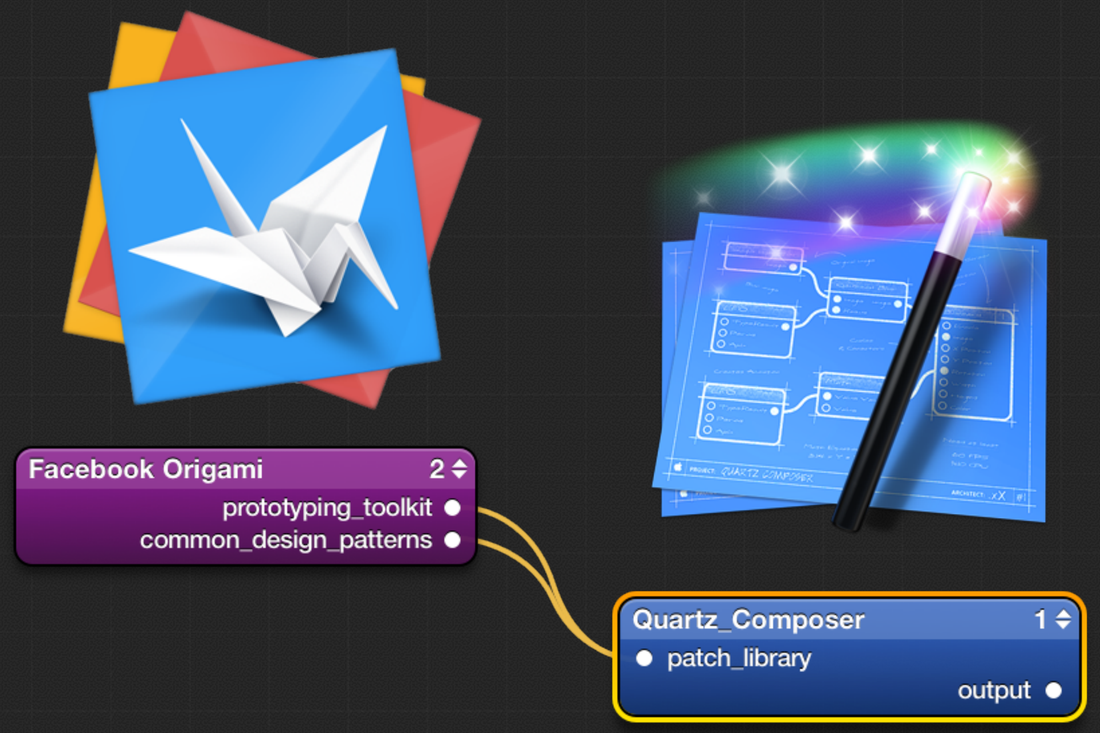
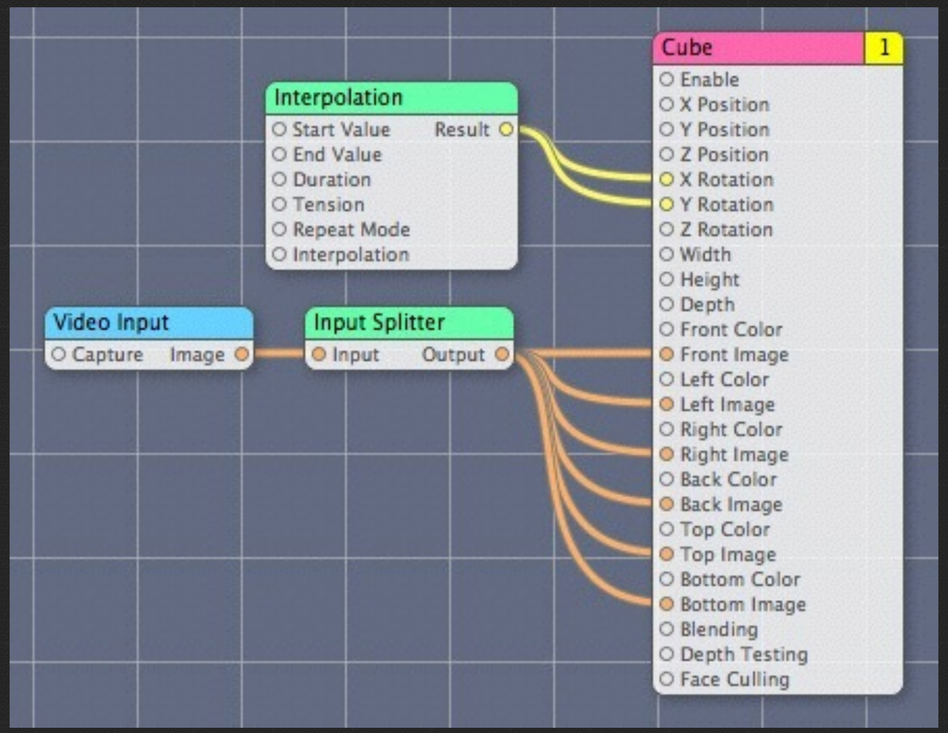
waynedahlberg



Quartz Composer & Origami

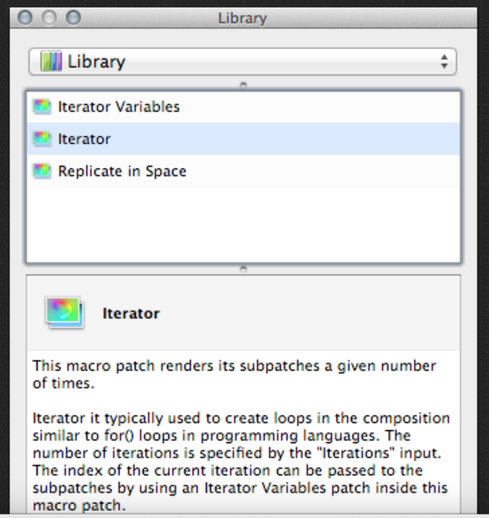

Cube
1
Enable
X Position
Y Position
Z Position
X Rotation
Y Rotation
Z Rotation
Width
Height
Depth
Front Color
Front Image
Left Color
Left Image
Right Color
Right Image
Back Color
Back Image
Top Color
Top Image
Bottom Color
Bottom Image
Blending
Depth Testing
Face Culling
Interpolation
Start Value
Result
End Value
Duration
Tension
Repeat Mode
Interpolation
Video Input
Capture
Image
Input Splitter
Input
Output

