

# 將網頁製成桌面 app

這就是所謂的「web app」嗎？不確定。

## Prism

Firefox「Gecko」核心的解決方案，<sup>(2016的)</sup>前幾年已中止。之前還在用 Linux 時有用過。

## Fluid

- 核心好像是 Safari「所以這是一個 Mac-only 的解決方案哦」
  - 但好像不是最新版 Safari
- 還不錯用
- 但可能因為核心太舊，執行部分網頁時功能會有小小的異常（例如 Gmail 內不能用「」換行「Facebook 影片無法全螢幕播放」  
↑<sup>20160908</sup> 寄信反映
- 功能可能是所有解決方案裡最豐富的（因為好像幾乎是自己重寫一個瀏覽器一樣）

## Chrome App Builder

- <sup>20160804</sup> 今天才發現原來 Google Chrome app 早就都是 Mac 桌面 app 了，並且可以用 [Chrome App Builder](#) 自製
- <sup>20160804</sup> 【實驗】做了一個 Google Drive 的 app
  - Export App 後會變成「未封裝的擴充功能」，可在 `chrome://extensions` 中載入並執行
  - 也可封裝成 .crx 檔，再拖到 Chrome 中安裝
  - 以上兩種方法都會產生 Mac 桌面 app「並且均可在 Chrome 中管理
- 「Kiosk app」好像是給 Chromebox 之類的迷你桌機用的，讓機器只重覆執行一個 app「所以我應該用不太到吧

## Electron

- [這邊](#)有教用 Electron 的方法 [Electron - Build cross platform desktop apps with JavaScript, HTML, and CSS.](#)
- 需要 Node.js「稍嫌麻煩了點

## WebDGap

- <sup>20160804</sup> 發現還有這個：[WebDGap](#)
- 應是以 Node-Webkit 及 Chromium 為基底
- 試用
  - <sup>20160804</sup> 但好奇怪，要上傳 icon 時卡住了
    - 下載桌面版試試，因為後來發現桌面版有 300 多 MB 大，似乎不只是網頁版的打包 執行結果一樣
    - 反映：[Not working on my machine · Issue #1 · mikethedj4/WebDGap](#) ← <sup>20160808</sup> 今天再試一次 作者修好了！耶比！
  - <sup>20160808</sup> 可以順利上傳 icon「然後選 32bit/64bit → 選平台 下載成品
    - 是個 zip 檔，打開後尚非 .app 格式，看來是要照作者的影片進行必要修改，再看怎樣打包成 .app 吧

- 修改 app 圖示等資源 [作者教學影片](#)
  1. 編輯 `<app 資料夾>/data/hello.app/Contents/info.plist`
    1. `CFBundleDisplayName` `CFBundleName` 改為應用程式名稱
    2. 存檔
  2. 出來以後，把 `<app 資料夾>/data/hello.app` 檔名改為應用程式名稱
  3. 改圖示：
    1. 製作好 Mac OS X 的 .icns 格式圖示檔，並改檔名為 nw.icns
    2. 以上述 nw.icns 取代 `<app 資料夾>/data/hello.app/Contents/Resources/` 中原有的 nw.icns
      - 替代方案：我把 `<app 資料夾>/data/XXX.app` 複製一份，拉到 /Applications 去，用 Litelcon 改好圖示後，再拉回來
- 都改好了，但整個資料夾有辦法進一步打包成 app 嗎？發問：[\[Q\] Howto: make Mac .app from the downloaded folder? · Issue #2 · mikethedj4/WebDGap](#) 得解如下：
  1. 用 Automator 製作一個 app (要與 WebDGap 做的 app 同名！)，只要放一個執行 shell script 的工序，並把 run.sh 的那行程式碼貼入，然後儲存 app
  2. 打開 app 的套件內容，把 WebDGap 提供的 /data 整個移到 /Contents 裡面
  3. 這樣就可以執行該 app 了
  4. 需要的話就修改該 app 的圖示

#### 優點

Node-webkit 及 Chromium 似乎蠻省資源的，以執行 Wikiwand 為例 Firefox 可能需要耗到 100% CPU 而 Node-Webkit 就只要 40 ~ 50%

#### 缺點

app 肥大，要 80~90MB cf. Fluid app 一個大概 7MB 多而已

## createGcApp

- <sup>20160810</sup> 又找到這個！[Create application shortcuts in Google Chrome on a Mac](#)，更扯的是還有人幫忙寫好一個簡單的圖形介面：<http://mait.vilbiks.com/files/createGcApp.dmg> (可直接從這個映像檔中執行)，所以只要決定好
  1. app 名稱
  2. 網址
  3. 圖示 (好像一定要正方形)

三者齊備，就可以製成一個以 Chrome 為底的 app 了~

- 基本上應是一個獨立的 Chrome profile
  - 那可否研究它的 Applescript 改套 Firefox profile 😊
- 能否運用 Chrome 的桌面通知功能？
  - <sup>20160810</sup> 初步試試 (用兩個 Gmail app 互寄信件)，好像沒有作用
- 建 app 時名字盡量不要取 ASCII 以外的字元，以免建 app 時比較有問題？

#### 優

差不多和 Fluid 一樣簡便

亦可自訂 icon

比起 Chrome App Builder 的成品更獨立，所以不會被與 Chrome 視為相同的 app instance 獨立切換 app 無虞

程式核心永遠是最新的 Chrome (只要本機的 Chrome 有一直更新)，不像 Fluid 用的 Safari 還是 Webkit 好像是封閉的

缺  
app 的大小可能會隨 profile 越來越肥 ( 其實 Fluid app 也是越來越肥啊, 只是存在其他地方, 所以這不成問題 )

## 其他

- 還有一招這個 : [Turn Any Website into an App with Automator - ChrisWrites.com](#)
  - 但一樣是用 Safari 為底
- [Flotato - lost in a sea of browser tabs?](#)

From:

<http://ciangbrides.myds.me/dw/> - **Ciang Brides** 黃金城

Permanent link:

[http://ciangbrides.myds.me/dw/it/%E5%B0%87%E7%B6%B2%E9%A0%81%E8%A3%BD%E6%88%90%E6%A1%8C%E9%9D%A2\\_app?rev=1601083034](http://ciangbrides.myds.me/dw/it/%E5%B0%87%E7%B6%B2%E9%A0%81%E8%A3%BD%E6%88%90%E6%A1%8C%E9%9D%A2_app?rev=1601083034) 

Last update: **2020/09/26 09:17**