozilla Firefox browser window with the address bar containing the URL `http://josephj.com:50019/mini?module=index&type=js&nomify`. The `&nomify` part of the URL is highlighted with a red box. The main content area displays the source code of a JavaScript file, which includes a copyright notice for Yahoo! Inc. (2010) and a list of comments describing the YUI module. The text `&nomify` is overlaid in large red font on the right side of the page.

```
/*  
Copyright (c) 2010, Yahoo! Inc. All rights reserved.  
Code licensed under the BSD License:  
http://developer.yahoo.com/yui/license.html  
version: 3.1.1  
build: 47  
*/  
/**  
 * The YUI module contains the components required for building the YUI seed file.  
 * This includes the script loading mechanism, a simple queue, and the core utilities for the library.  
 * @module yui  
 * @submodule yui-base  
 */  
  
if (typeof YUI === 'undefined') {  
  
/**  
 * The YUI global namespace object. If YUI is already defined, the  
 * existing YUI object will not be overwritten so that defined  
 * namespaces are preserved. It is the constructor for the object  
 * the end user interacts with. As indicated below, each instance  
 * has full custom event support, but only if the event system  
 * is available.  
 *  
 * @class YUI  
 * @constructor  
 * @global  
 * @uses EventTarget  
 * @param o* 0..n optional configuration objects. these values  
 * are store in Y.config. See config for the list of supported  
 * properties.  
 */  
/*global YUI*/  
/*global YUI config*/  
}
```

# 這是原本又臭又長的 static.php



```
1 <?php
2 /*
3  * Static files configuration
4  * The proxy between tool and web pages
5  */
6 $static = array();
7
8 $static["index"][] = array(
9     "type"         => "css",
10    "href"          => STATIC_URL . "yui/3.1.1/cssreset/reset.css",
11    "is_top"        => TRUE,
12 );
13 $static["index"][] = array(
14     "type"         => "css",
15     "href"          => STATIC_URL . "yui/3.1.1/cssfonts/fonts.css",
16     "is_top"        => TRUE,
17 );
18 $static["index"][] = array(
19     "type"         => "css",
20     "href"          => STATIC_URL . "yui/3.1.1/cssgrids/grids.css",
21     "is_top"        => TRUE,
22 );
23 $static["index"][] = array(
24     "type"         => "css",
25     "href"          => STATIC_URL . "core.css",
26     "is_top"        => TRUE,
27 );
28 $static["index"][] = array(
29     "type"         => "css",
```

29 個 Request

# 網頁的 Request 量超級多 Orz...

Modular Development (Raw) @ webrebuild.org | josephj.com

http://josephj.com:50010/raw.php

Power by Y!YUI

## muchiii Modular Development

### Photo Viewer



### Photo Filter

Search:

Find Photos

### Flickr Photo List



主控台 HTML CSS Script DOM 網路

清除 Persist All HTML CSS JS XHR Images Flash

URL	Status	Domain	Size	Timeline
▶ GET yui.js	304 Not Modified	josephj.com:50019	96.9 KB	4.34s
▶ GET oop.js	304 Not Modified	josephj.com:50019	12.5 KB	4.39s
▶ GET dom.js	304 Not Modified	josephj.com:50019	84.7 KB	4.41s
▶ GET event-custom.js	304 Not Modified	josephj.com:50019	68.1 KB	4.54s
▶ GET event.js	304 Not Modified	josephj.com:50019	61.2 KB	4.55s
▶ GET pluginhost.js	304 Not Modified	josephj.com:50019	10 KB	4.57s
▶ GET node.js	304 Not Modified	josephj.com:50019	62.9 KB	4.58s
▶ GET classnamemanager.j	304 Not Modified	josephj.com:50019	2.2 KB	4.6s
▶ GET attribute.js	304 Not Modified	josephj.com:50019	45.1 KB	4.62s
▶ GET base.js	304 Not Modified	josephj.com:50019	31 KB	4.66s
▶ GET plugin.js	304 Not Modified	josephj.com:50019	7.8 KB	103ms
▶ GET event-simulate.js	304 Not Modified	josephj.com:50019	21.7 KB	97ms
▶ GET node-event-simulat	304 Not Modified	josephj.com:50019	793 B	112ms
▶ GET node-focusmanager	304 Not Modified	josephj.com:50019	23.9 KB	102ms

完成

# 精簡過後的 static.php... 哇！

```
1 <?php
2 /*
3  * Static files configuration
4  * The proxy between tool and web pages
5  */
6 $static = array();
7 $static["index"][] = array(
8     "type"      => "css",
9     "href"      => STATIC_URL . "mini?module=index&type=css",
10    "is_top"     => TRUE,
11 );
12 $static["index"][] = array(
13     "type"      => "js",
14     "src"       => STATIC_URL . "mini?module=index&type=js",
15     "is_top"    => TRUE,
16 );
17 ?>
18
```

~~29 個 Request~~  
2 個 Request

# 網頁的 Request 量大幅減少 (大拇指)

The screenshot shows a web browser window displaying a website titled "muchiii Modular Development". The browser's address bar shows the URL "http://josephj.com:50010/". The website content includes a "Photo Viewer" section with a large landscape image, a "Photo Filter" section with a search input and a "Find Photos" button, and a "Flickr Photo List" section with a row of small photo thumbnails. The browser's developer tools are open at the bottom, showing the "Network" tab. A table of network requests is visible, with a red box highlighting a specific request.

URL	Status	Domain	Size	Timeline
<a href="http://josephj.com:50019/mini?module=index&amp;type=js">http://josephj.com:50019/mini?module=index&amp;type=js</a>		j.com:50019	113.1 KB	4.29s
Request_Count			113.1 KB	4.29s (onload: 5.9s)