Quartz Composer

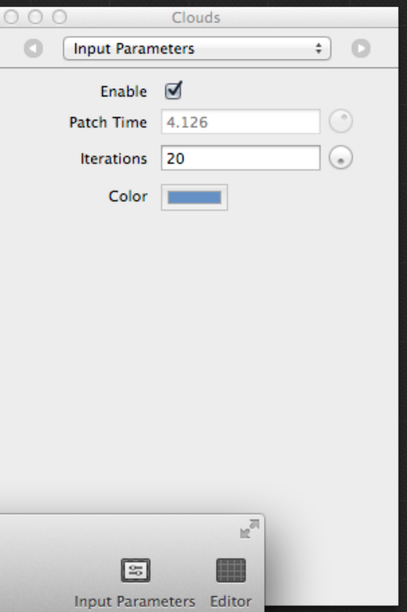
- node-based visual programming
- state-based graphics canvas
- built on OpenGL & Core Image
- portable .qtz file format
- VideoDJ’s love this thing

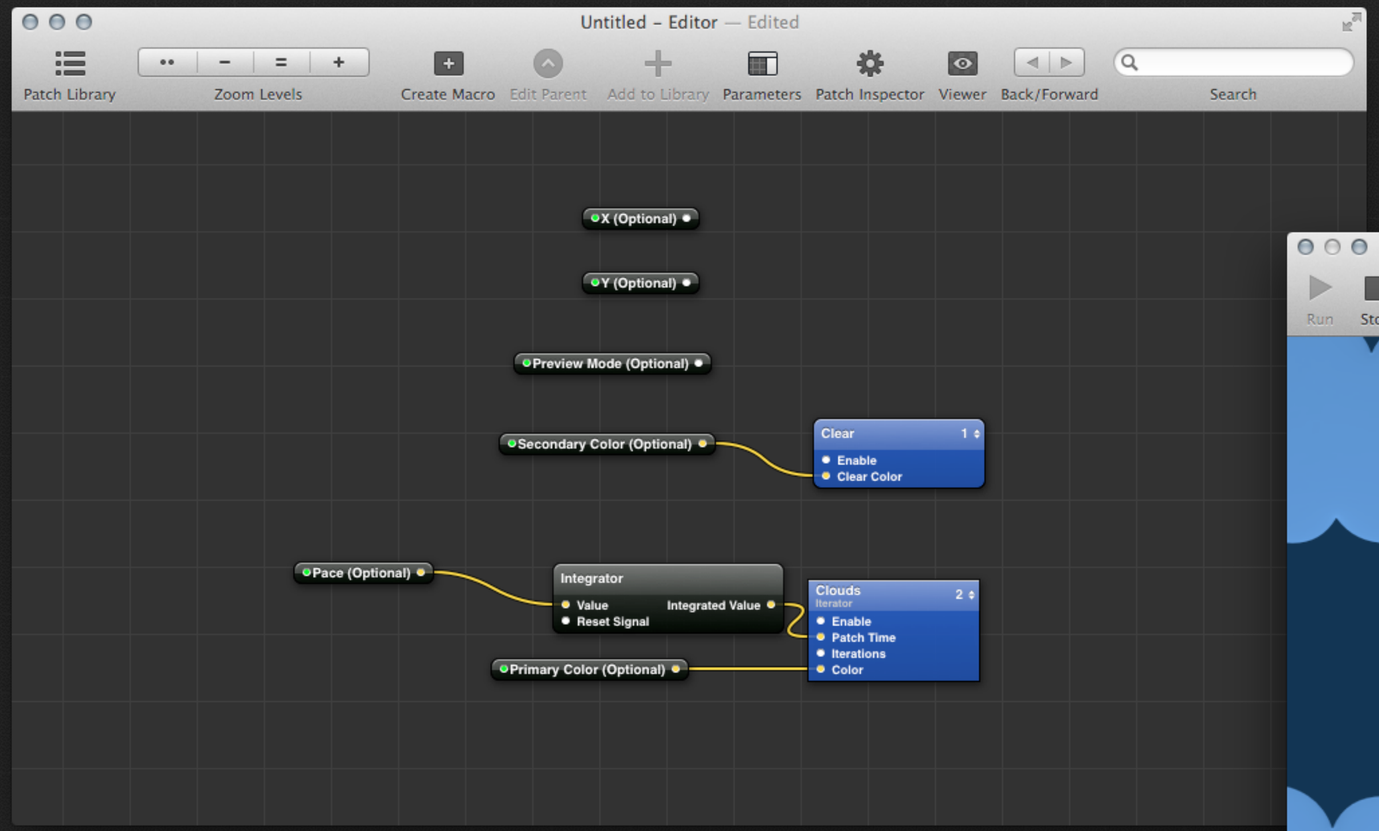
QC Interface

Patch Library



Patch Inspector

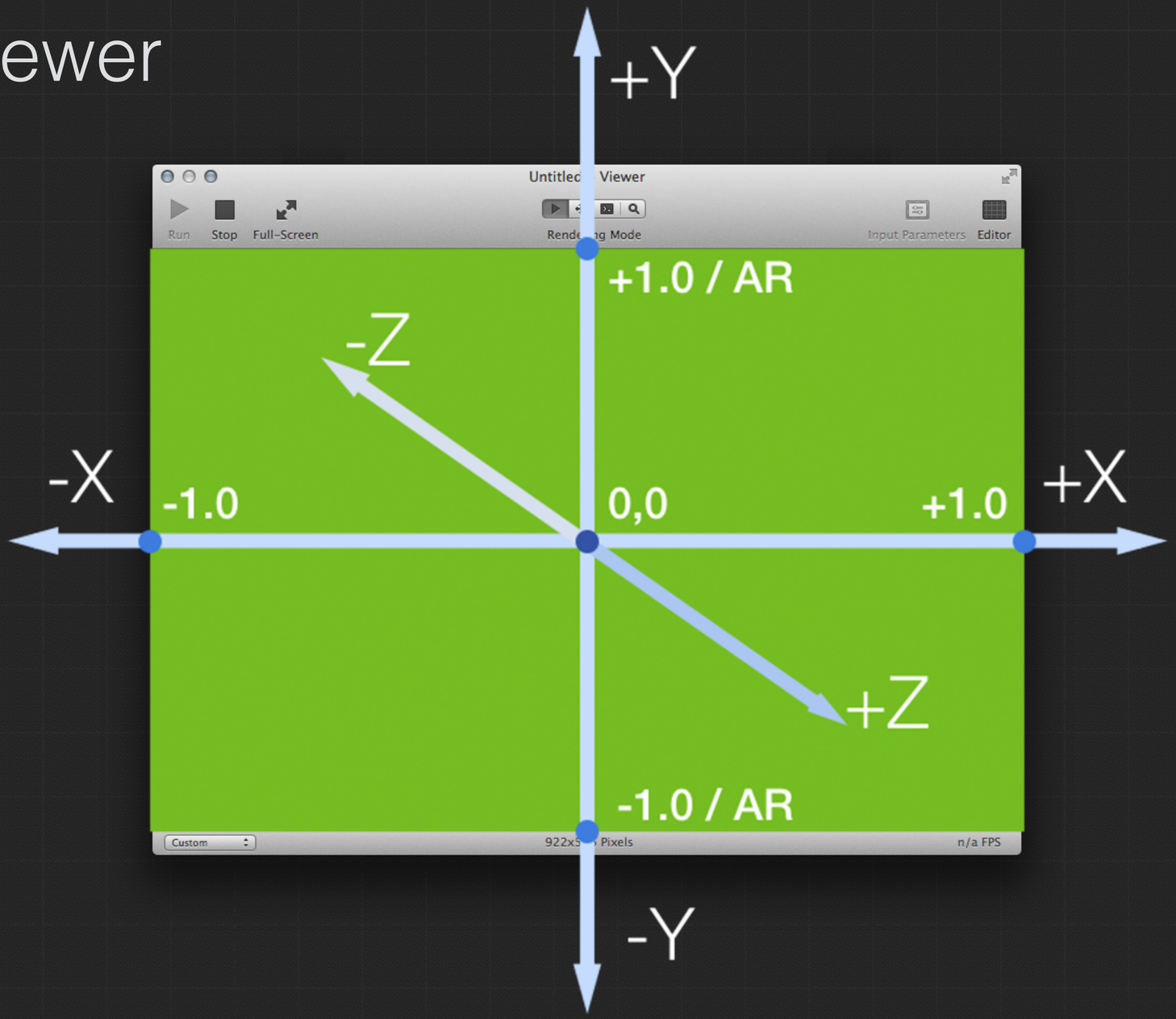




