

# YUI Library

前端工程開發實務訓練 (4/8)

講師：蔣定宇 / josephj



# Why YUI?

- 幫助了解 JavaScript。
- 一致的物件導向基礎。
- 自訂事件介面。
- 完整 Software Stacks。
- 模組與 Loader 架構。
- 與 Node.js 整合。
- 非社群、有公司資源開發跟品質管理。

# Why not YUI?

- 知名度跟普及率較差。
- 語法較複雜、不直覺、門檻較高。
- 與 Yahoo! 有關聯。

整體來說 YUI 提供了團隊開發的重要基礎  
物件導向、模組化、Loader、自訂事件等  
對程式碼的一致性、擴充幫助非常大

搞懂架構之後，用任何東西都會比別人有 Sense!

# YUI 的架構

模組 / 載入器

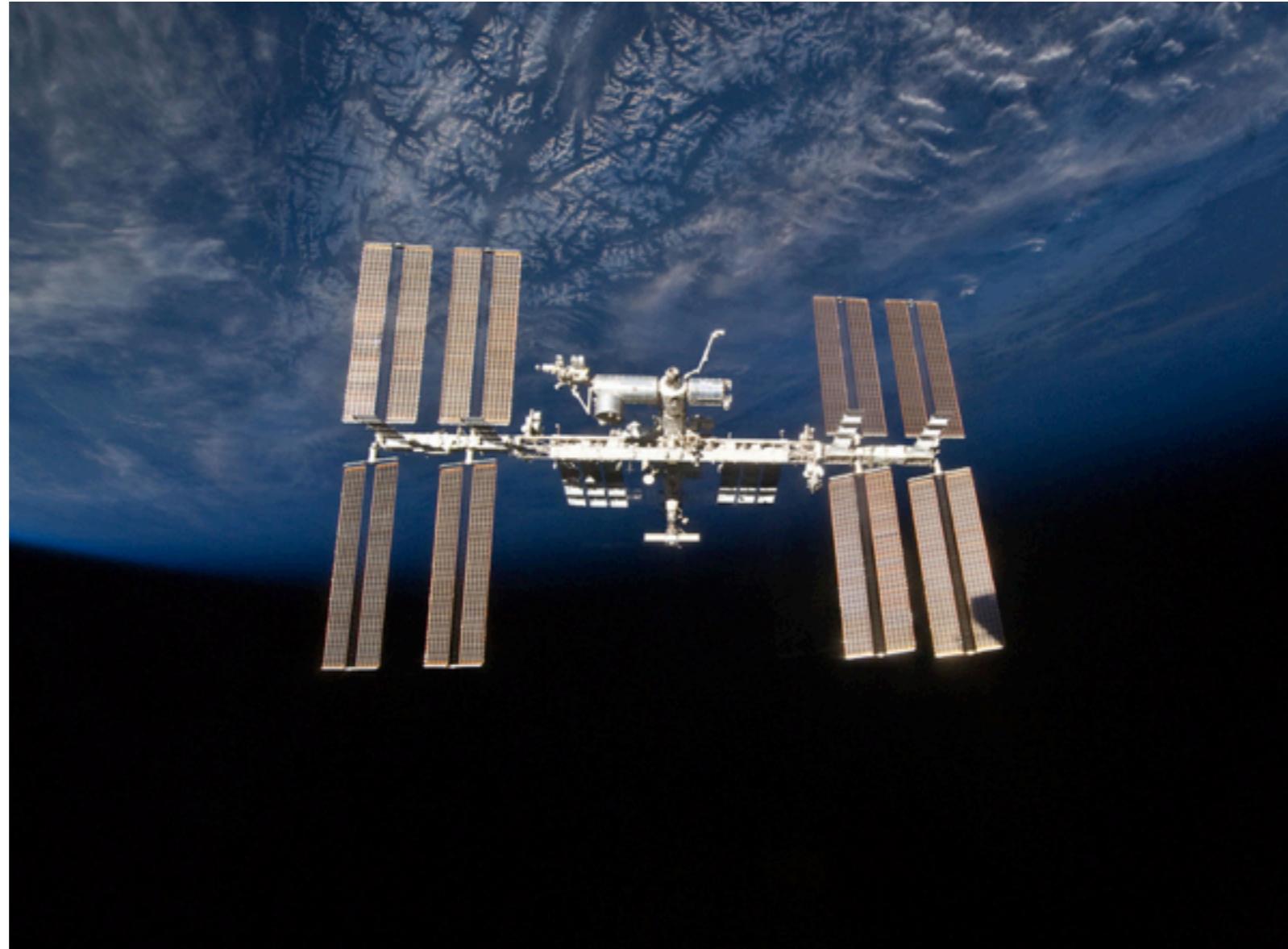


# 模組化 Modularization

將功能分門別類、要用的時候挑選組合

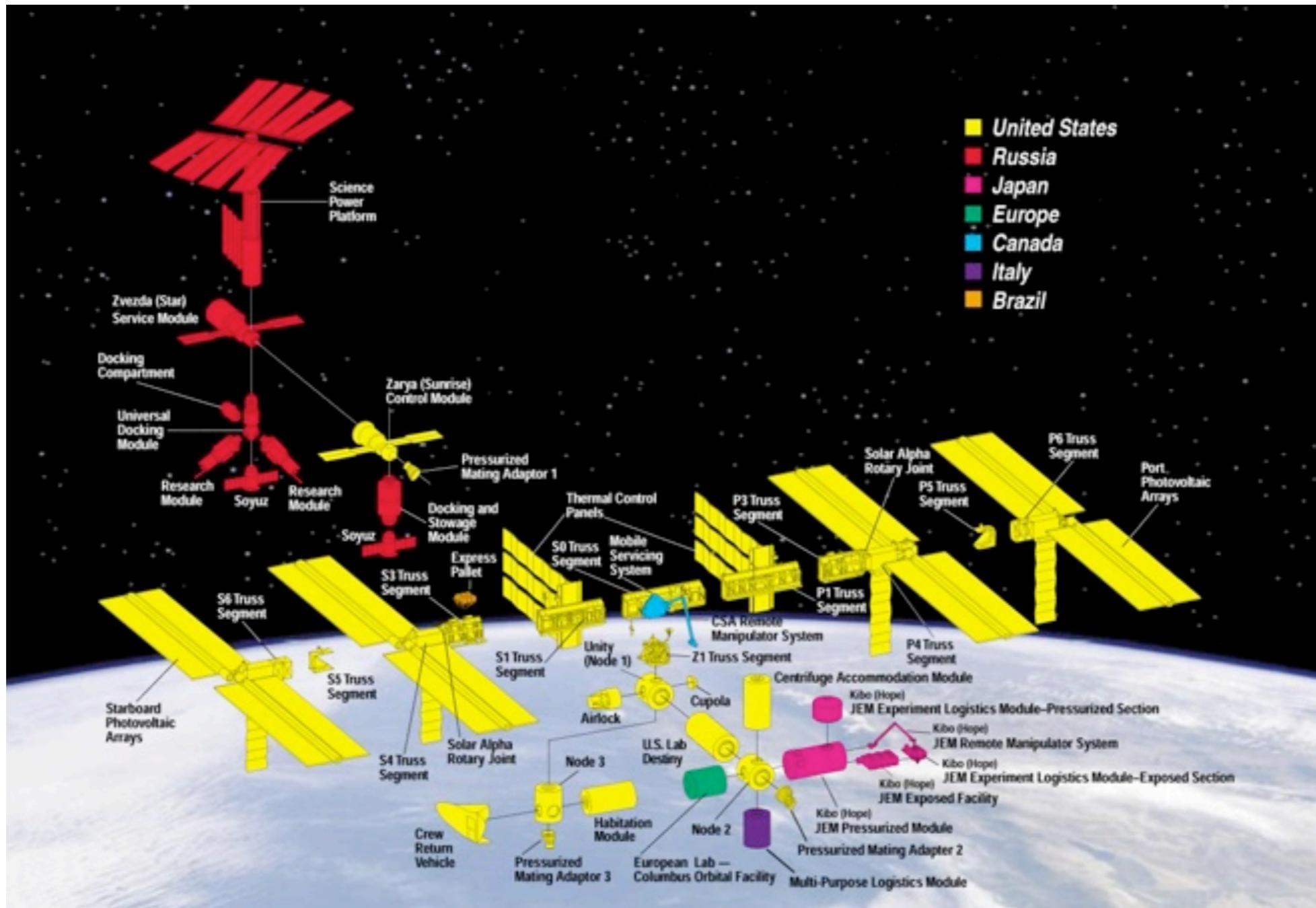
<http://www.flickr.com/photos/bdesham/2432400623>

# NASA 國際太空站



除了樂高積木、國際太空站也是模組化的好範例

# NASA 國際太空站



由 7 個不同國家所提供的 50 多個模組

# 模組化的好處

舉凡建築、航太、機械、硬體到軟體

在任何分工精細的工程領域

模組化開發是必然趨勢

「採用既有模組、避免重新打造輪子」

# 最基本的 JS 模組化

```
<script src="jquery-1.7.2.js"></script>          // 核心函式庫  
<script src="jquery.ui.core.js"></script>        // 模組 1  
<script src="jquery.ui.widget.js"></script>       // 模組 2  
<script src="jquery.ui.accordion.js"></script>     // 模組 3
```

開發小網站這樣做沒問題，但稍微複雜的就不適合了  
全域變數、維護性、前後順序、模組效率問題

# YUI 模組架構

當覺得程式太多、需重複使用，切分模組會是很棒的作法  
 對於使用者來說、不用為了一個小功能把一堆東西拉近來

所有 YUI 模組檔案皆必須以這樣的方式包覆：

```
editor.js YUI.add("editor", function (Y) {
```

// 上方的參數 Y 代表了 YUI 的 Instance

```
function Editor() {
```

// 這是物件的建構式

```
}
```

// 將建構式提昇到 Global

```
Y.Editor = Editor;
```

Why callback?

放到頁面上時不會立即執行、  