透過 Mini 的設定檔，可將原本各自獨立的 CSS/JavaScript 檔案**合併且最小化**，達到效能最佳最佳化的目的！

但需要注意 Mini 並不適用於正式環境、通常放置 Statif File 的 Web Server 不安裝 PHP、而就算有也會對效能造成影響。

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
<meta http-equiv="content-type" content="text/html; charset=utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=EmulateIE7" />
<title>Modular Development @ webrebuild.org | josephj.com</title>
<link type="text/css" rel="stylesheet" href="http://img.modev.josephj.com/min_index_abf2a38f11b42c992888552d6fe3408c.css" media="all">
<script type="text/javascript" src="http://img.modev.josephj.com/min_index_58f8402b3d9ca4ed2dbca1e1179ad18b.js"></script>
</head>
```

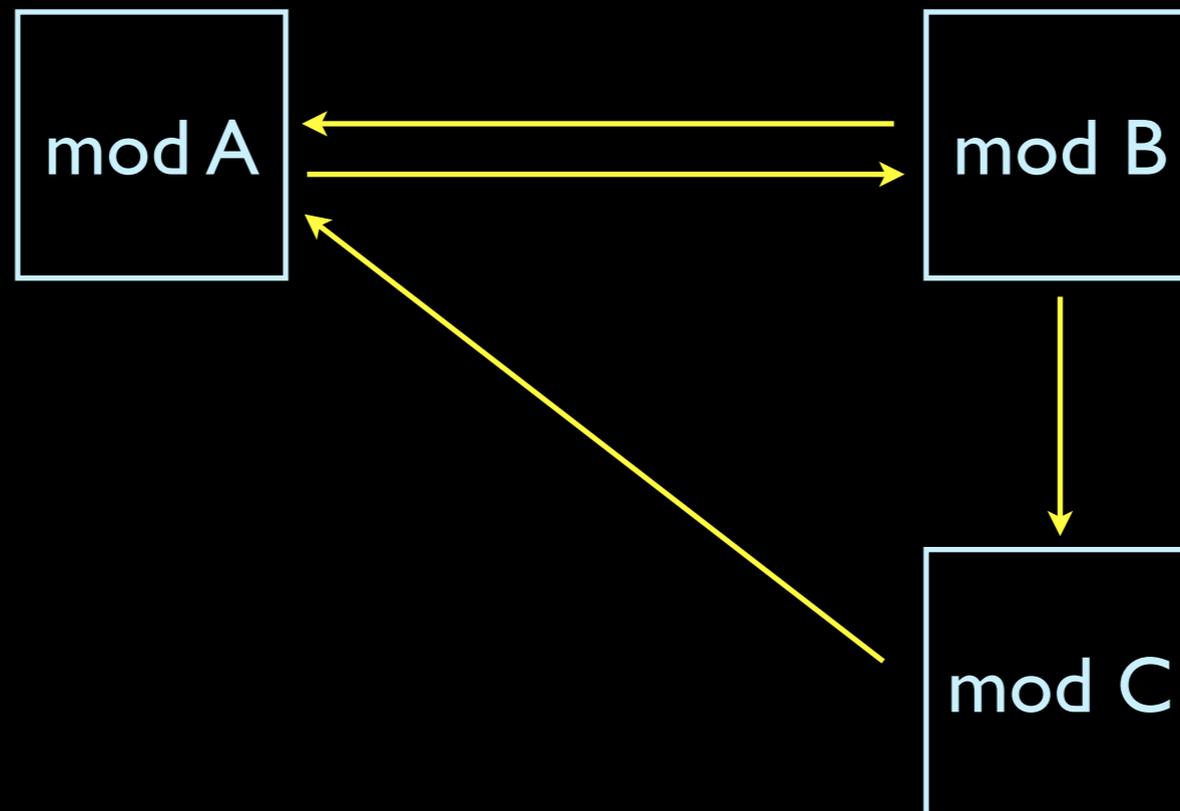
線上的 **CSS/JS** 在 **Deploy** 後變成單一檔

\* 用 `md5_file()` 檢查檔案內容產生的 stamping  
可減少 CDN 上有重複內容的檔案

# 跨模組溝通的問題

# JavaScript 跨模組溝通

JavaScript 若以模組為單位區分  
通常是獨立的 Function 或 Object  
有系統的跨模組溝通將會是大挑戰



## Photo Viewer



## Photo Filter

Search:

Flickr Photo 使用者送出表單 取得資料



## About JavaScript Platform

### 問題

- JavaScript 需處理模組之間溝通的問題，透過全域變數或方法傳遞會沒辦法彼此獨立。

### 概念

- 請參考 Nicholas Zakas 的 [Scalable JavaScript Architecture](#)

### 好處

- 每個人開發邏輯一致、花點時間了解概念都可以快速上手。

## Photo Viewer

```

window.mods["photo-show"] = {
  showPhoto : function (src) {
    ....
  }
}

```

九成的人會用全域變數解決問題

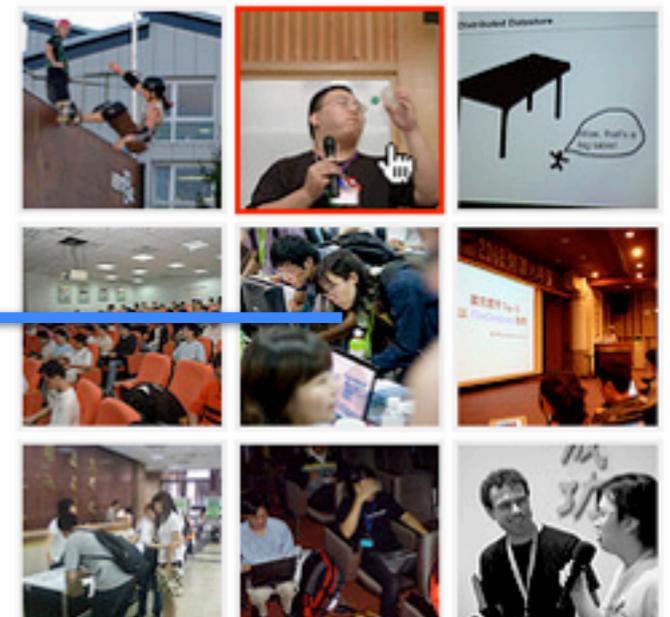


## Photo Filter

Search:

**Find Photos**

## Flickr Photo List



使用者點選

## About JavaScript Platform

### 問題

- JavaScript 需處理模組之間溝通的問題，透過全域變數

### 概念

- 請參考 Nicholas Zakas 的 [Scalable JavaScript Architec](#)

### 好處

- 每個人開發邏輯一致、花點時間了解概念都可以快速上手。