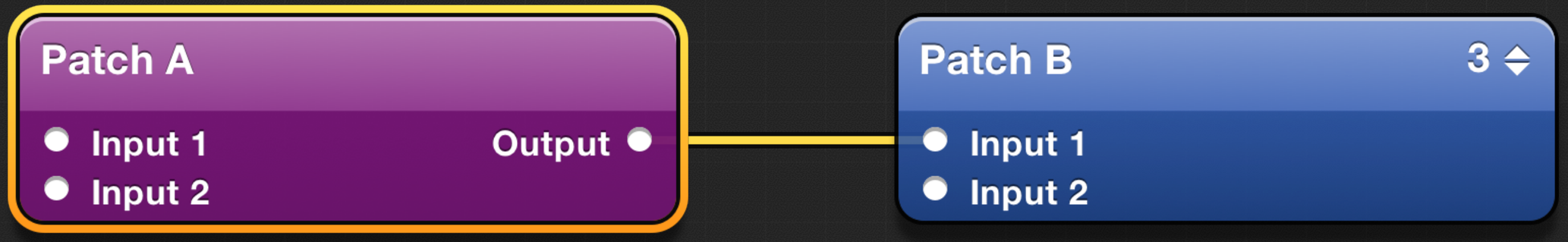
Editor

Viewer

QC Viewer


+Y
Viewer
Run
Stop
Full-Screen
Input Parameters
Editor
+1.0 / AR
-Z
-X
-1.0
0,0
+1.0
+X
+Z
-1.0 / AR
Custom
Pixels
n/a FPS
-Y

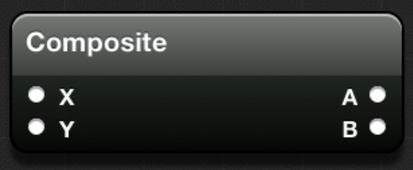
QC Patches



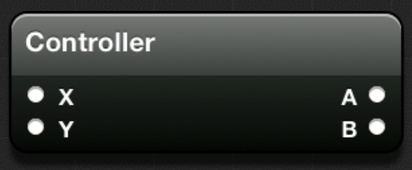

```
Result = function (time, {0 or more input params})
```