 等到要用時再執行即可。好處  
 是做程式碼的隔離、不再需要  
 處理每個模組間的先後順序。

```
}, requires:["所需模組 1", "所需模組 2"]);
```

# 使用 YUI 模組

對使用者來說、只需指定模組名稱即可使用

```
YUI().use("editor", function (Y) {  
    // 此 Callback 代表所需模組皆已載入完成  
    // 你就可以直接建立 Y.Editor 的例項  
  
    var editor = new Y.Editor();  
  
});
```

# 其他選擇 - RequireJS

目前除了 YUI，尚未有其他函式庫內建模組化  
但 RequireJS 可以做到「從零開始」

```
// YUI Module
YUI.add("editor", function () {
    function Editor /* Constructor */ {}
    Y.Editor = Editor;
});
YUI.use("editor", function (Y) {
    new Y.Editor();
});
```

```
// AMD Module
define(function () {
    function Editor /* Constructor */ {}
    return Editor;
});
require(["editor"], function (Editor) {
    new Editor();
});
```

概念幾乎一樣，但 YUI 已經有豐富的模組  
RequireJS 則只是個架構，你要自己把模組加進去

# 模組名稱與載入

新手大多只要負責「用」模組即可  
需大家理解 「模組是 YUI 的最小單位」  
要用任何功能，你必須知道模組的**名稱**：

## YUI 的所有模組列表

<http://yuilibrary.com/yui/docs/yui/modules.html>

一個模組通常就是一個檔案  
所以模組化的缺點是... **檔案會變多！**  
背後也需要有機制**解決 Request 數量**的問題

# 練習：YUI 模組

<http://f2eclasse.com/lab/yui/module.html>

```
// 新增一個 YUI 模組  
YUI.add("模組名稱", function () {  
    // 模組內容  
}, "自定版號", {requires: ["相依模組1"]});
```

- 已經存在一個 class 模組，內有 Y.Class 介紹基本資訊及 getStudents 的函式。
- 請利用 YUI.add 新增一個 fitpi 模組，裡面需定義學生名稱列表 (Y.Class.students) 並指定相依模組。
- 最後 YUI().use 指定 fitpi 並呼叫 Y.Class.getStudents 函式。