```

window.mods["photo-list"] = {
  init : function () {
    ...
    $("img").on("click", function(e) {
      window.mods["photo-show"].showPhoto(this.src);
    });
  }
}

```

當照片顯示模組沒有一起在同一頁時，列表模組就會出現 Bug!

我們都希望球員（**模組**）們扮演好自己的角色  
在球場（**頁面**）上合作有場好比賽（**功能**）



但是缺乏紀律（無架構、使用全域變數）  
最後必將造成（模組）之間的衝突

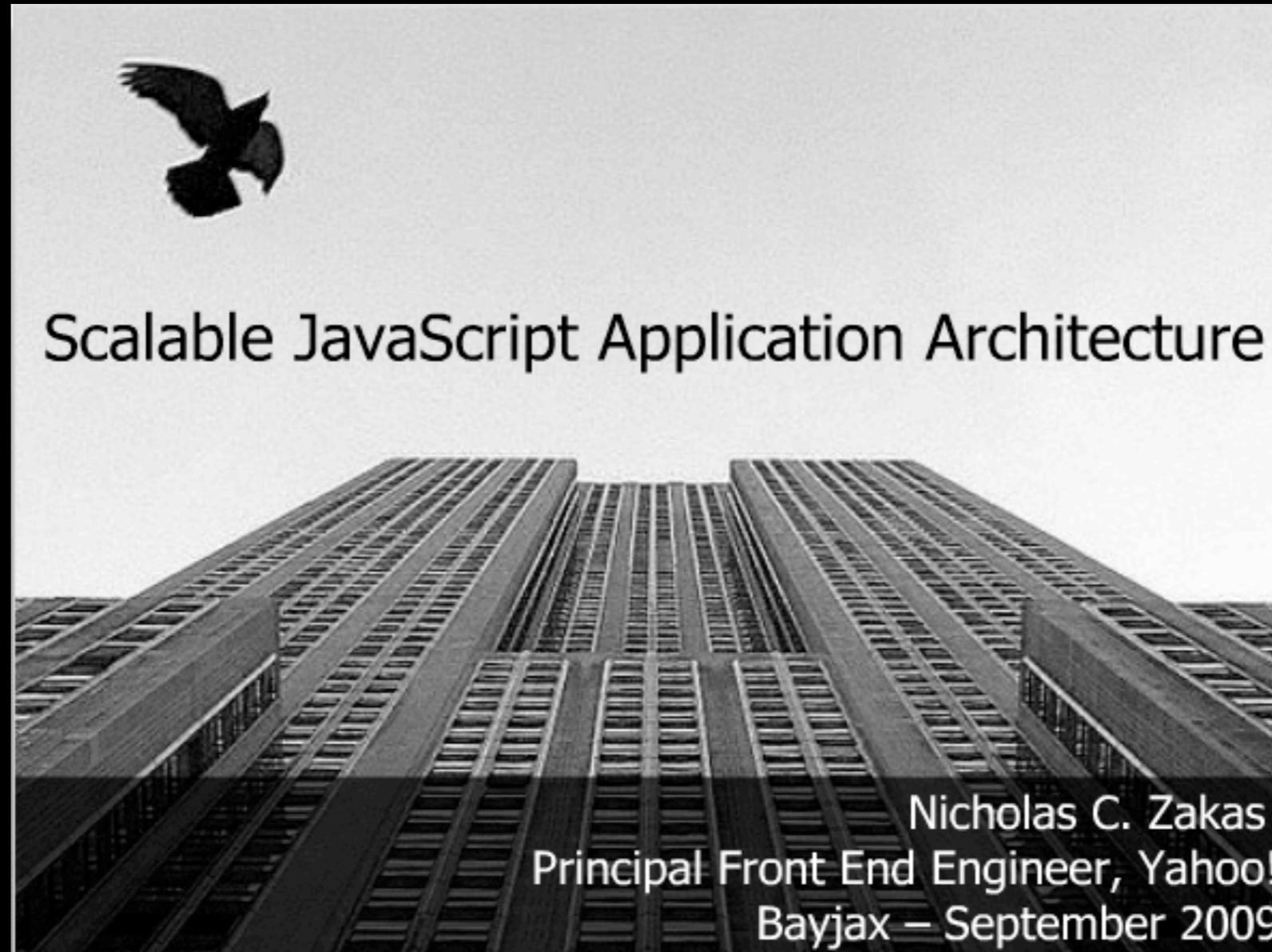


<http://www.flickr.com/photos/acaben/2477822120/>

# 雅虎首頁前端架構師 Nicholas Zakas 的啟發



# JavaScript 應用程式架構



<http://www.slideshare.net/nzakas/scalable-javascript-application-architecture>



## Loose Coupling 鬆散耦合 與其他模組的關係是鬆散的

[http://www.thesoftwaredevotional.com/images/coupling\\_train.jpg](http://www.thesoftwaredevotional.com/images/coupling_train.jpg)