QC Demo

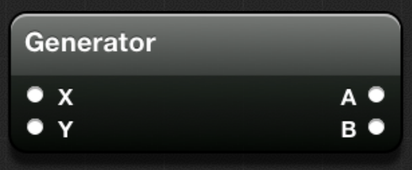
QC Patch Types



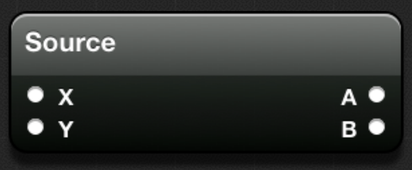
- addition
- screen
- multiply
- overlay
- burn/Dodge
- luminosity
- many more…



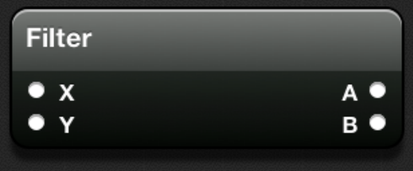
- mouse
- audio
- video
- other waveforms



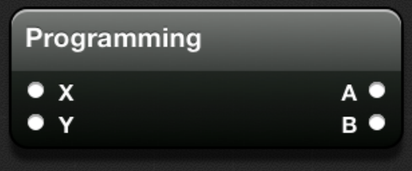
- image with String
- checkerboard
- starShine
- randomGenerator



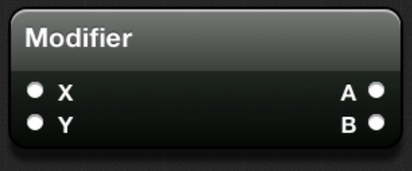
- image loader
- movie loader
- audio loader
- directory scanner
- rss downloader



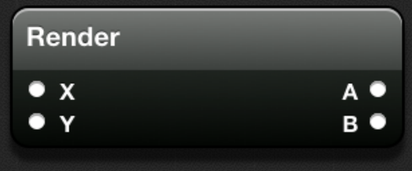
- pixel operations
- pixellate
- unsharp mask
- gloom
- comic book
- edges



- code execution
- javascript();
- CoreImage filters
- GLSL shaders



- modify inputs
- color transform
- image crop
- modify Strings
- modify textures
- inspect Structures



- render to viewer
- Clear
- Billboard
- Particle Systems

Quartz Composer

Pros

- Time to interactive prototype is much faster.
- State based, many outcomes with one comp vs timeline based tools.

  Changes are instant. No compiling or refreshing.
- Easier for non-coders to visualize parameters of objects and experiment.
- Review, validate & iterate interaction concepts quicker.
- Built on OpenGL and Core Image.

Quartz Composer

Cons

- Poor official documentation. (2007)
- Increased complexity, buggy :(
- Built primarily for dynamic motion graphics, not interface design prototyping.
- QC environment has drastically different tools, conventions than traditional design tools. (Photoshop, Illustrator, Sketch)
- QC does not currently play with iOS. Several workarounds.
    - ‘LiveView’ is slow, unreliable.
    - ‘QC-mobile’ limited patch support. No Origami.

Why Quartz Composer?

“Our hands feel things, and our hands manipulate things.
Why aim for anything less than a **dynamic medium**
that we can see, feel, and manipulate?”

Bret Victor
@worrydream

‘A brief rant on the future of interaction design”

Mike Matas

Our Choice for iPad

PPP joins FB

“this is one of the most elegant, fluid, impressive apps you've ever seen. It's a showpiece for the new world of touch-screen gadgets.” *- David Pogue*

Now we're taking our publishing technology and everything we've learned and are setting off to help design the world's largest book, Facebook.

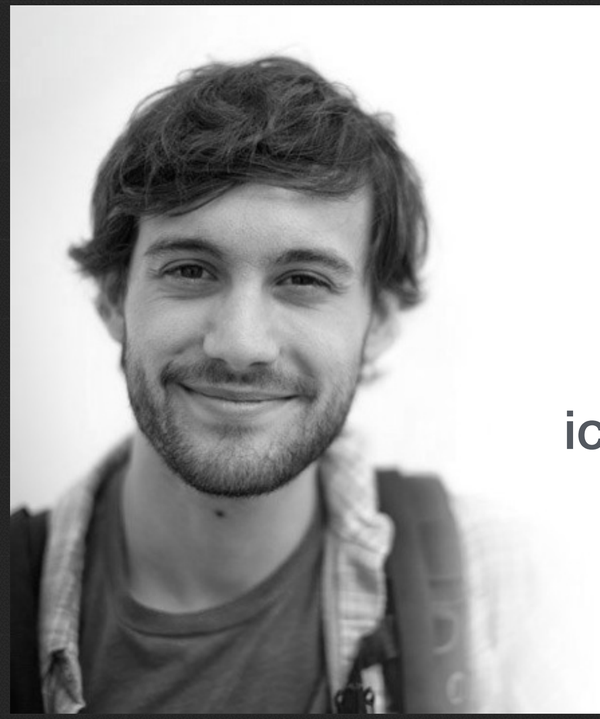
pushpoppress.com/about

2011

Mike Matas

“My secret weapon is
Quartz Composer. These days,
I use that more than anything else.”

Mike Matas



“**Quartz Composer** is more about animations and interactivity. […] There are very few buttons and icons in Push Pop Press so these days I spend most of my time using it”

December 2013

Origami

Free-to-use set of simple starter patches that allow for a number of common design patterns and interactions.

http://facebook.github.io/origami/

Design Pattern Interactions

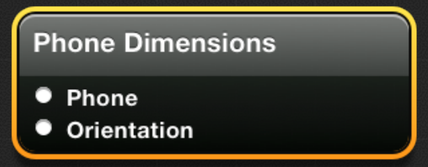
1. Drag an image of your interface design into a device template (*iPhone, Android and Windows phone templates are provided*) and make that image scrollable within the device.