# 載入器 Loader

有效率地載入所需要的的 CSS 與 JavaScript 檔案

<http://www.flickr.com/photos/bdesham/2432400623>

# 小網頁不需考慮載入問題

在初期、所需的 JS 與 CSS 都很小



`<script src="http://code.jquery.com/jquery-1.4.2.min.js"></script>` 可做很多事！

<http://www.flickr.com/photos/halfbisqued/2353845688/>

# 網站必需考慮載入方式

發展到一定規模，該如何選擇**每頁所需的檔案**呢？



模組化會切分出很多檔案，如何管理及載入是一門學問

<http://www.nipic.com/show/2/55/d96dde66860c5190.html>

# 較差的載入方式

## 效能及維護性問題

較不專業的網站在面對這樣的問題  
 都是純手工一頁一頁寫、或者是打成一大包  
 得考慮先後順序、維護性低、也無法最佳化

```

11   <link type="text/css" rel="stylesheet" media="all" href="/modules/book/book.css?H" />
12 <link type="text/css" rel="stylesheet" media="all" href="/modules/cck/theme/content-module.css?H" />
13 <link type="text/css" rel="stylesheet" media="all" href="/modules/ctools/css/ctools.css?H" />
14 <link type="text/css" rel="stylesheet" media="all" href="/modules/date/date.css?H" />
15 <link type="text/css" rel="stylesheet" media="all" href="/modules/dhtml_menu/dhtml_menu.css?H" />
16 <link type="text/css" rel="stylesheet" media="all" href="/modules/fckeditor/fckeditor.css?H" />
17 <link type="text/css" rel="stylesheet" media="all" href="/modules/lightbox2/css/lightbox.css?H" />
18 <link type="text/css" rel="stylesheet" media="all" href="/modules/node/node.css?H" />
19 <link type="text/css" rel="stylesheet" media="all" href="/modules/panels/css/panels.css?H" />
20 <link type="text/css" rel="stylesheet" media="all" href="/modules/system/defaults.css?H" />
21 <link type="text/css" rel="stylesheet" media="all" href="/modules/system/system.css?H" />
22 <link type="text/css" rel="stylesheet" media="all" href="/modules/system/system-menus.css?H" />
23 <link type="text/css" rel="stylesheet" media="all" href="/modules/user/user.css?H" />
24 <link type="text/css" rel="stylesheet" media="all" href="/modules/forum/forum.css?H" />
25 <link type="text/css" rel="stylesheet" media="all" href="/modules/cck/modules/fieldgroup/fieldgroup.css?H" />
26 <link type="text/css" rel="stylesheet" media="all" href="/modules/views/css/views.css?H" />
27 <link type="text/css" rel="stylesheet" media="all" href="/sites/default/files/color/garland-034bfcl1/style.css?H" />
28 <link type="text/css" rel="stylesheet" media="print" href="/themes/garland/print.css?H" />
29   <script type="text/javascript" src="/misc/jquery.js?H"></script>
30 <script type="text/javascript" src="/misc/drupal.js?H"></script>
31 <script type="text/javascript" src="/sites/default/files/languages/zh-hant_d14f9c85075d8afd0fe49f2395bd863e.js?H"></script>
32 <script type="text/javascript" src="/modules/dhtml_menu/dhtml_menu.js?H"></script>
33 <script type="text/javascript" src="/modules/lightbox2/js/lightbox.js?H"></script>
34 <script type="text/javascript" src="/modules/panels/js/panels.js?H"></script>
35 <script type="text/javascript">
36 <!--//--><![CDATA[//]><!--
37 jQuery.extend(Drupal.settings, { "basePath": "/", "dhtmlMenu": { "slide": "slide", "siblings": "siblings", "relativity": "rel"

```

# YUI 內建載入機制

YUI 的 Loader 機制

協助載入零散且多的模組檔案



<http://www.flickr.com/photos/billjacobus1/123644872/>

Copyright © 2012 FITPI. All rights reserved.

# YUI 模組載入機制

以 YUI DataTable 為範例 (datatable 模組)

ID	Type	Component	Priority	Owner	Target Release	Status	Summary
#2528873	enhancement	2in3	P3 (normal)	 Reid Burke	FUTURE	assigned	Secure server-side session management
#2531060	defect	2in3		 Reid Burke		new	Using XMLHttpRequest to make requests
#2531099	defect	2in3		 Reid Burke		new	YUI2: Using XMLHttpRequest to make requests
#2531777	enhancement	2in3		 Reid Burke		new	Integrate with YUI 2's XMLHttpRequest API
#2528743	task	Animation	P3 (normal)	 Matt Sweeney	3.NEXT	accepted	Request for proposal
#2529174	enhancement	Animation	P3 (normal)	 Matt Sweeney		accepted	Animation API
#2529537	enhancement	Animation	P3 (normal)	 Matt Sweeney	3.NEXT	accepted	Extract Anim's timer management logic to a standalone module
#2529811	enhancement	Animation	P3 (normal)	 Matt Sweeney	FUTURE	accepted	Add a template attribute to animate complex properties
#2529915	defect	Animation		 Matt Sweeney	FUTURE	accepted	Anim doesn't update end location on pause/change/resume
#2530022	defect	Animation	P3 (normal)	 Matt Sweeney	FUTURE	accepted	Animation keeps running after destroyed
#2530056	defect	Animation	P3 (normal)	 Matt Sweeney	3.6.0	accepted	Anim doesn't work if all the dependencies are already on the page
#2530107	enhancement	Animation		 Matt Sweeney	3.6.0	accepted	Make all animations take advantage of requestAnimationFrame
#2530689	defect	Animation	P3 (normal)	 Matt Sweeney	3.NEXT	accepted	"Alternating Iterations" Animation example to use 'hover' synth event
#2531318	defect	Animation	P3 (normal)	 Matt Sweeney	3.NEXT	accepted	Y.Anim should ignore nodes that aren't bound to a dom node
#2531415	defect	Animation	P3 (normal)	 Matt Sweeney	3.6.0	accepted	IE6, when a container div is faded, its children don't also fade unless container is position:relative
#2531879	defect	Animation		 Matt Sweeney		new	Anim color needs to check for TRANSPARENT

