

從「學習一次、隨處應用」 談 **React Native** 帶來的 開發典範轉移

Jeremy Lu

React 用起來爽很大
好東西不學嗎？XD

Jeremy Lu

@thecat

jeremy@pubulous.com

Founder, builder and instructor,

FullStackRocks.com

- **React**
- **React Native**
- 業界應用現況
- 典範轉移
- 為何你該開始學習
- 相關資訊

- **React**
- React Native
- 業界應用現況
- 典範轉移
- 為何你該開始學習
- 相關資訊

簡介

- Facebook 於 2013 年中開源
- React 內建 VDOM 是輕量的顯示元件
- Flux 管理 Data + Flow
- React 負責 view，Flux 負責 model，兩者合作無間

核心概念

**single
source of
truth**

**thinking in
component
<jsx />**

**always
redraw
(VDOM)**

優點

- 簡單易學、一天入門
- 思維模式單純
- 強大防呆、結構單純、不易出錯
- 頁面效能優越
- **server-rendering, Isomorphic for SEO, faster response**

- React
- **React Native**
- 業界應用現況
- 典範轉移
- 為何你該開始學習
- 相關資訊

簡介

- 2015 年 1 月於 **React Conf** 推出
- 讓所有網頁開發者可製作 **native mobile app**
- 支援 **iOS** 與 **Android**