當頁面上多了一個模組

它將不會破壞其他模組的結構、樣式與行為

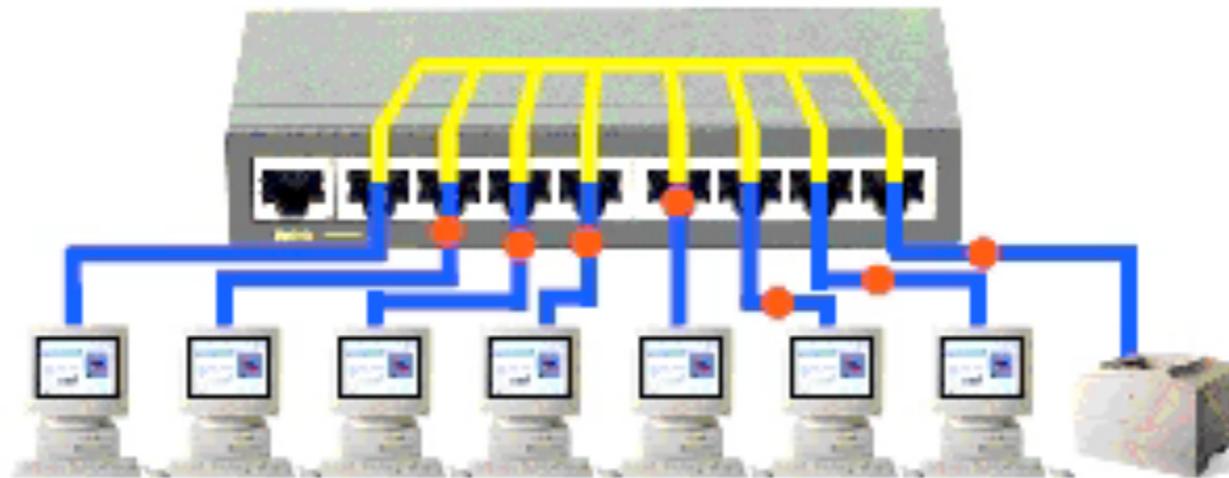
它將提供新的功能、讓其他的模組可以與它溝通互動

拿掉它時不需修改、也不會影響與其溝通模組的原有功能

# **Broadcast & Listen**

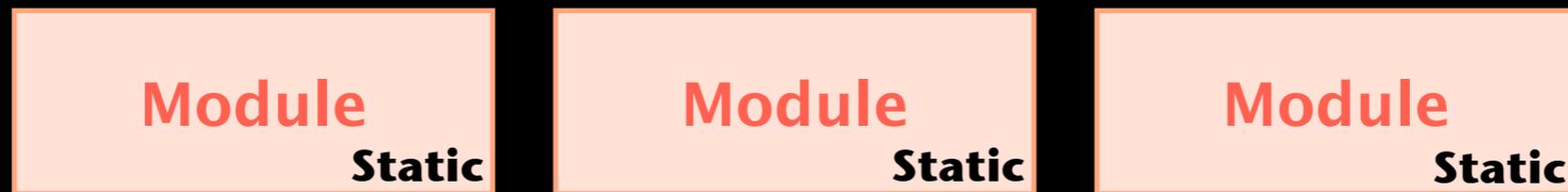
**模組溝通的方式類似 Hub 的概念**

# Hub



一台電腦出任何訊息，所有電腦都會收到  
但是其他電腦只處理它有興趣的訊息

# Module - 獨立運作的個體



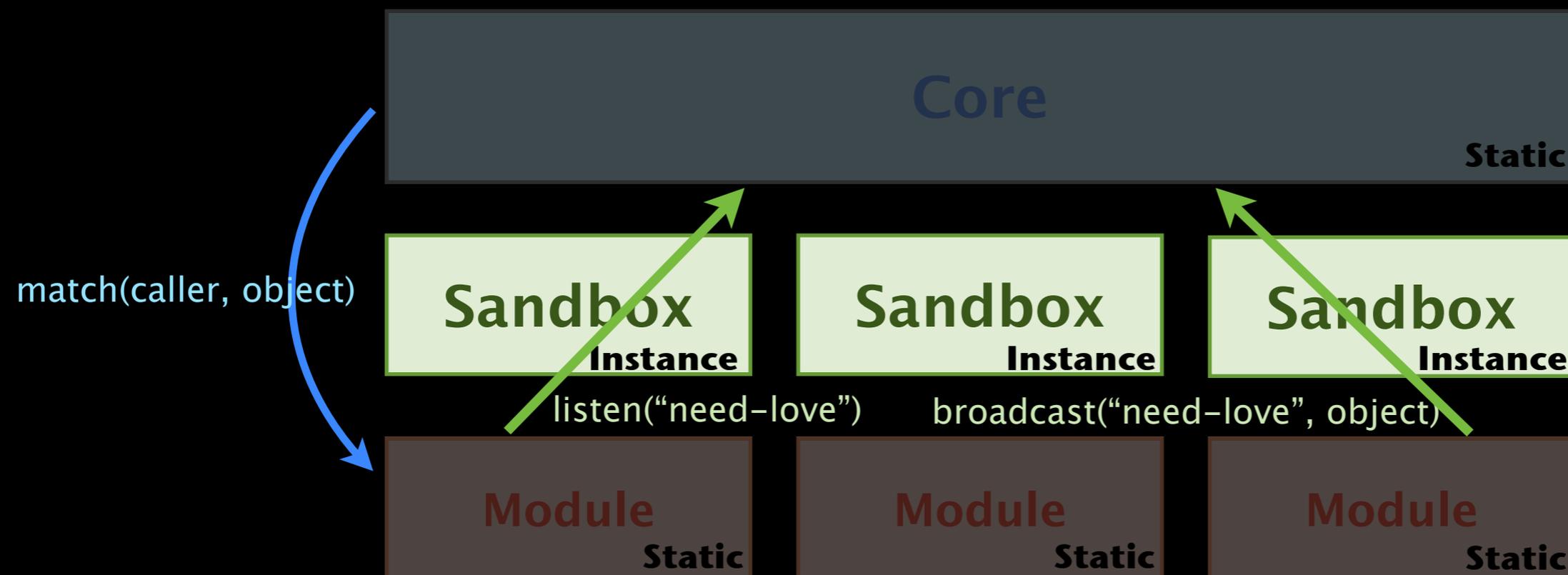
- 模組之間不知道彼此的存在
- 模組只在意自己本身的功能

# Core - 整個架構的核心



- 接受模組註冊 (register)
- 控制模組啟動關閉 (start, stop)
- 所有廣播訊息的傳遞與對應 (match)

# Sandbox - 模組 API 介面



- 以 Instance 的方式讓模組不受污染
- 模組與 Core 溝通都需要透過 Sandbox
- 模組只能使用 Sandbox 提供的方法
- 提供模組 broadcast 與 listen 功能

# \_photo\_list.js 照片列表模組