不 reload 的行為很多：換頁、排序、直接編輯。也可以有各種資料來源，是個很複雜的模組。

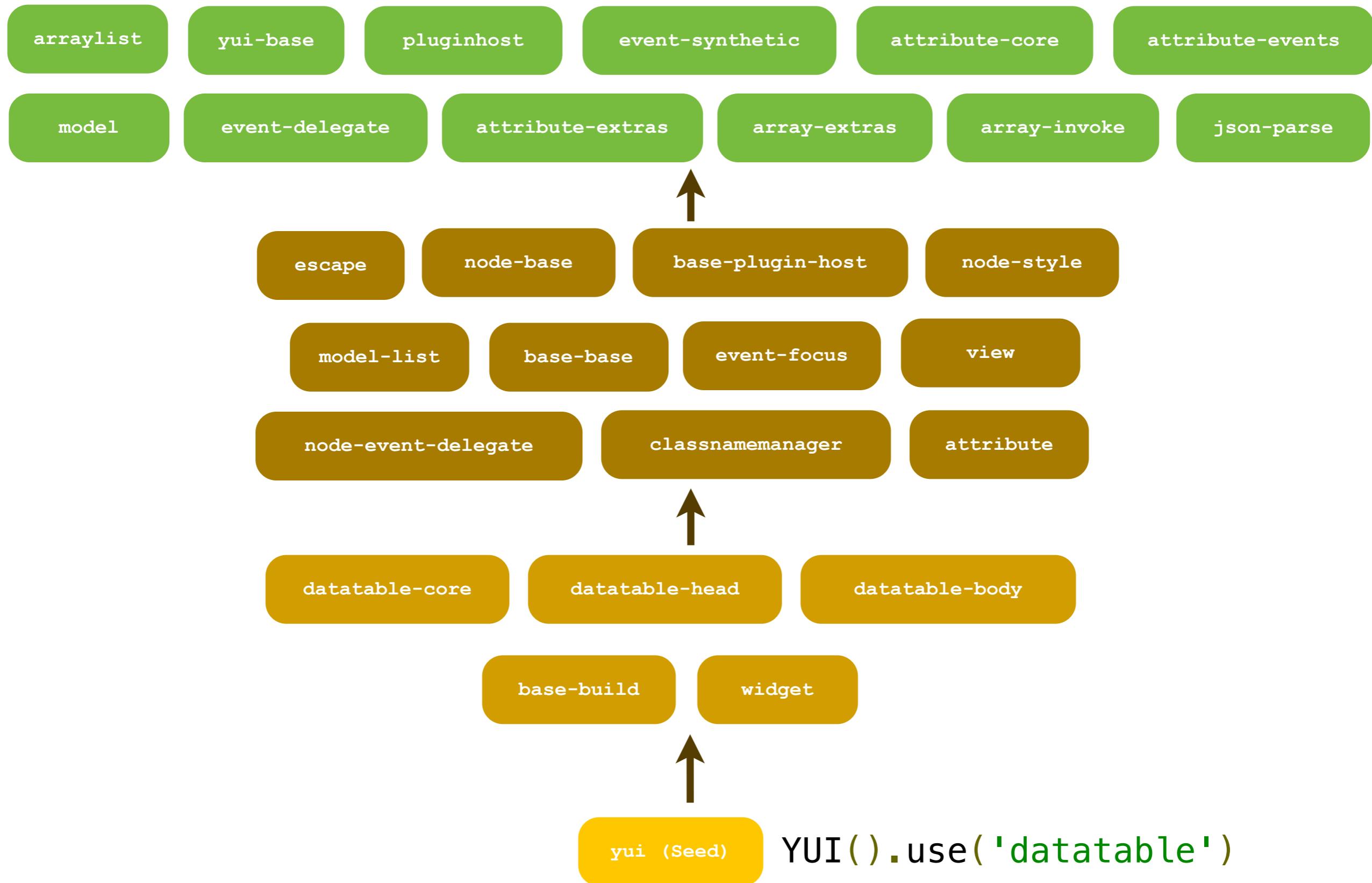
# YUI 模組載入機制

只要幾行程式碼就可以載入所需模組

```
<script src="http://yui.yahooapis.com/3.7.3/build/yui/yui-min.js"></script>
<script>
YUI().use("datatable", function (Y) {
    alert(Y.DataTable);
});
</script>
```

能想像 6 行程式碼背後、做了多少事嗎？

# 計算相依模組的過程



# Combo Handler

強大的壓縮機

```
YUI().use('datatable')
```

前一頁只列出了 28 個模組，但實際上會有 64 個  
64 個模組就代表了 64 個檔案、你可能會覺得很誇張  
但模組化就是要**把不相干的程式抽離**、盡可能精簡  
YUI 的 Combo Handler 可**把數量的問題徹底解決**

<http://yui.yahooapis.com/combo?>

<模組1的檔案路徑>&

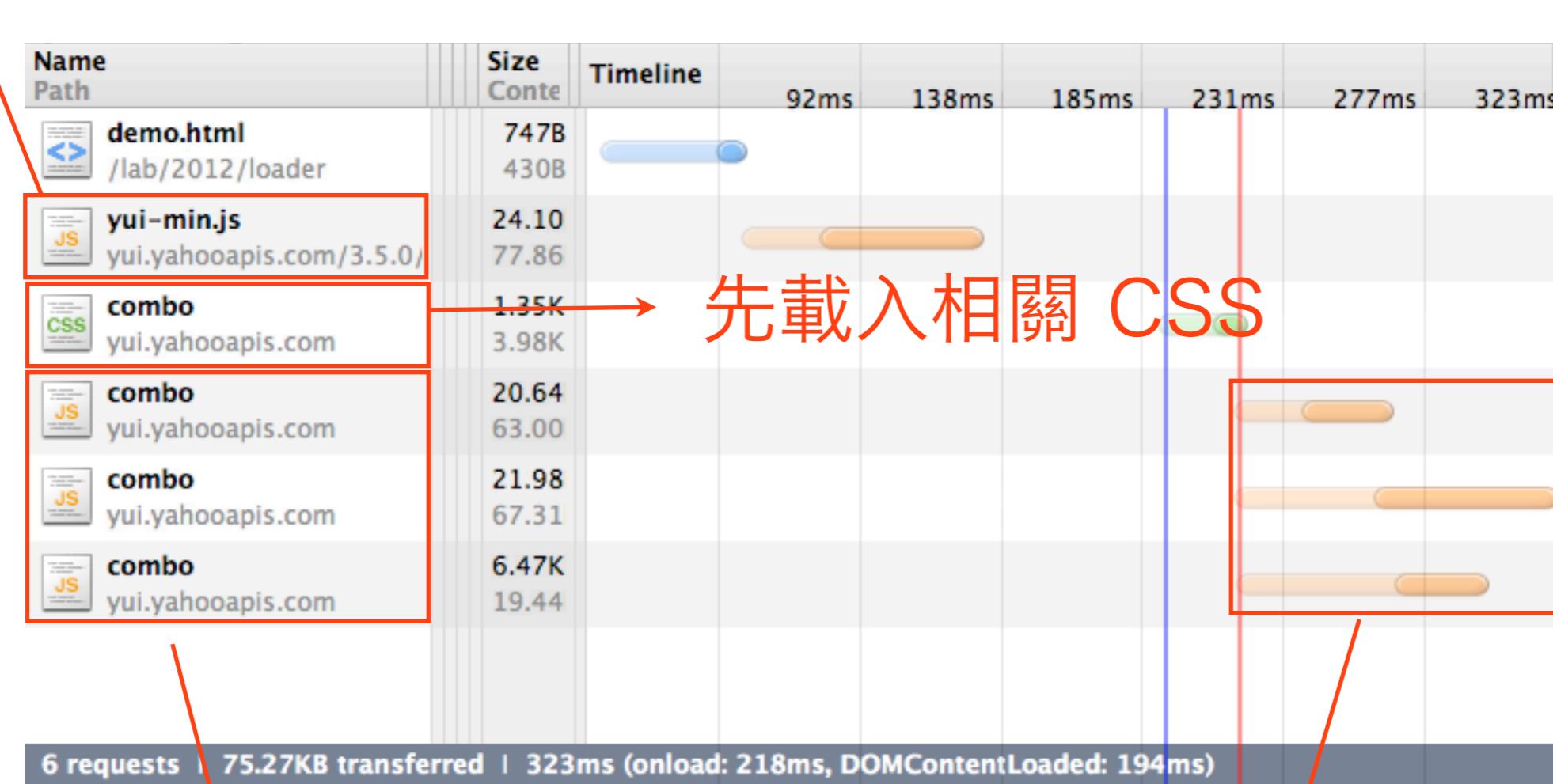
<模組2的檔案路徑>&

...