媽我在這

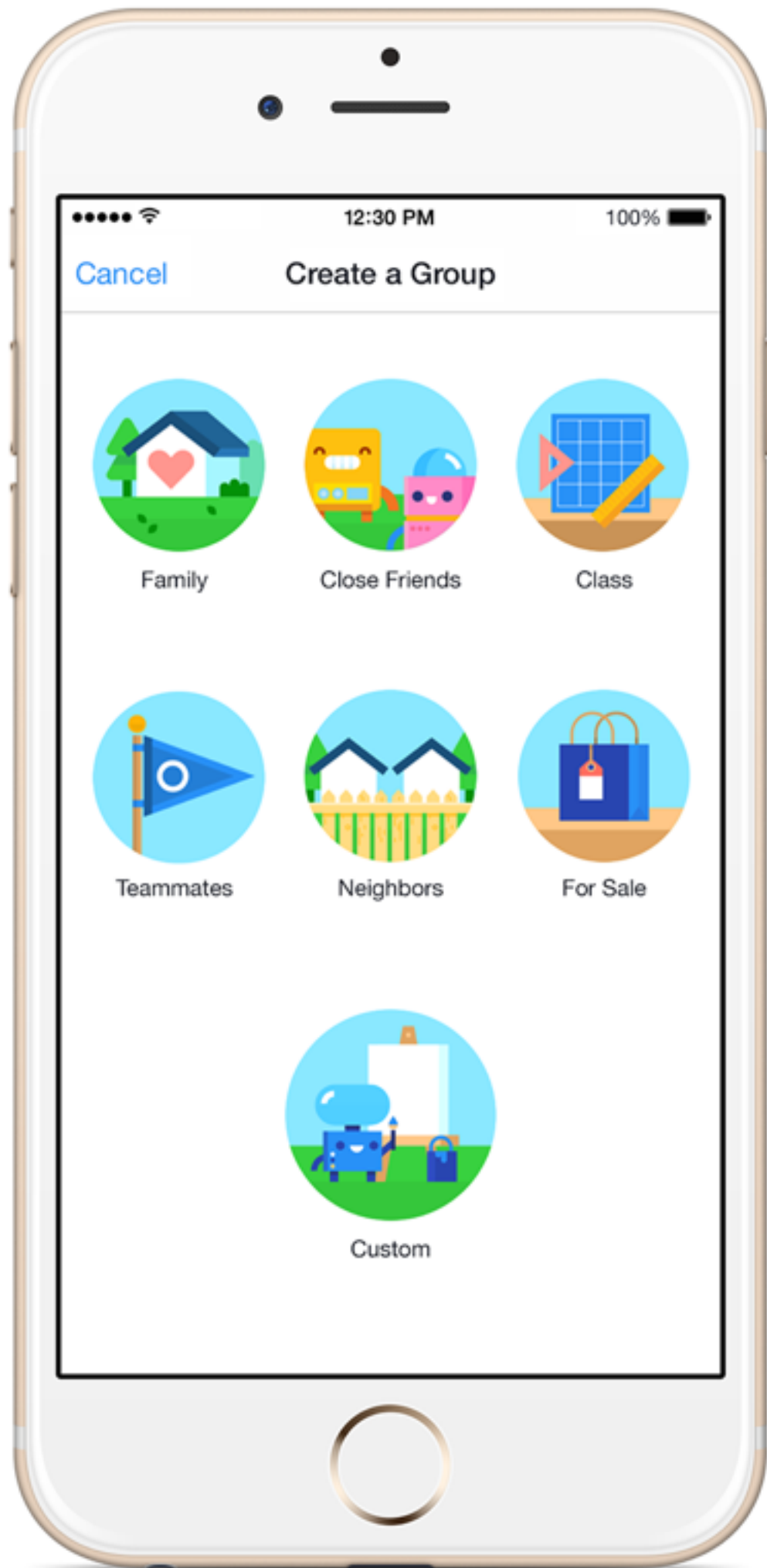