2. Easily hook in mouse input and keyboard commands to make something happen on the screen.

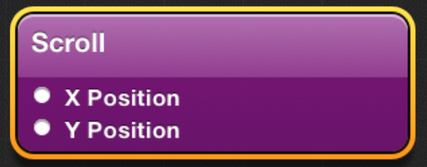
3. Implement common animations such as sliding panes and drawers, zooming and shrinking photos, rotating icons, etc with springs.

Origami Patches

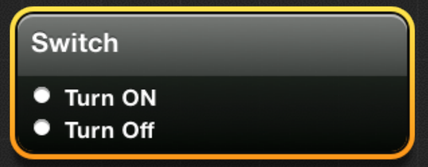
draws a phone with a specified screen image



provides image dimensions for phone patch

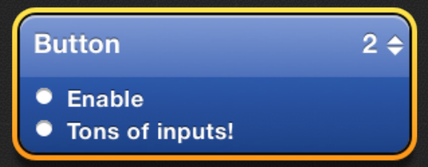


scrolls a layer that's drawing an image

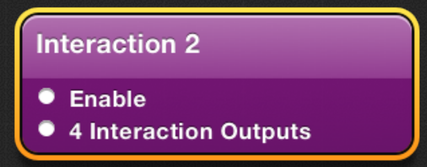


toggles ON / OFF remembers state

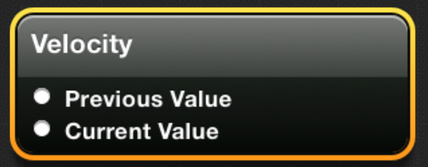
animate a changing value with a bouncing spring



button with customizable label and background



provides various input information from layer patches, down, tap, click, drag, enable.

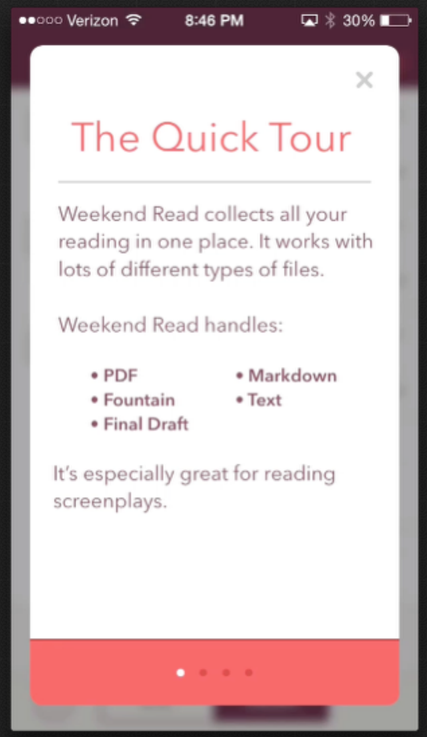


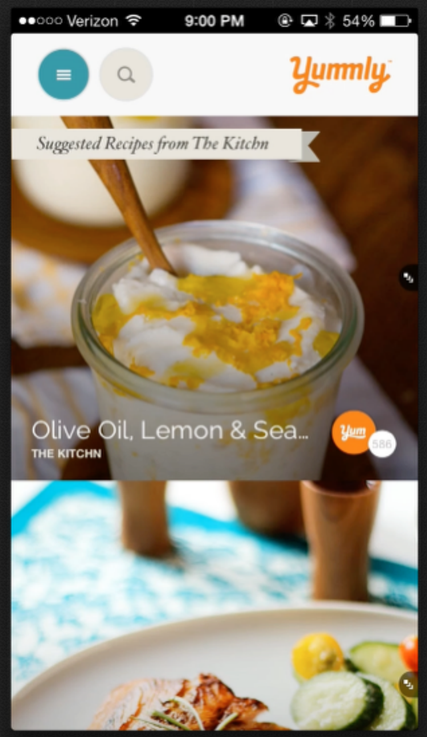
Outputs the difference in value between the previous frame and current frame.