# 超自動化的平行載入機制

`YUI().use('datatable')`

你所引用的 YUI 種子檔



分散的邏輯：「檔案總數量」、「瀏覽器同時請求數量」、「瀏覽器 GET 長度的限制」

# YUI 載入效能

從種子變成大樹只花了 323 ms

`YUI().use('datatable')`



→  
323ms



是一棵完全沒有贅肉的大樹、因 Module 的切分非常的細、用不到的絕不載入

# 連 jQuery 作者都說...



THE CREATOR OF JQUERY

# JOHN RESIG

*YUI().use() + pulling code off of Yahoo's CDN is  
damn sexy and should be promoted \*VERY\*  
heavily.*

YUI 自動載入函式庫的方式實在  
太 Sexy 了、應該被大力推廣

<http://www.quora.com/How-could-YUI3-improve-its-image-compared-to-jQuery-MooTools-etc>

# 練習：Loader

<http://f2eclass.com/lab/yui/loader.html>

這個練習只要按 F12、選取 Network (網路)  
觀看 YUI 的 request 即可

```
<script type="text/javascript" src="http://yui.yahooapis.com/3.7.3/build/yui/yui-min.js"></script>
<script>
YUI().use("datatable", function (Y) {
    Y.log(Y.DataTable);
    alert("已經載入 datatable 模組");
});
</script>
```

# YUI 官網介紹

<http://yUILibrary.com>

# YUI 官網介紹

- Quick Start

新手上路

- User Guides

分門別類列出 YUI 官方所有內容

- Getting Started

針對一個模組的介紹

- API Documentation

列出一個模組的所有屬性、方法、事件

- Module

大致了解模組的分法

<http://yuilibrary.com/yui/docs/yui/modules.html>

# User Guides

列出 YUI 所包含的所有東西

The screenshot shows the YUI User Guides page with several sections highlighted by red boxes and Chinese annotations:

- Core** (Basic DOM Event): 基本 DOM Event
- Infrastructure** (Element Structure): 元件架構
- Developer Tools** (Testing and Debugging Tools): 測試及除錯工具
- Utilities**: 無介面的 JS 功能 (Interfaceless JS Features)
  - Anim
  - AsyncQueue
  - Cache
  - Cookie Utility
  - DataSchema
  - DataSource
  - DataType
  - Drag and Drop
  - FocusManager Node Plugin beta
  - Get
  - Graphics beta
  - Handlebars beta
  - History
  - ImageLoader
  - Internationalization
  - IO Utility
  - JSON
  - JSONP
  - Pjax beta
  - Recordset
  - Resize
  - Rich Text Editor beta
  - Sortable beta
  - StyleSheet
  - SWF Utility
  - Transition beta
  - Uploader beta
  - Uploader (deprecated version) beta
  - YQL Query
- CSS** (CSS Tools): CSS 工具
  - CSS Base
  - CSS Fonts
  - CSS Grids beta
  - CSS Reset
- Widgets** (Interface-based JS Control Items): 有介面的 JS 控制項
  - AutoComplete
  - Button beta
  - Calendar beta
  - Charts beta
  - DataTable beta
  - DataTable (deprecated version) beta
  - Dial beta
  - MenuNav Node Plugin beta
  - Overlay beta
  - Panel beta
  - ScrollView beta
  - Slider
  - TabView



# YUI Core

用 YUI 處理 JavaScript 的 DOM 與 Event

# YUI 使用方式

建議放置位置: </body> 前

```
<script src="http://yui.yahooapis.com/3.7.3/build/yui/yui-min.js"></script>
```

模組1, 模組2, 模組3... 模組n

```
<script>
```

```
    YUI().use("node", "event", function (Y) { ← 有 YUI 的 Instance 可用
```

```
});
```

模組載入完畢所執行的函式

```
</script>
```

# node - 存取節點

Y.all();  
Y.one();

```
<div id="article">
  <h1>新店市好吃的餐廳</h1>
  <ul class="restaurant">
    <li>可愛村便當：台北縣新店市大豐街8號</li>
    <li>阿輝現炒黃牛肉：台北縣新店市中興路三段225號</li>
    <li>黑人牛肉麵：台北縣新店市中興路三段176號</li>
    <li>蘇杭：台北縣新店市民權路25號2F</li>
    <li>王家包子：台北縣新店市國校路25號之1</li>
    <li>辣。四川成都小炒：台北縣新店市吉安路2-1號</li>
    <li>山東餃子館：台北縣新店市中央路133巷19號1樓</li>
    <li>樂陶然健康小火鍋：台北縣新店市中央五街2號</li>
    <li>六六火鍋：台北縣新店市中央路89號</li>
    <li>龍門客棧：台北縣新店市中央路98號之1</li>
    <li>麵對麵：台北縣新店市中央路33號</li>
  </ul>
</div>
```

## 用 CSS 選取器取得節點

- 取得一個 .restaurant 節點

```
var node = Y.one(".restaurant");
```

- 取得所有 li

```
var nodes = Y.all("li");
```

- 取得第 5 個 li

```
var nodes = Y.one("li:nth-child(5)");
```

## 取得的節點可以再去取節點

- 取得一個 #article 節點

```
var node = Y.one("#article");
```

- 取得 #article 下面的所有 li 節點

```
var itemNodes = node.all("li");
```

練習 - <http://f2e阶级.com/lab/yui/node.html>

# node - 存取節點

- get() / set() 取得 / 設定屬性
- addClass() / replaceClass() / removeClass() / toggleClass()
- getHTML() / setHTML 取得 / 更新內容
- getStyle() / setStyle() 取得 / 設定樣式
- ancestor() 取得母節點
- append() / remove() 新增子節點 / 刪除節點
- 更多介紹: <http://yuilibrary.com/yui/docs/node/>

練習 - <http://f2eclasse.com/lab/yui/node2.html>

# event - 綁定事件

- on(<事件名稱>, <處理函式>)
- e.stopPropagation() 停止氣泡事件
- e.preventDefault() 停止預設行為
- delegate() 委任
- fire() 觸發自訂事件
- 更多介紹: <http://yuilibrary.com/yui/docs/event/>