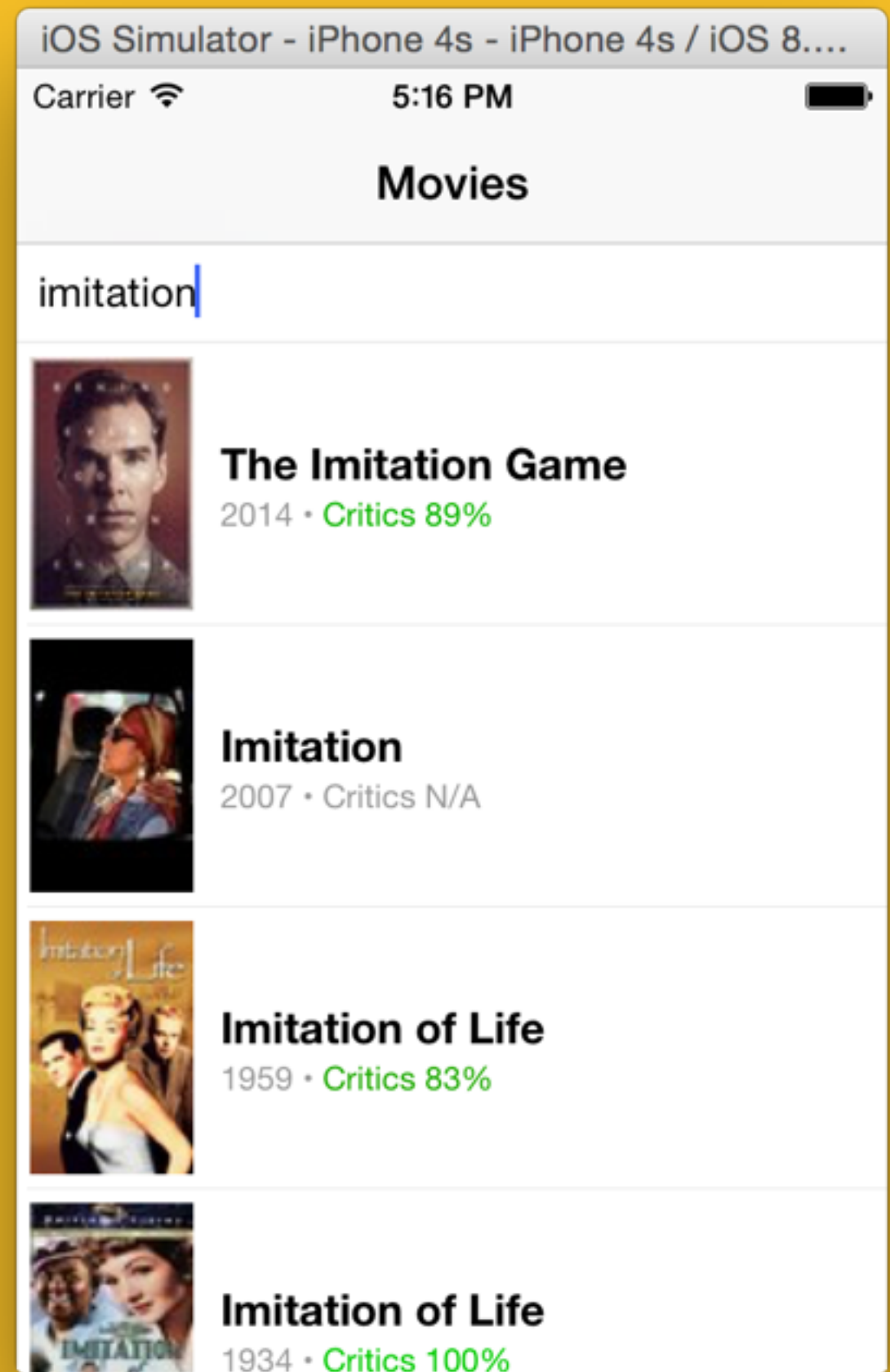
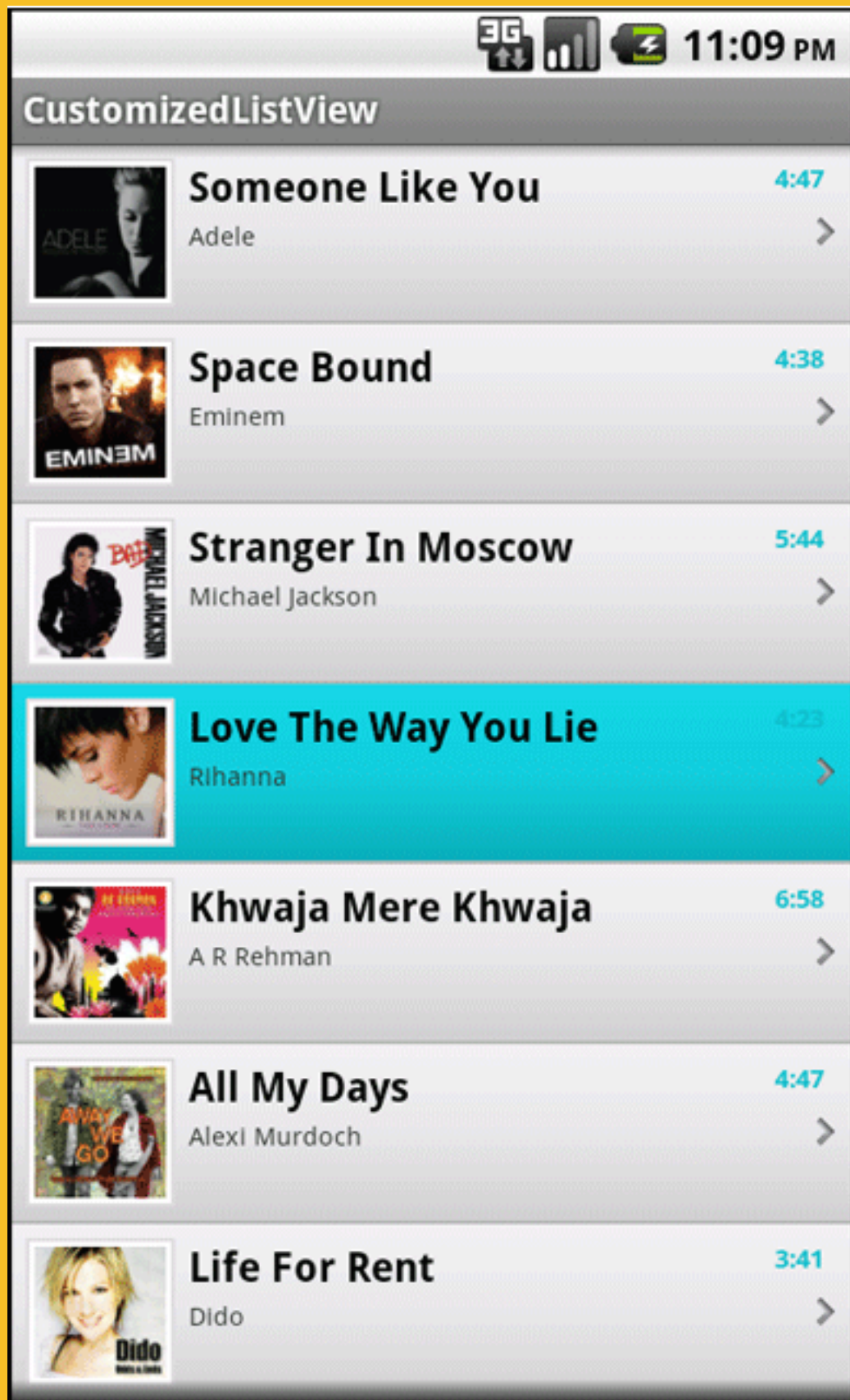
核心概念