```
Y.Core.register("photo-list", {  
  // 此模組的初始化  
  init: function (api) {  
  
  },  
  // 此模組在頁面上 ContentReady  
  onviewload: function () {  
  
  },  
  // 此模組收到訊息時  
  onmessage: function (eventType, callerName, data) {  
  
  }  
});
```

# 範例 **Demo**

<http://josephj.com/project/javascript-platform-yui-demo>

# 模組架構 **Keep Going**

有了這樣的架構後  
可以依據需求加 Extension !  
替每個模組添增新 API Method

## Language Extension

# muchii JavaScript Modular Platform



## Focused Figure

香港與台灣的朋友  
大家好，我是大家  
最熟悉的豆花妹！



## Language Selector

Language:

Browser developer console showing network activity:

清除	Persist	All	HTML	CSS	JS	XHR	Images	Flash		
▶							200 OK	josephj.com	1.7 KB	43ms
▶							200 OK	josephj.com	1.9 KB	45ms
▶							200 OK	yui.yahooapis.com	7.1 KB	25ms
▶							200 OK	yui.yahooapis.com	61.6 KB	967ms
▶							200 OK	josephj.com	250 B	3ms

The entry `GET demo_zh-Hant.js` is highlighted with a red box.

YUI Intl 自動會將目前 JS 語系檔下載

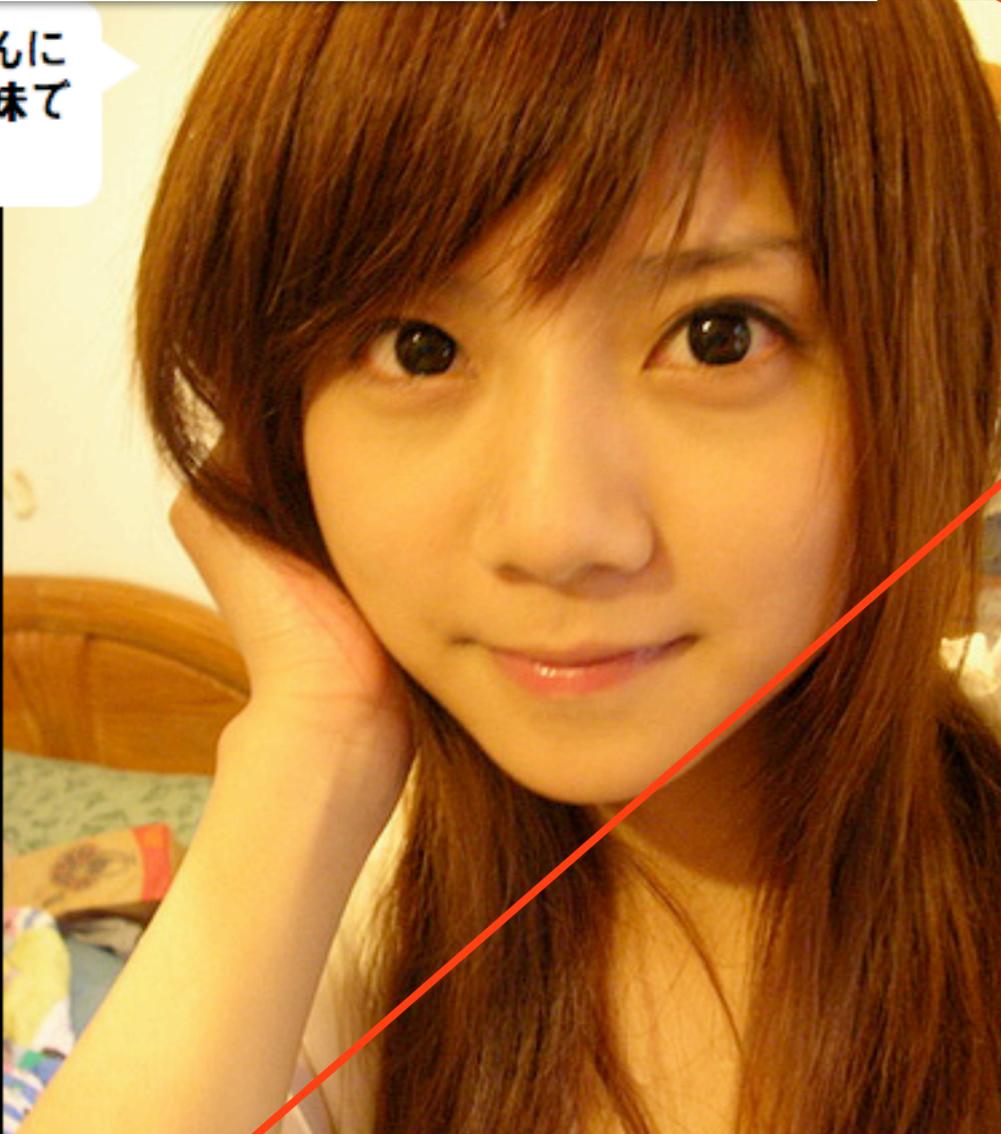
# muchii JavaScript Modular Platform



Foc

3. 原本就註冊了 `api.listen("lang-change")`  
因此直接再取得一次 `api.getTrans(<key>)` 即可;

日本の皆様 こんにちは！私は豆花妹です



## Language Selector

Language: Japanese

1. 執行 `api.setLang("jp-JP");`  
會自動取得日文語系檔

2. 新的語系檔被動態拉下來後  
執行 `api.broadcast("lang-change")`

主控台 HTML CSS Script DOM 網路

zh-Hant

清除 Persist All HTML CSS JS XHR Images Flash

URL	Status	Domain	Size	Timeline
GET demo_jp-JP.js	200 OK	josephj.com	360 B	57ms
Request_Count			360 B	57ms

順道一提！

## PHP 與 JavaScript 的 Language Key 都是基於模組為單位去定義的

```
<?php  
$trans = get_trans("lang-show_title", "Focus Figure", tokens);  
?>
```

```
<script>  
var trans =  
  api.getTrans(  
    "lang-show_message",  
    "Nice to see you, I am Tofu flower girl",  
    tokens);  
</script>
```

好處：

模組化開發讓各別團隊成員能專注在單一功能的開發上、不需要煩惱整頁複雜的架構、或共用 API 後面複雜的邏輯 (剛的多國語系不好搞)。另外在測試上也可以以模組為單位單獨測試、符合 Unit Test 的精神

# 自動化：讓犯錯變得困難

Make it difficult to make mistakes

到從零開始公司的好處就是先搶先贏

我第一個來的！  
老子說了算！

muchiii

josephi

<http://www.flickr.com/photos/eye1/3184963395/>

by Ivan Mlinaric

一開始我就制定了許多 Guidelines

# Front-end Engineering

## New Hire

- [F2E Software Installation](#)
- [⇒ Accessibility](#)
- [⇒ Yahoo! F2E Guidelines](#) : Our basic ideas about front-end development
- [⇒ Introduction to Yahoo! User Interface Library 3](#) : Our JavaScript framework
- About muchiiiBar

## Conventions

- [HTML Code Conventions](#)
- [HTML Code Conventions For CodeIgniter View](#)
- [HTML Code Conventions For CodeIgniter Layout](#)
- [CSS Code Conventions](#)
- [JavaScript Code Conventions](#)
- [File Naming Conventions](#)
- [Image Optimization Rules](#)
- [JavaScript L10N](#)

在前家公司時最難忘的話  
一位主管對我們的品質提出質疑  
「As a team，你們的價值究竟在哪裡？」

軟體開發團隊必須要有**規範的制定**  
**基本框架的設計**  
代碼品質才能得到提升

任何規範或框架都是開發人員的限制  
但若不設限制的情況就會像...

煙囪上架屋頂？  
這就是疊床架屋



Winchester House

<http://www.crockford.com/codecamp/quality.ppt>

有 Guideline 不錯，但是老祖宗形容地很貼切：「積習難改」、「陽奉陰違」，每個人對於自己的**習慣與理念有所不同**，有意無意地 Guideline 放到一旁。

久而久之，當初大家同意的 Guideline 就會被遺忘、整個團隊的程式品質開始陷入混亂...

**Code Review** 是一個好方法，但得花許多時間、沒辦法常辦，即使發現了問題，也只能用**妥協**的方式做小部分的修改...

軟體開發不應該存在的妥協

電腦的世界不就是 0 跟 1 嗎？

軟體開發不應該存在的妥協

電腦的世界不就是 0 跟 1 嗎？

# 1. Commit/Push 時的作法

每次 Commit/Push 以 Hook 對變動檔案做檢查

- 執行 **JSLint** :  
必須符合 JavaScript Good Parts
- 執行 **PHP CodeSniffer** :  
可自訂 Code Convention 檢查 PHP、JavaScript、CSS
- 執行 **OptImg** :  
提示圖檔尚未最佳化、是否要用程式自動最佳化？

若沒通過檢查就無法 Commit !