# 練習：node/event

<http://f2eclass.com/lab/yui/node-event.html>

- 利用 Y.delegate 繫事件
  - Y.delegate("click", <函式>, <選擇器>);
  - 點擊滑鼠把 li 變成黃色
    - <node>.addClass("highlight");
  - 雙重點擊 dblclick 把 li 紿刪除
    - <node>.remove();
  - 刪除完畢觸發一個事件
    - Y.fire("deleted", <餐廳名稱>);

# YUI Anim

動畫

# 練習：Y.Anim

<http://f2eclass.com/lab/yui/anim.html>

- var anim = new Y.Anim(<設定物件>);
- 屬性 from (Object) - 動畫開始時
- 屬性 to (Object)- 動畫結束時
- 屬性 duration (Number) - 總共秒數
- 屬性 easing (String) - 效果
- 函式 anim.run();
- 事件 end



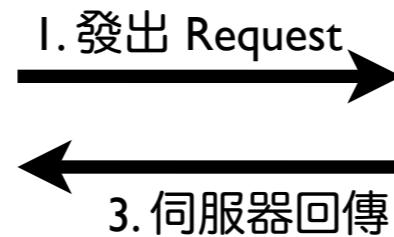
# YUI IO

AJAX - Web 2.0 的基礎

# 傳統網頁更新方式

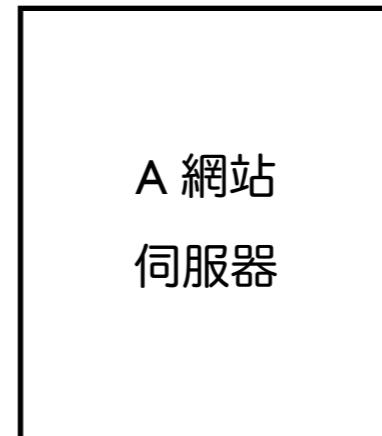
每一次都是整頁重新整理，但可能只需小部分更新

傳統流程：



4. 整份網頁重新整理

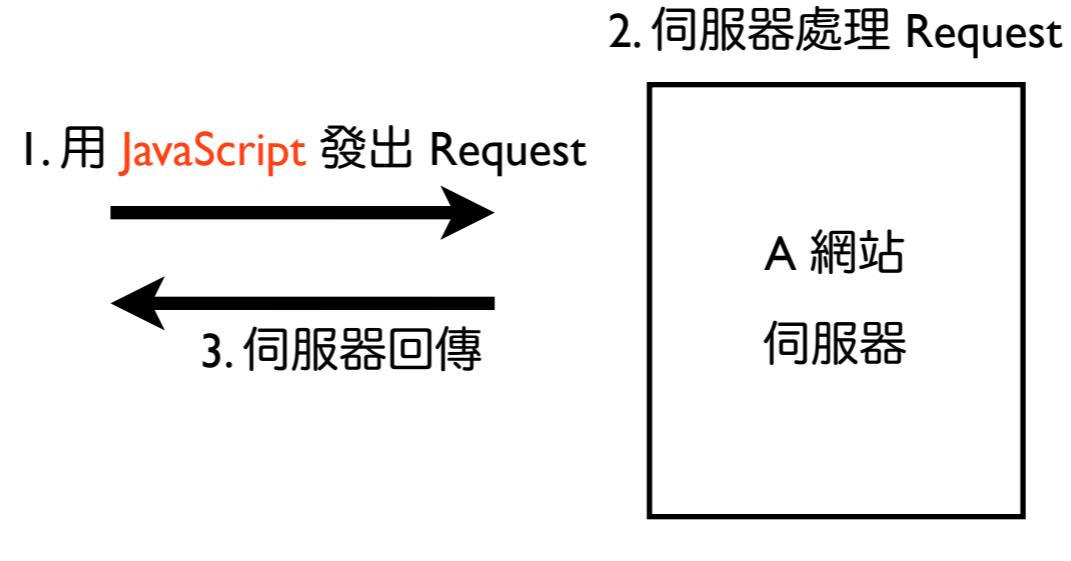
2. 伺服器處理 Request



# AJAX 網頁更新方式

只從伺服器上載入所要更新的資料、而非整頁

## AJAX流程：



# 練習：Y.io

<http://f2e阶级.com/lab/yui/io.html>

- 放在伺服器端同一個目錄下的檔案：

http://f2e阶级.com/lab/yui/io.txt

- 發出請求： Y.io(<url>);
- 伺服器回應： Y.on("io:complete", function (id, o) {});
- 更簡單的作法： <Node>.load(<url>); 即可將資料塞入節點中。
- 更多資料：<http://yuilibrary.com/yui/docs/io/>

# Y.JSONP

存取跨網域的資料

資料存取有個叫 Same Origin Policy 的限制  
JavaScript 是不能存取目前網域以外的資料的  
JSONP 是利用 JS 可被跨域嵌入的特性而實作的

# 常用於網站資料交換

The screenshot shows a Flickr search results page for the tag "f2e". The URL in the address bar is [www.flickr.com/photos/tags/f2e/?page=24](http://www.flickr.com/photos/tags/f2e/?page=24). The page title is "Explore / Tags / f2e". The user is signed in as "f2e-class". The main content area displays a grid of 16 photos from "Dustin Diaz", showing various office scenes and a computer setup. A large red watermark "找不到 JSON" is overlaid on the bottom left of the grid. At the bottom of the page, there is a "Subscribe to stuff tagged with f2e" section with links for "Latest", "geoFeed", and "KML". The footer contains the FITPI logo.

Sort by:  
Most recent • Most interesting

Hey! Are you wondering why  
your photostream isn't showing  
up here?  
[Find out why...](#)