- 不使用 Webview 與 HTML
- 採用 javascript/html/css 等熟悉技術開發
- 轉換為各平台原生介面，效能極佳



優點

- 懂 **react** 即可開發
- 開發一次，可支持不同 **drawing backend**



- React
- React Native
- 業界應用現況
- 典範轉移
- 為何你該開始學習
- 相關資訊

業界應用現況

- 北美 **2014** 年起出現大規模轉換潮，持續至今
- 不是已經在用 **react**，就是在轉換到 **react** 的途中
- **Apple, Yahoo, Instagram, Netflix, CodeCademy, Khan Academy, Microsoft...**
- 台灣 **KKBOX, KKTIX, 聯發科**與多家 **startup**
- 人力需求恐急為十年來少見

- React
- React Native
- 業界應用現況
- 典範轉移
- 為何你該開始學習
- 相關資訊

一、開發手法

- 以往是人工維護狀態
 - 極致表現是 **two-way data-binding**
 - 極易出錯、效能不佳

Search...



Pavan D

Blogging



DayaSagar

Available



Pavan B

Busy



Maddy

In a meeting



Pavan H

At work

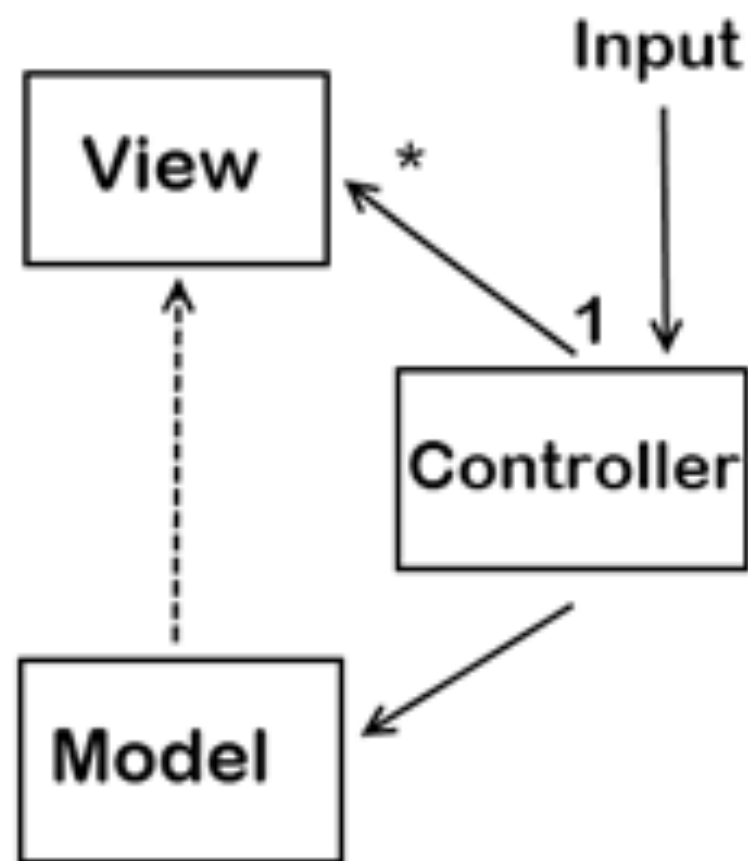


Nikhil

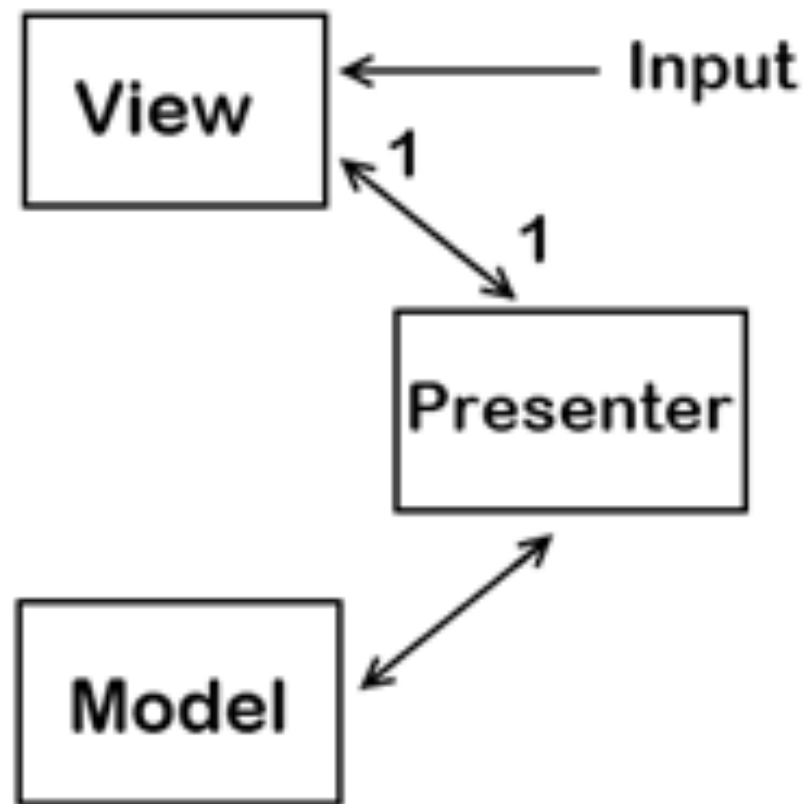
At gym



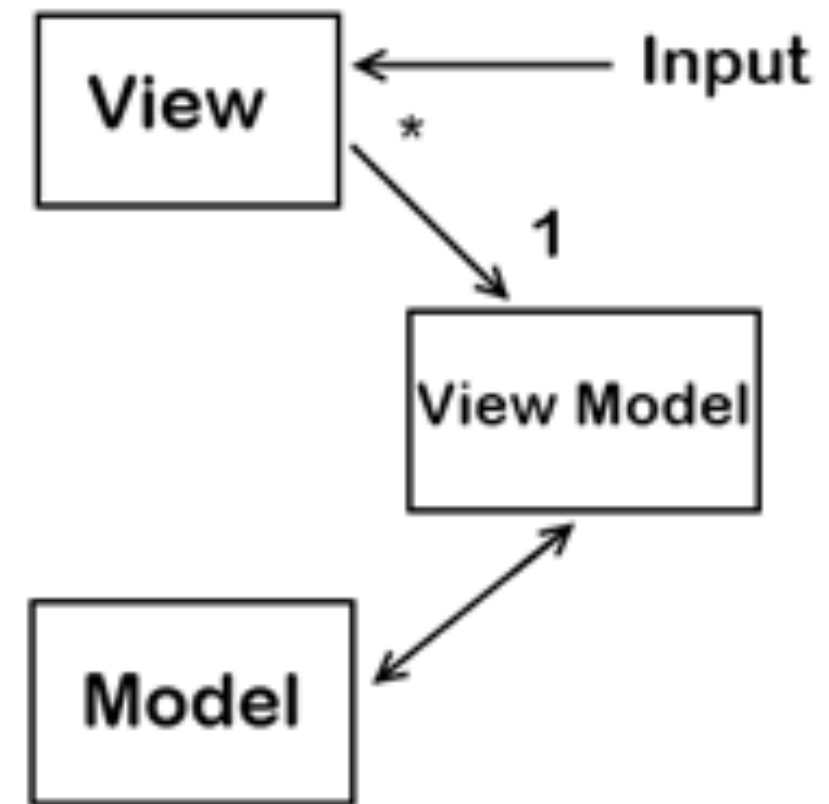
MV*



MVC