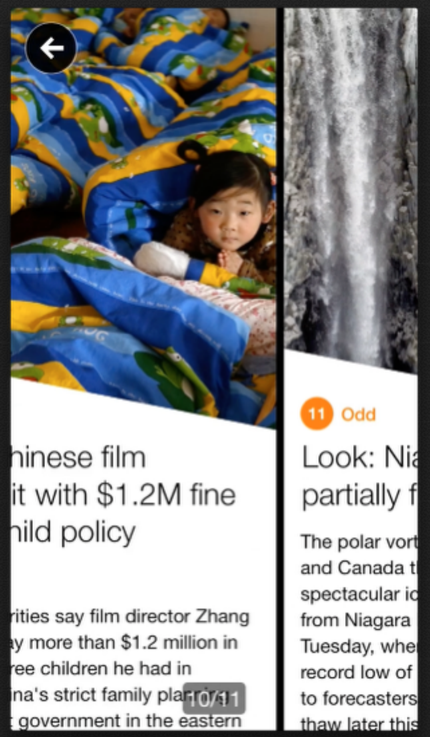
Origami Patches

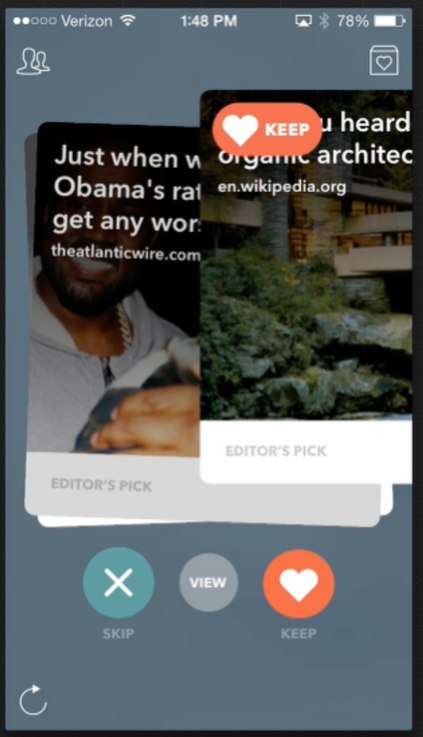
- **Switch**
  Toggles between on and off and remembers its current state.
- **Scroll**
  Lets you allow the user to scroll an image.
- **Bouncy Animation**
  Animate a changing value with a bouncy spring.
- **Classic Animation**
  Animate a changing number with a specified duration and easing curve.
- **Transition**
  Takes a number in the range of 0 to 1 and converts it to a range starting with Start Value and ending in End Value.
- **Color Transition**
  Fades between two colors.
- **Image Transition**
  Fades between two images.
- **Interaction 2**
  Lets you get clicks on Layer patches. It works with Render in Image patches, iterators, and the Phone patch.
- **Layer Group**
  Lets you group layers together.
- **Progress**
  Takes a value in a given range and outputs the progress of where the value is in the range.
- **Hex Color** NEW IN 1.2
  Converts a hex string to a color.
- **Cursor Control** NEW IN 1.2
  Allows you to hide or change the OS X cursor when it's over the viewer window.
- **Layer**
  Lets you put an image on screen and adjust how it shows up.
- **Text Layer**
  Lets you put text on screen.
- **Fill Layer**
  Fills the screen with a color.
- **Button**
  Shows a button with a customizable label and background. Connect it to the Interaction 2 patch to make it interactive.
- **Hit Area**
  Lets you specify an area on screen to get clicks on. Connect it to an Interaction 2 patch to make it interactive.
- **Phone** IMPROVED IN 1.2
  Draws a phone (iPhone, Android or Windows Phone) or iPad with a specified screen image.
- **Browser Chrome** NEW IN 1.2
  Styles the viewer window to look like a web browser.
- **Timer**
  Runs a timer for a specified duration starting with a specified signal. It outputs the state of the timer as well as the time elapsed.
- **Counter 2**
  Represents a number that can be incremented, decremented or set to a specific value.
- **Index Switch** NEW IN 1.2
  Switches between multiple index values and remembers its current index.
- **Velocity** NEW IN 1.2
  Outputs the difference in value between the previous frame and current frame.

Origami Demo





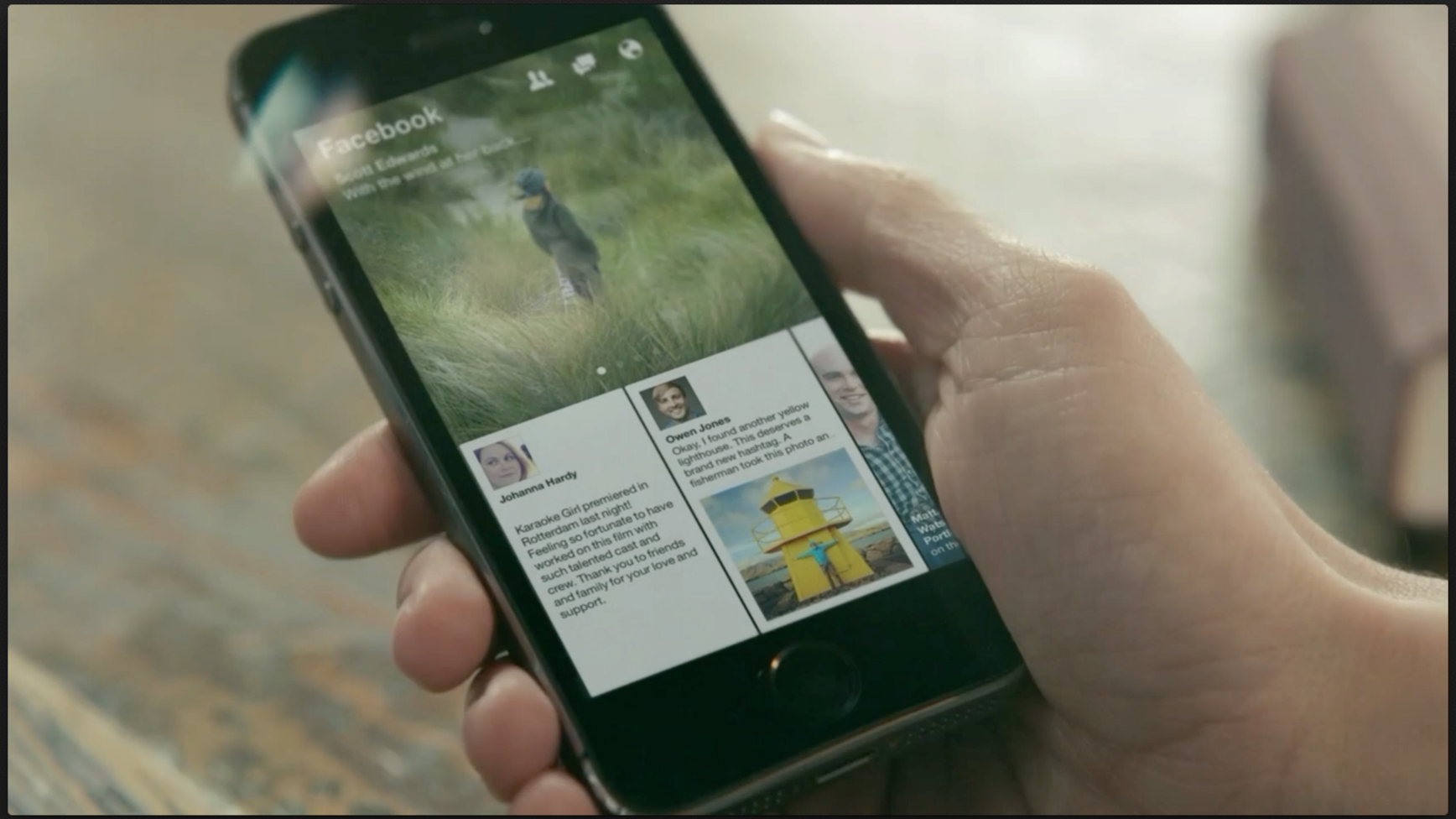




Transitional, dynamic interfaces

Paper

Paper


Facebook
Johanna Hardy
Karaoke Girl premiered in Rotterdam last night! Feeling so fortunate to have worked on this film with such talented cast and crew. Thank you to friends and family for your love and support.
Owen Jones
Okay, I found another yellow lighthouse. This deserves a brand new hashtag. A fisherman took this photo an...