From Dustin Diaz   From Dustin Diaz   From Dustin Diaz   From Dustin Diaz

From Dustin Diaz   From Dustin Diaz   From Dustin Diaz   From Dustin Diaz

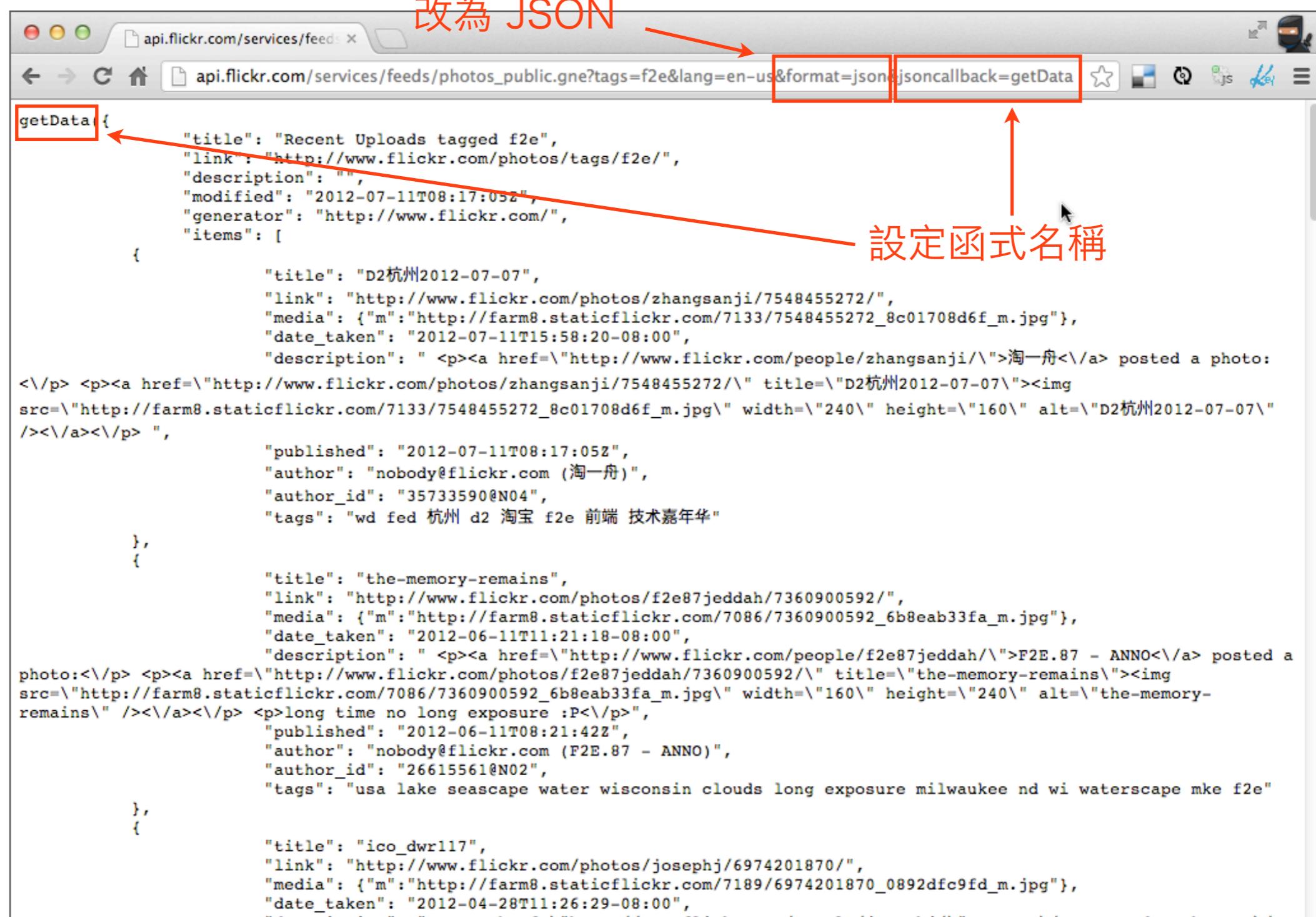
From Dustin Diaz   From Dustin Diaz   From Dustin Diaz   From Dustin Diaz

找不到 JSON

Subscribe to stuff tagged with f2e – [Latest](#) | [geoFeed](#) | [KML](#)

[www.flickr.com/photos/polvero/140897847/](http://www.flickr.com/photos/polvero/140897847/)

# JSONP 就是外部 JS!



```
<script src="http://api.flickr.com/services/feeds/photos_public.gne?
tags=f2e&lang=en-us&format=json&jsoncallback=getData"></script>
```

# 練習：Y.jsonp

取得跨網域資料

- 練習網址：

<http://f2e阶级.com/lab/yui/jsonp.html>

- 利用 Flickr 提供的 API 介面

- ❖ <http://www.flickr.com/photos/tags/<關鍵字>>
- ❖ [http://api.flickr.com/services/feeds/photos\\_public.gne?tags=<關鍵字>&lang=en-us&format=json&jsoncallback={callback}](http://api.flickr.com/services/feeds/photos_public.gne?tags=<關鍵字>&lang=en-us&format=json&jsoncallback={callback})

- Y.jsonp(<URL>, <回傳函式>);

- 在回傳函式中把所需 HTML 組出來

- 更多資料：<http://yuilibrary.com/yui/docs/jsonp/>

# Y.Handlebars

將 HTML 的組合從 JavaScript 中分離出來

JSONP 的範例中我們必須在 JS 中組合 HTML  
十分的不好寫，也不符合網頁標準化的精神  
Handlebars 是專門解決此問題的樣板系統

# 練習：Y.Handlebars

- 練習網址：

<http://f2e阶级.com/lab/yui/handlebars.html>

- YUI().use('handlebars');

- 樣板：`<script type="text/x-handlebars-template">...</script>`

- 編譯：`template = Y.Handlebars.compile(<HTML>);`

- 輸出：`template(<data>);`

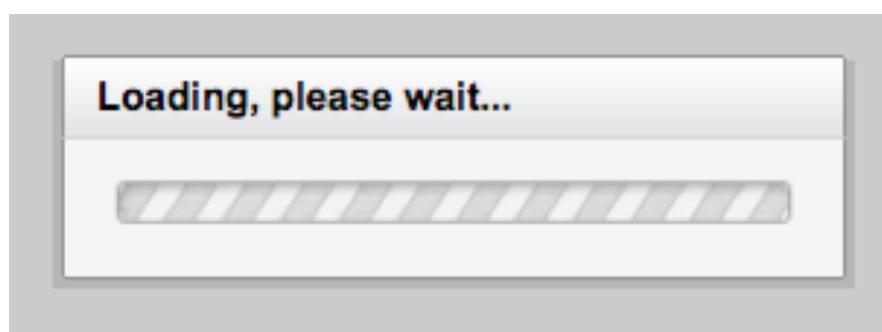
- 更多：<http://yuilibrary.com/yui/docs/handlebars/>

# YUI Panel

Web 2.0 的「開新視窗」

過去我們用 `window.open` 開小視窗  
但現在都是直接開一層 `<div>` 直接浮在頁面上

# YUI 視窗



Please enter your information X

First Name:

Last Name:

E-mail:

State:

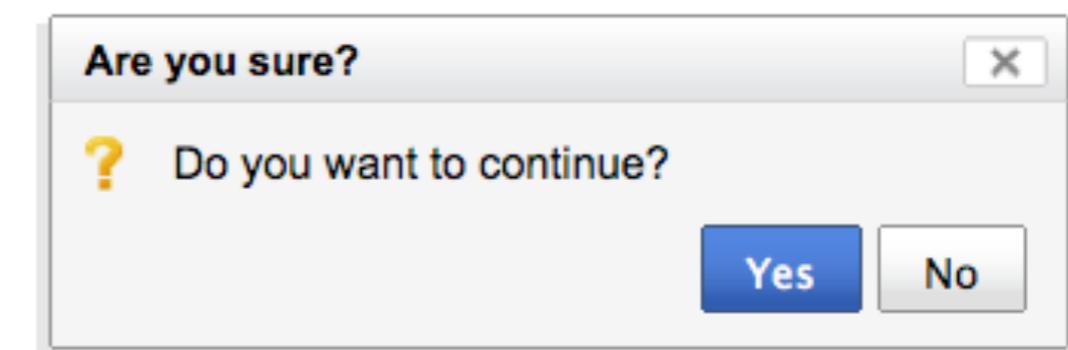
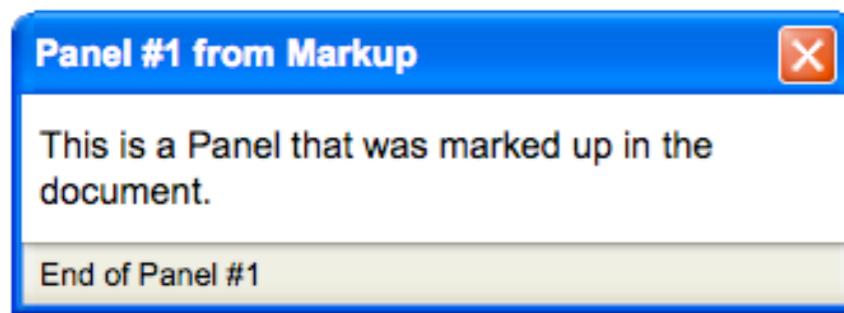
Radio buttons:  1  2

Single checkbox:  1

Text area:

Multi checkbox:  1  2

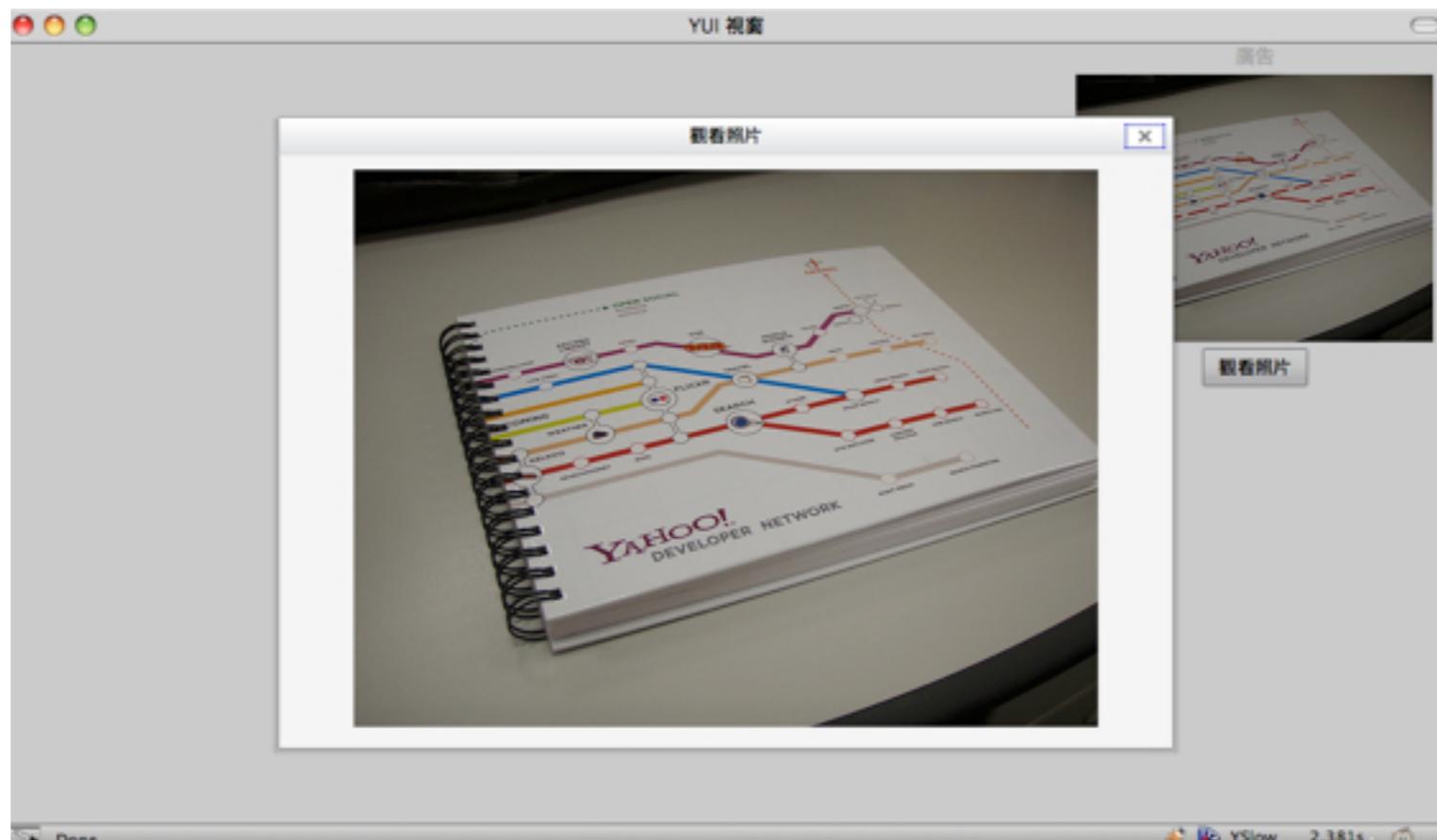
Submit Cancel



# 練習：YUI Panel

<http://f2e阶级.com/lab/yui/panel.html>

- body 增加 .yui3-skin-sam 的 class
- 指定 YUI().use 的模組為 panel
- new Y.Panel(屬性值);



# Panel 所提供的屬性

centered	是否要置中對齊
modal	是否為獨佔模式
visible	預設是否顯示
width	視窗的寬度
height	視窗的高度

# 課程回顧

- YUI 的優缺點、及它的的架構：模組化、載入器、物件導向開發（進階再提）
- YUI 的官網如何使用。
- YUI 的核心函式庫：Node 與 Event，補足了瀏覽器的不足。
- YUI 的函式庫工具、控制項該如何使用。

# Q & A