```
Default (115,35)
[joseph_chiang@devm1:/home/dev/joseph_chiang/muchiii/static/i]$ jslint toolbar.js
/home/dev/joseph_chiang/muchiii/static/i/toolbar.js

Lint at line 60 character 17: document.write can be a form of eval.
document.write([

Lint at line 67 character 17: document.write can be a form of eval.
document.write('<script id="' + scriptId + '" src="' + routerUri + '"></script>');

Lint at line 91 character 13: The body of a for in should be wrapped in an if statement to filter unwanted properties from the prototype.
for (i in iframeAttrs) {

Lint at line 135 character 119: Insecure '^'.
obj.outerHTML = html.replace(/<param\s+name\s*=\s*('|")flashvars('|")\s+value\s*=\s*('|")[^"]*('|")\s*\V?\s*>/i, "<param name='flashvars' value='" + child[j].getAttribute('value') + "' />");

Lint at line 167 character 17: document.write can be a form of eval.
document.write([

Lint at line 193 character 17: The body of a for in should be wrapped in an if statement to filter unwanted properties from the prototype.
for (i in scriptEls) {

Lint at line 232 character 13: The body of a for in should be wrapped in an if statement to filter unwanted properties from the prototype.
for (i in bodyStyles) {

Lint at line 250 character 13: The body of a for in should be wrapped in an if statement to filter unwanted properties from the prototype.
for (i in wrapperStyles) {

[joseph_chiang@devm1:/home/dev/joseph_chiang/muchiii/static/i]$ |
[0]jslint @ devm1 0.00 0.00 0.00 | 2010-07-03
0$ jslint | 13:59:34
```

document.write  
can be form of eval

# JSLint

```
Default (99,34)
[joseph_chiang@devm1:~dev/joseph_chiang/muchiii/static/lab]$ phpcs dogfood.css

FILE: /home/joseph_chiang/muchiii/static/lab/dogfood.css
-----
FOUND 21 ERROR(S) AND 0 WARNING(S) AFFECTING 19 LINE(S)
-----
 2 | ERROR | Expected 1 space after colon in style definition; 0 found
 2 | ERROR | CSS colours must use shorthand if available; expected #fff but
   |       | found #ffffff
 3 | ERROR | Expected 1 space after colon in style definition; 0 found
 6 | ERROR | Expected 1 space after colon in style definition; 0 found
 7 | ERROR | Expected 1 space after colon in style definition; 0 found
 8 | ERROR | Expected 1 space after colon in style definition; 0 found
12 | ERROR | Expected 1 space after colon in style definition; 0 found
15 | ERROR | Expected 1 space after colon in style definition; 0 found
16 | ERROR | Expected 1 space after colon in style definition; 0 found
18 | ERROR | Whitespace found at end of line
20 | ERROR | Expected 1 space after colon in style definition; 0 found
20 | ERROR | CSS colours must be defined in lowercase; expected #16387c but
   |       | found #16387C
21 | ERROR | Expected 1 space after colon in style definition; 0 found
24 | ERROR | Expected 1 space after colon in style definition; 0 found
26 | ERROR | Expected 1 space after colon in style definition; 0 found
27 | ERROR | Expected 1 space after colon in style definition; 0 found
28 | ERROR | Expected 1 space after colon in style definition; 0 found
29 | ERROR | Expected 1 space after colon in style definition; 0 found
32 | ERROR | Expected 1 space after colon in style definition; 0 found
33 | ERROR | Expected 1 space after colon in style definition; 0 found
34 | ERROR | Expected 1 space after colon in style definition; 0 found
-----