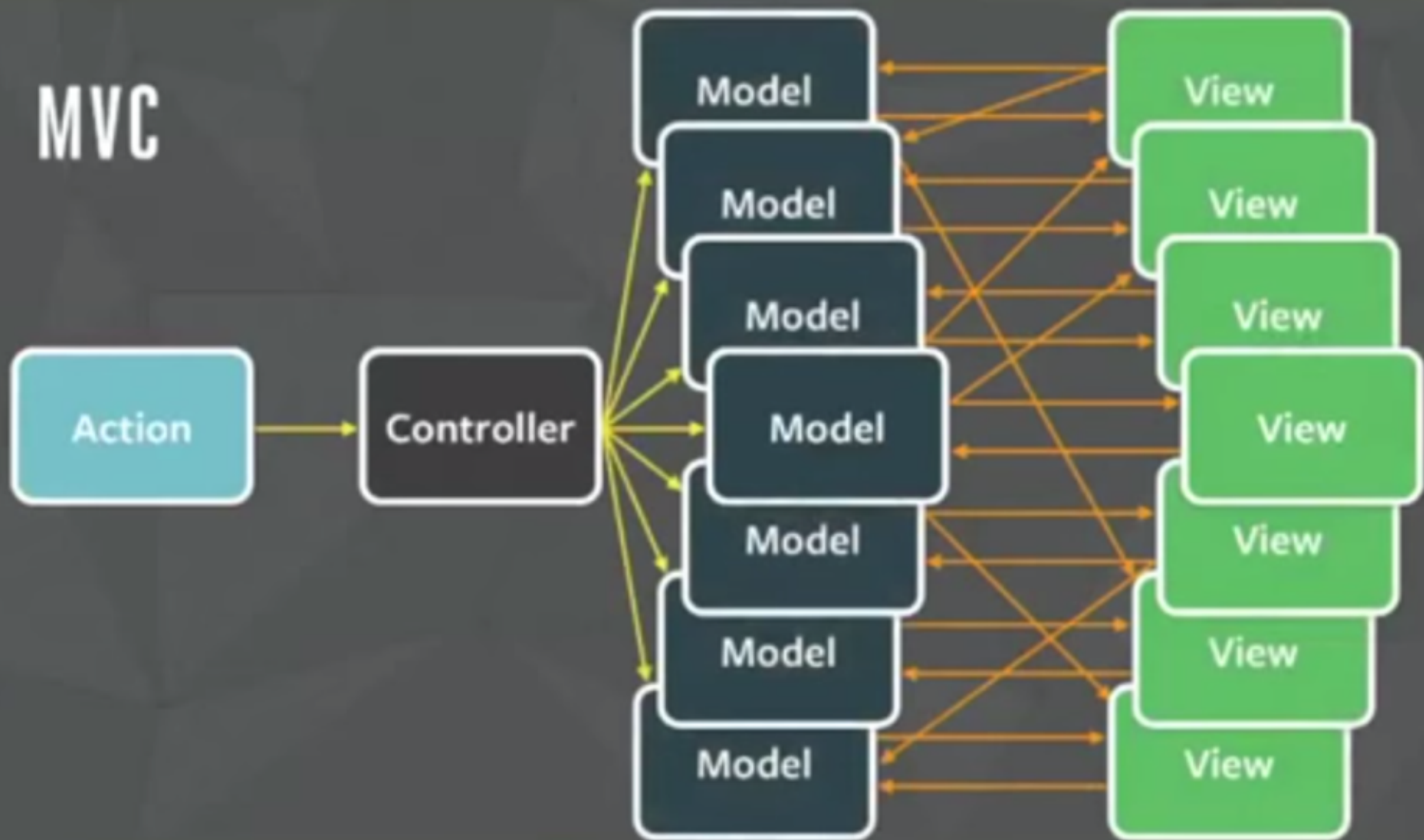


MVP



MVVM

實作結果



開發手法

- 現在是一律重繪
 - 心理負擔輕
 - 出錯機會低
 - 效能佳

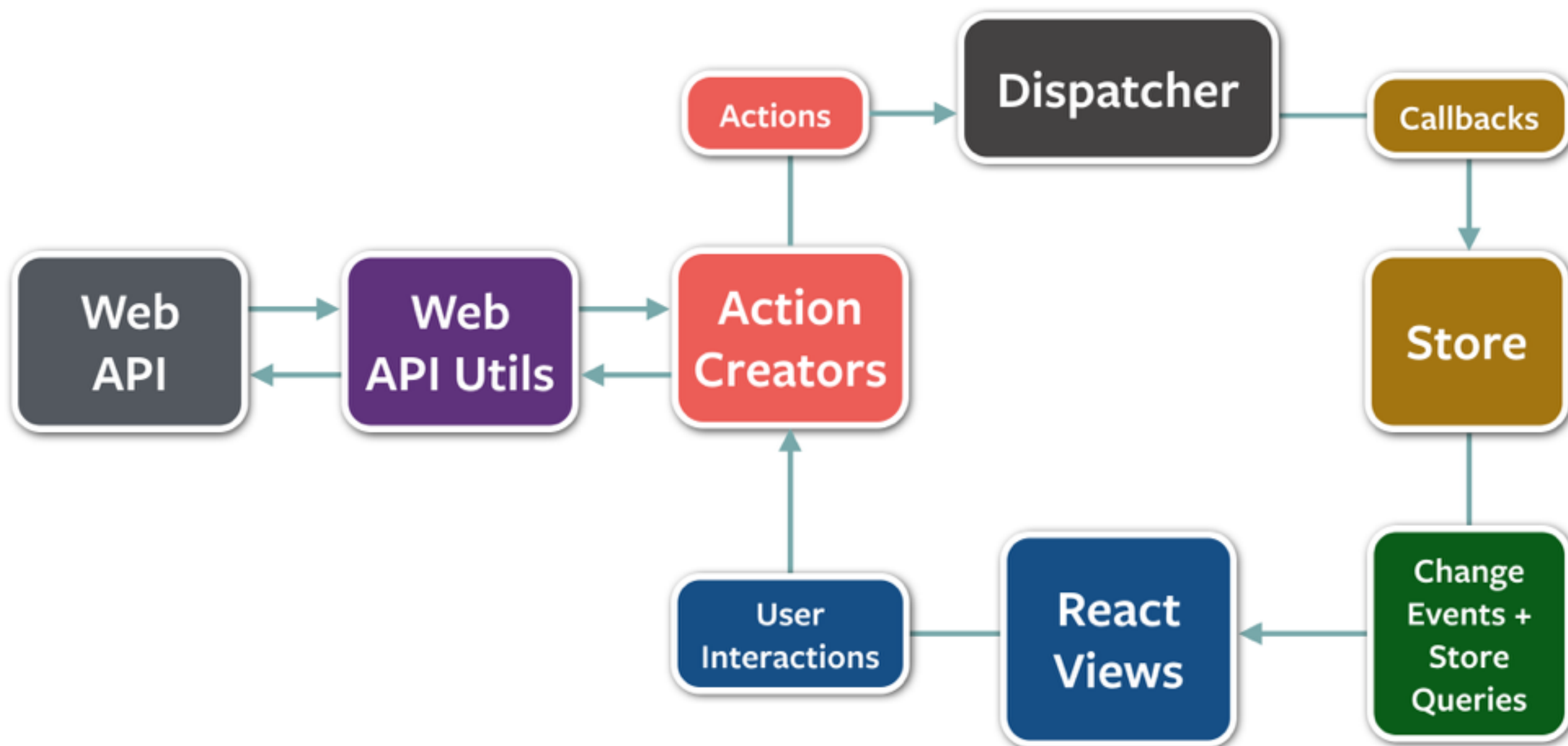


**single
source of
truth**

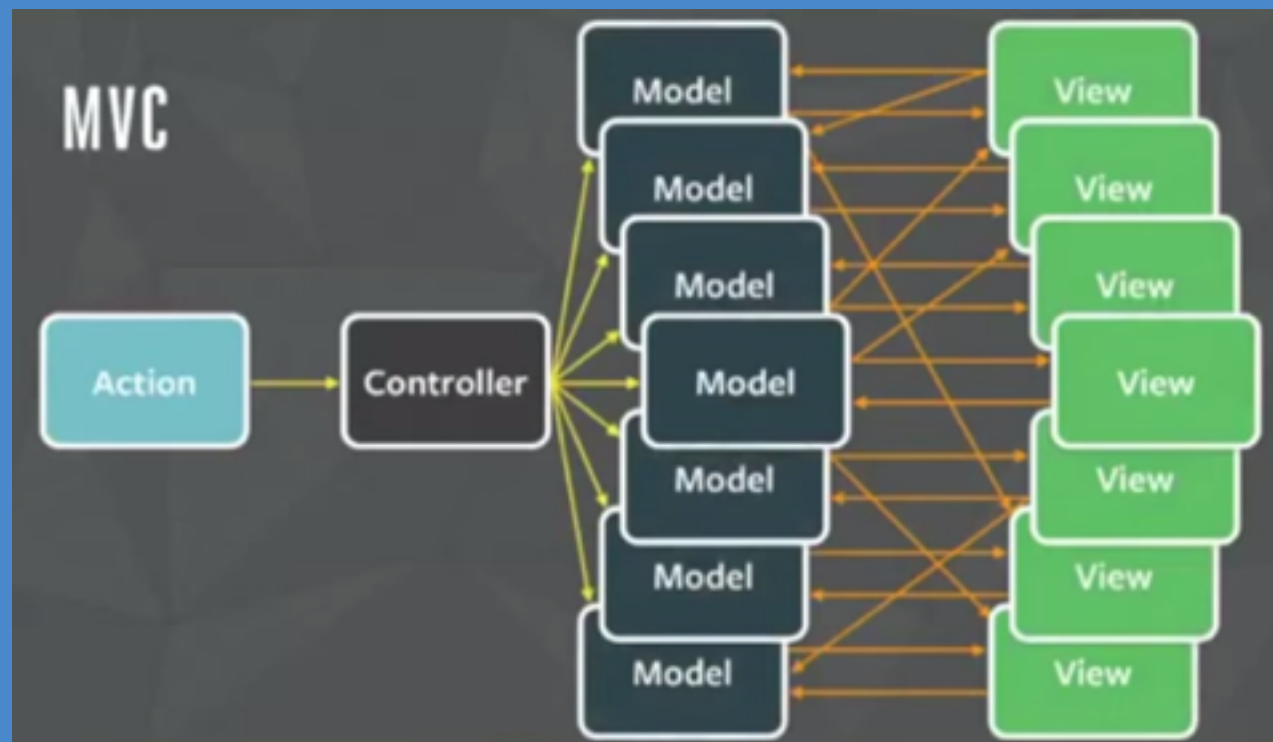
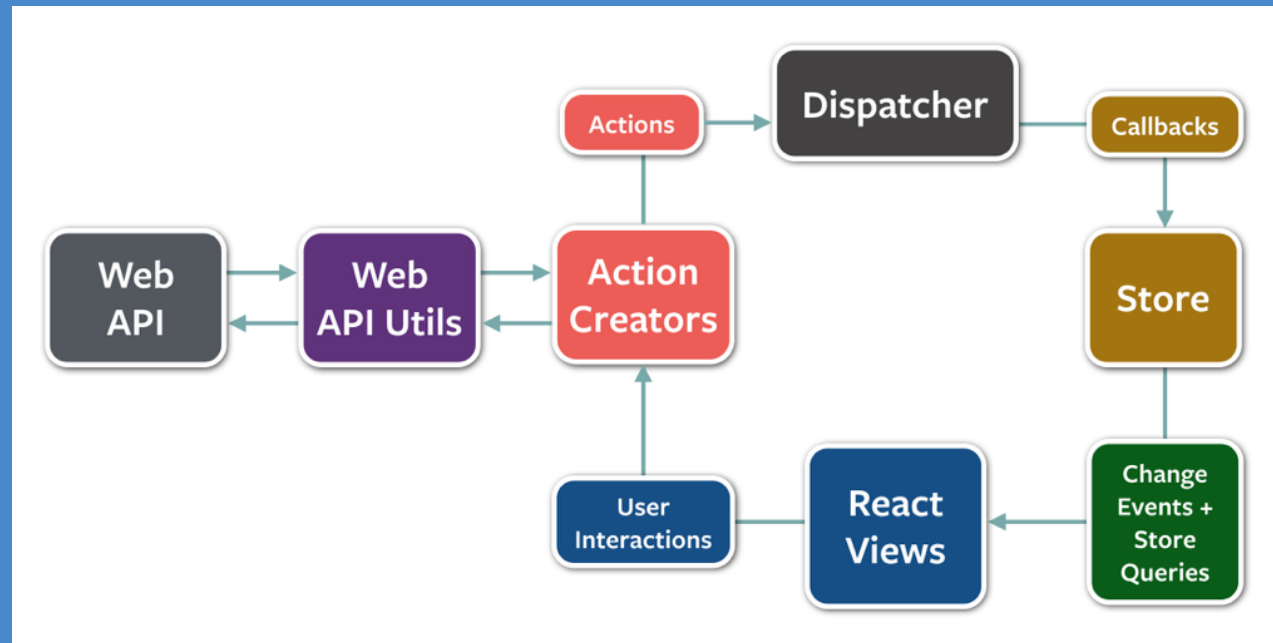
**thinking in
component**