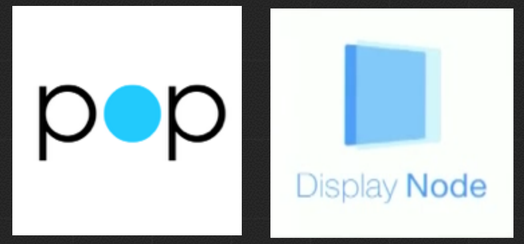
Paper

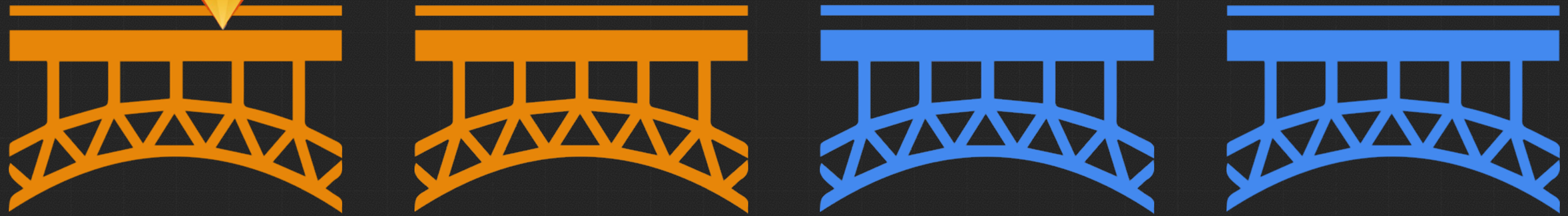
- Had no ship date
- Philosophy of honoring the content users took the time to create / share
- Focus on treating content as a collection of interactable objects, obeying laws of physics
- Many iterations on interaction designs before writing a single line of code
- QC helped team break constraints of traditional desktop software
- QC helped to design things they didn’t know how to build

Paper


Mike Matas
Thanksgiving with the family.
42 Likes 1 Comment
Julie Zhuo, Keegan Jones