[joseph_chiang@devm1:~dev/joseph_chiang/muchiii/static/lab]$ |
[1]phpcs @ devm1 0.00 0.00 0.00 | 2010-07-03
0-$ jslint 1$ phpcs | 14:10:03
```

.rule {  
 attr: value;  
}

# PHP CodeSniffer (a.k.a phpcs)



敝公司的 Visual Designer

只負責製作 PSD 檔案

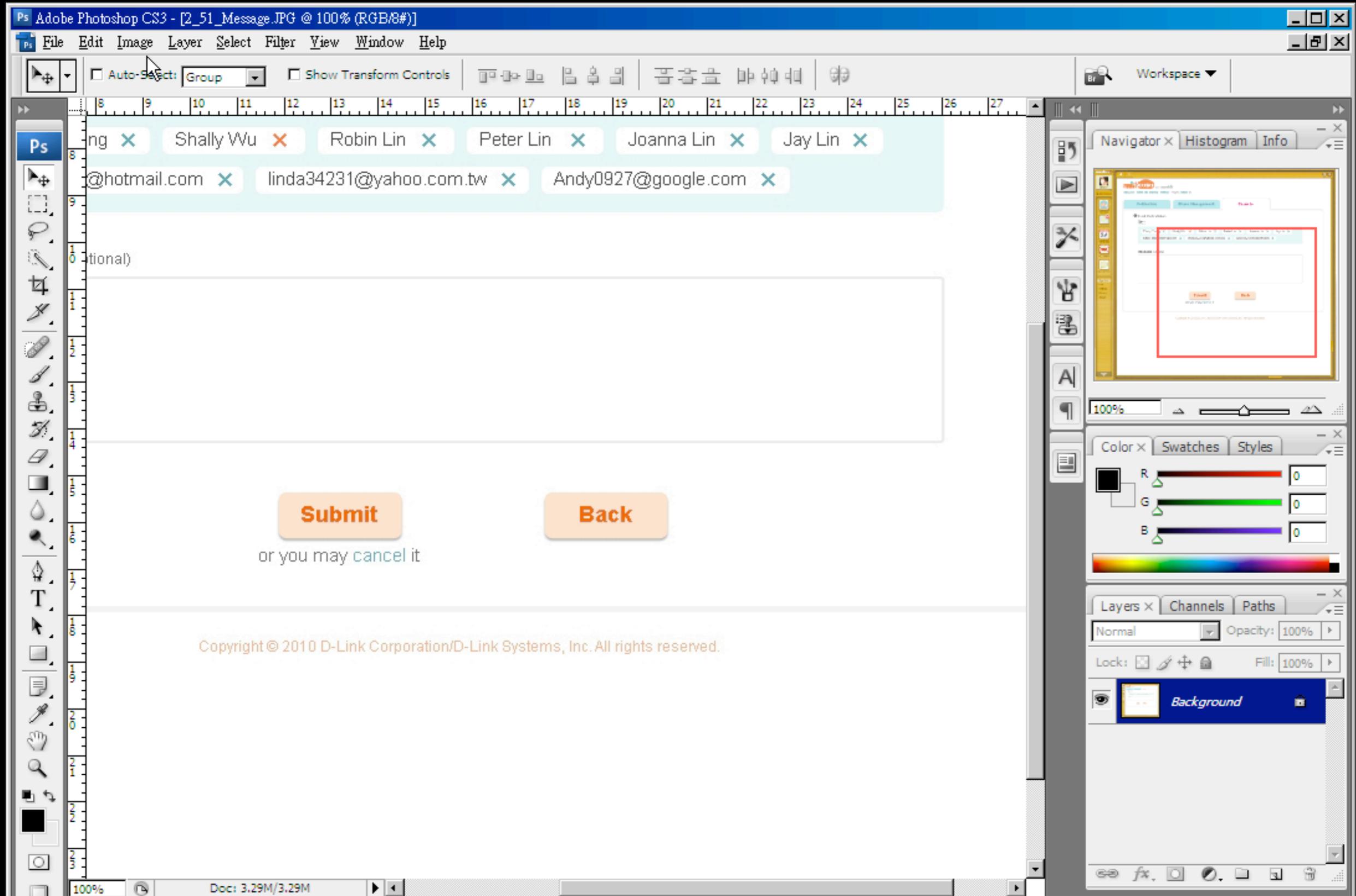
F2E 得自己處理圖檔最佳化的問題



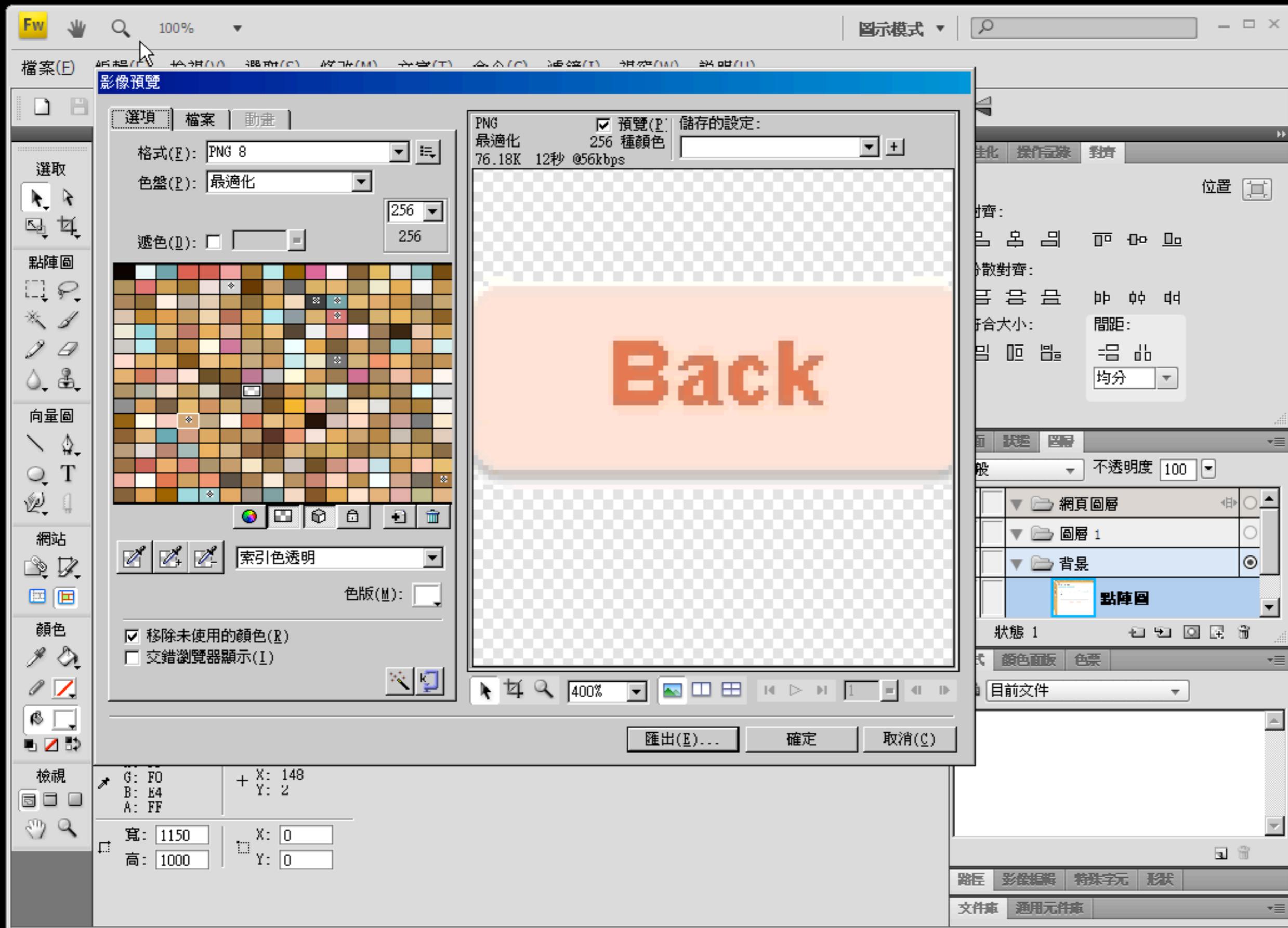
PNG8 是目前公認在 Web 最好的格式

但是只有 Fireworks 支援轉換  
Alpha Transparency (索引色透明)

# 每次切圖都超痛苦，先用 Photoshop 打開 PSD



# 再開 Firework 轉換成索引色透明



生命就這樣浪費在沒有意義的事物上了

# OptImg: 一個指令按下去幫你搞定

```
Default (113,35)
37. toggle_shrinking.png : save 322 bytes
38. rc_grip_rb.png.bak : save 21 bytes
39. ico_telephone.png : save 53 bytes
40. btn_close.png : save 111 bytes
41. sep_vertical.png : save 42917 bytes
42. ico_plurk.jpg : optimized already!
43. bg_total_text.png : save 116 bytes
44. bg_total_left.png : save 145 bytes
45. ico_mydlink.png : save 44984 bytes
46. btn_search.png : save 114 bytes
47. bg_pane_left.png : save 21 bytes
48. logo_minimize.png : save 44723 bytes
49. ico_game.png : save 46747 bytes
50. lock_open.png : save 21 bytes
51. ico_phone.png : save 188 bytes
52. ico_information.png : save 129 bytes
53. rc_grip_rt.png.bak : save 21 bytes
54. bg_pane_right.png : save 110 bytes
55. lock.png : save 21 bytes
56. ico_dlink.jpg : optimized already!
57. bg_pane_top.png : save 22 bytes
58. btn_expand.png : save 155 bytes
59. logo_dlink.png : save 90 bytes
60. ico_facebook.png : save 45233 bytes
61. bg_toggle.png : save 323 bytes
62. bg_profile_photo.png : save 21 bytes
63. blank.gif : optimized already!
64. sep.png : save 65843 bytes
65. bg_content.png : save 124550 bytes
66. ico_warning.png : save 130 bytes