**always
redraw**

唯一真相與單向資料流



兩相比較

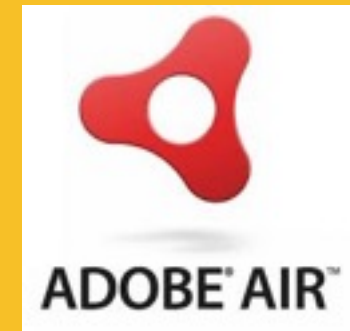


Views -> Action Creators -> Dispatcher -> Stores ->
Views -> Action Creators -> Dispatcher -> Stores ->
Views -> Action Creators -> Dispatcher -> Stores ->
Views -> Action Creators -> Dispatcher -> Stores ->
Views -> Action Creators -> Dispatcher -> Stores ->
Views -> Action Creators -> Dispatcher -> Stores ->
Views -> Action Creators -> Dispatcher -> Stores ->
Views -> Action Creators -> Dispatcher -> Stores ->
Views -> Action Creators -> Dispatcher -> Stores ->
Views -> Action Creators -> Dispatcher -> Stores ->
Views -> Action Creators -> Dispatcher -> Stores ->



二、架構策略

- 以往是 **write once, run everywhere**
 - 早期用 **Java Swing, Adobe Flash/Air**
 - 後來是 **Phonegap** 等 **webview-based** 方案(ionic...)
 - 結果
 - 用戶體驗不佳
 - 效能不佳
 - 評價不佳



架構策略

- 現在是 **learn once write anywhere**
- 效能
 - 不受 **webview** 效能限制，可常保 **60FPS**
- 體驗
 - **native UI** 帶來最佳用戶體驗(操作習慣、介面語言)

架構策略

- 節省成本
 - 時間
 - **60%** 程式可跨平台通用
 - 僅 **UI** 部份需針對不同平台重寫
- 品質
 - 不易犯錯，維護成本降低
- 人力
 - 易學易上手，快速產生極戰力

VDOM

**Abstract
Component**

<View />



Drawing Backend

Browser

<div />

Canvas

rect()

iOS

UIView

WebGL

gl.createShader

Android

**android.view.
View**

windows
mac
linux
windows phone
....

- React
- React Native
- 業界應用現況
- 典範轉移
- 為何你該開始學習
- 相關資訊

為何你該開始學習

- **react** 的核心概念可終身受用，將來在任何框架下皆可應用
- 學會一項工具可同時開發 **web**、**app**、**desktop**，應用範圍極廣
- 多年來少見時間投資成本效益極高的選擇，好東西不學嗎？



Andy Matuschak
@andy_matuschak



Follow

I say with confidence as a former UIKit author: React's model for the UI layer is vastly better than UIKit's. React Native is a *huge* deal.



RETWEETS FAVORITES
571 **519**



2:54 AM - 29 Jan 2015

用最好的工具
做开心的工作
賺豐厚的報酬
過恰意的生活

學習資訊

- <https://www.facebook.com/groups/reactjs.tw/>

2000+ members

- <https://www.facebook.com/twthecat>

歡迎加臉書好友，有問必答™

B.B. KING

1925 - 2015



Questions?

@thecat | jeremy@pubulous.com

Founder, builder and instructor,

FullStackRocks.com