Transitional Interfaces - Prototype to Development





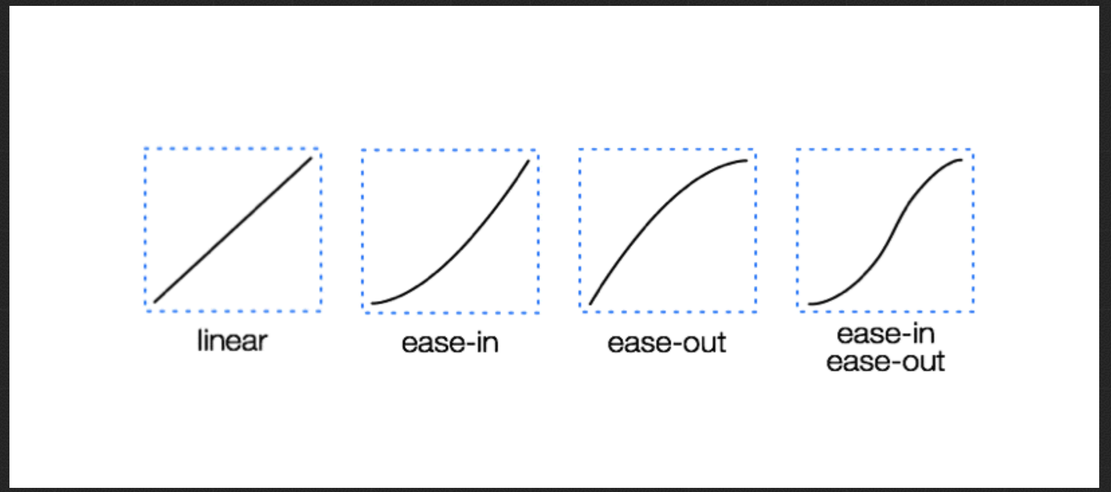
Designer

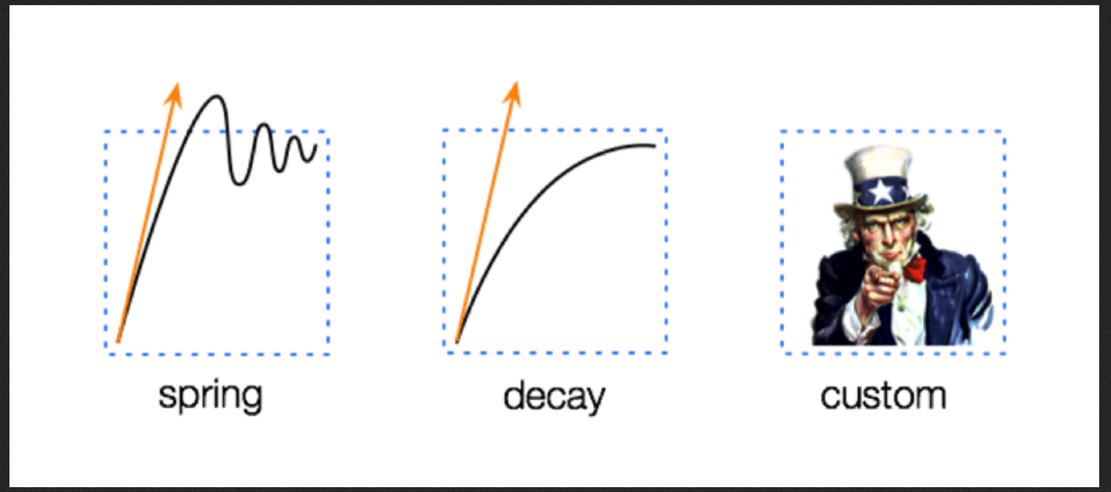
Developer

April 2014

Pop

- Animation Engine
- Dynamic, velocity driven easing
- All scrolling, bouncing, folding effects in Paper
- Captured gesture velocity reflects user intent
- 3 animation class primitives ->
- Custom = extensible
- Animation decoupled from display
- Animate any property of any Objective-C object





Pop

API is almost identical to Core Animation interface

```objc
@interface NSObject (pop)

/* Attach an animation to the layer. */
- (void)pop_addAnimation:(POPAnimation *)anim forKey:(NSString *)key;

/* Remove all animations attached to the layer. */
- (void)pop_removeAllAnimations;

/* Remove any animation attached to the layer for 'key'. */
- (void)pop_removeAnimationForKey:(NSString *)key;

/* Returns an array containing the keys of all animations currently
 * attached to the receiver. */
- (NSArray *)pop_animationKeys;

/* Returns the animation added to the layer with identifier 'key', or nil
 * if no such animation exists. */
- (POPAnimation *)pop_animationForKey:(NSString *)key;

@end
```

Pop

```objc
@interface CALayer

/* Attach an animation to the layer. */
- (void)addAnimation:(CAAnimation *)anim forKey:(NSString *)key;

/* Remove all animations attached to the layer. */
- (void)removeAllAnimations;

/* Remove any animation attached to the layer for 'key'. */
- (void)removeAnimationForKey:(NSString *)key;

/* Returns an array containing the keys of all animations currently
 * attached to the receiver. */
- (NSArray *)animationKeys;

/* Returns the animation added to the layer with identifier 'key', or nil
 * if no such animation exists. */
- (CAAnimation *)animationForKey:(NSString *)key;

@end
```

Core Animation

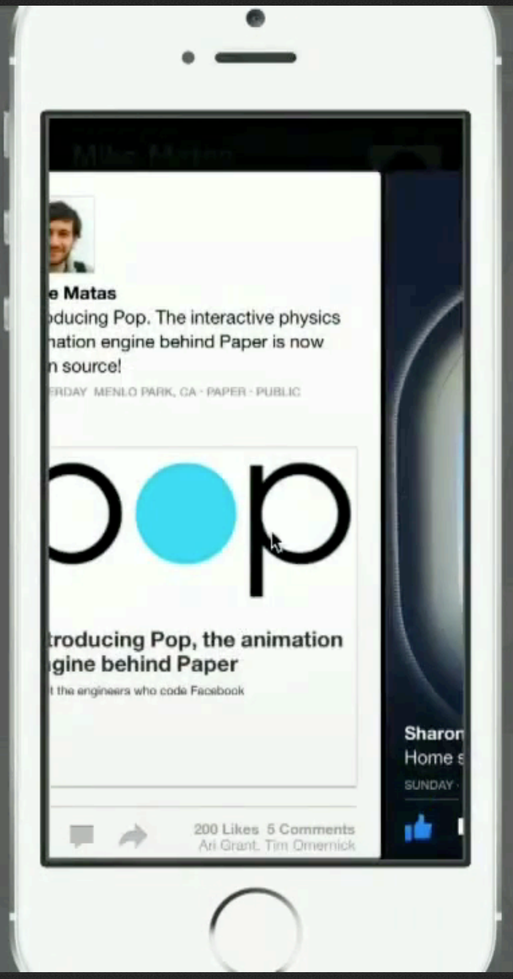
pop

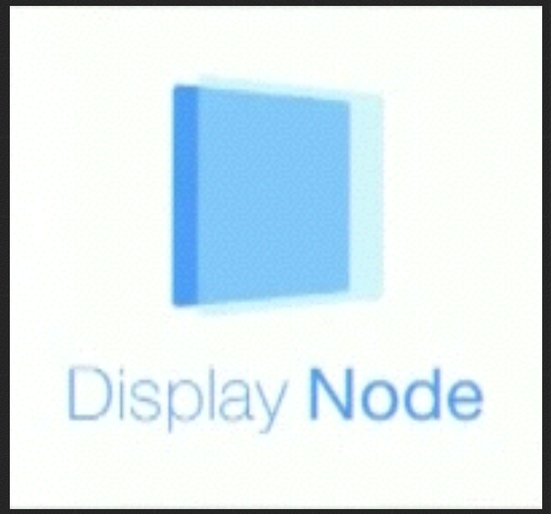
Not enough

CHALLENGE #2: PERFORMANCE

It was immediately obvious we had a major structural problem.

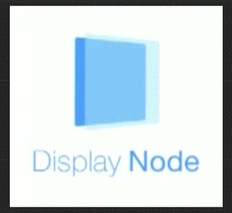
Pop was a huge success, but did not address performance directly.





Asynchronous UI

(coming soon)



Asynchronous UI

- Core Animation is great for handling taps, didFinish: gesture methods, and pre-wired easing curves.
- Not so great for dynamic, interruptible animations with physics
- iOS trained us acceptable response times of 100ms or more.
- With Async UI’s, tolerance for response time is over

waynedahlberg

Thanks

Gist with links - http://d.pr/XIMO

Floppy Diskette

Shameless plug - Free in the App Store