Total saved: 873599 bytes
[joseph_chiang@devm1:/home/dev/joseph_chiang/muchiii/static/lab]$ optimg --report .
[2]optimg @ devm1 0.27 0.19 0.08 | 2010-07-03
0$ jslint 1-$ phpcs 2$ optimg | 14:21:25
```

圖檔占網站總頻寬的 46%

## Image Optimization

# 不修改就不给 Commit

```

josephj.com (99,44)
Default  josephj.com

[josephj@cockatoo:modev/static]$ git commit -am 'code check test' (07-16 19:28)

COMMIT FAILED!
=====
JSLint Error report
=====
8cb14e387d723da33a8d12d3908a89f7c6f2ff53
static/core.js
Lint at line 1 character 9: 'YUI' is not defined.
var Y = YUI();

Lint at line 6 character 6: Missing semicolon.
}

Lint at line 7 character 9: Too many var statements.
var proto = Sandbox.prototype;

Lint at line 26 character 9: Too many var statements.
var listeners = {};

Lint at line 27 character 9: Too many var statements.
var Core = {

Lint at line 30 character 22: Too many var statements.
for (var i in listeners) {

Lint at line 31 character 26: Too many var statements.
for (var x in listeners[i]) {

Lint at line 46 character 59: Unexpected use of '++'.
for (var i = 0, j = listeners.length; i < j; i++) {

Lint at line 50 character 21: Unreachable 'break' after 'return'.
break;
=====

Please fix your code and commit again.

[josephj@cockatoo:modev/static]$ | (07-16 19:28)
[?]fnontend @ cockatoo 0 12 0 04 0 01 | 2010-07-16
```

# 1. Commit/Push 時的作法

唯有將檢查放入每天的開發流程中  
Guideline 才能真正地被落實

## 2. Packaging 時的作法

我深深認為...

F2E 寫程式最重要的就是 **Maintenance**

你的程式必須很容易地被其它成員看懂

其它像是 Performance 或 Security 的議題

大多有機會交由工具或架構處理掉

打包 (Packaging) 就是一個很好的機會

## 2. Packaging 時的作法

- 執行 JS/CSS 的最小化與合併

例如：`miii_index_d38eba51136ed0bf6d159770853b393b.css`

是首頁所有 CSS 合併最小化產生的

- 執行 Static 設定檔的取代

是首頁所有 CSS 合併最小化產生的

- 執行 CSS 背景圖檔取代為 DataURI/MHTML

不再使用超手工的 CSS Sprites

```
Default (113,35)
[joseph_chiang@devm1:~]$ pkg_create index development /tmp/muchiii/

Create working directory...(done)
Minify CSS & JavaScript file...(done)
Update static configuration...(done)
Modify URLs in static and configuration directories...(done)
Packing necessary files...(done)

Packages are generated under "/tmp/muchiii/development_josephchiang_201007031426/pkg/" directory
You can use the following command to install packages:
=====
pkg_deploy /tmp/muchiii/development_josephchiang_201007031426/pkg/index.tar.gz devm1.corp.muchiii.com;pkg_deploy
/tmp/muchiii/development_josephchiang_201007031426/pkg/static.tar.gz devm1.corp.muchiii.com;
=====

[joseph_chiang@devm1:~]$ pkg_deploy /tmp/muchiii/development_josephchiang_201007031426/pkg/index.tar.gz devm1.corp.muchiii.com;
Install packages to devm1.corp.muchiii.com:22...
Connection to devm1.corp.muchiii.com closed.
(done)
Install packages to devm1.corp.muchiii.com:22...
Connection to devm1.corp.muchiii.com closed.
(done)
[joseph_chiang@devm1:~]$ |
```

pkg\_create

合併 JS/CSS 檔案與最小化

檔案合併後、檔名也會改變 (不再使用 Mini 工具)

pkg\_deploy

[3]package @ devm1

0.25 0.28 0.14

2010-07-03

0\$ jslint 1-\$ phpcs 2\$ optimg 3\$ package

| 14:27:27

# 打包 pkg\_create & pkg\_deploy

## 2. Packaging 時的作法

版本控制中不應該有無法維護的檔案，例如：

合併或最小化的 CSS/JS

CSS Sprites 圖檔

DataURI/MHTML 字串

就讓打包流程去產生、以其它環境做驗證

以自動化的方式、讓妥協與錯誤不會重複地發生

# Review

- 模組化開發

將每個模組檔案獨立、用工具減少 Performance 爭議、用架構讓模組相互溝通。

- 自動化開發

透過流程整合自動化機制、減少每天浪費的時間、及每個人犯錯的機率。

# 相關連結

- **JavaScript Module Platform @ GitHub**  
<http://github.com/josephj/javascript-platform-yui>
- **Online Demo**  
<http://josephj.com/project/javascript-platform-yui-demo>
- **Demo Source @ GitHub**  
<http://github.com/josephj/javascript-plaform-yui-demo>

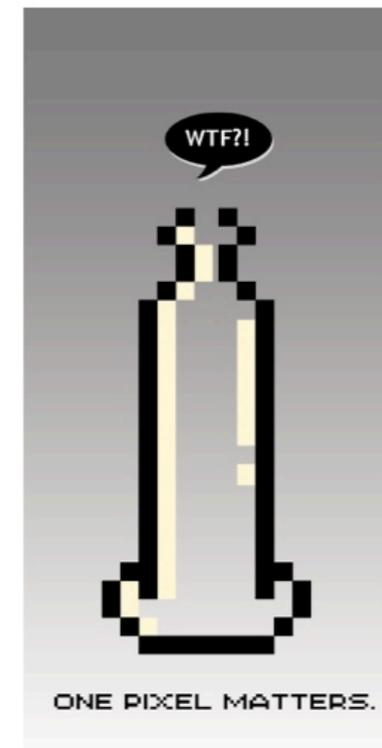
Finally... 有貼紙可以拿喔！



Front-end Engineering  
Olympic Game

facebook-addicted

MEETING ENGINEER



BUGS

FREE



MEETING ENGINEER

FREE





<http://www.flickr.com/photos/phploveme/2847931240/>

Thank You  
so

**muchiii**  
Powered by D-Link

Any Questions?

josephj6802@gmail.com