

Corona® SDK

Code less. Play more.



```
display.newImage( "sky.png" )
local ground = display.newImage("ground.png", 0, 400)
local crate = display.newImage("crate.png", 160, 50 );
crate.rotation = 30
```

```
local physics = require("physics")
physics.start()
```

```
physics.addBody( ground, "static" )
physics.addBody( crate, { density=50 } )
```

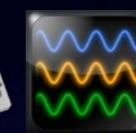
Write once

Publish to top stores



Corona

The ultimate mobile platform



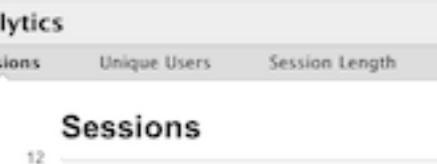
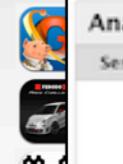
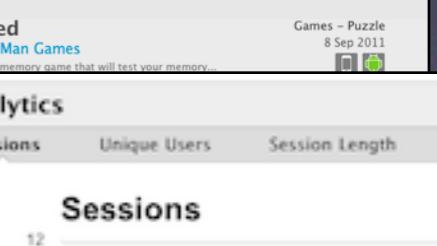
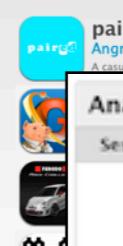
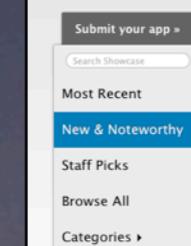
Develop 10x faster

Monetize and distribute

inneractive.



Showcase Top games and apps built with
Corona SDK | #1 IN THE WORLD



Developers Like You

Indies + Game Studios + Publishers + Agencies



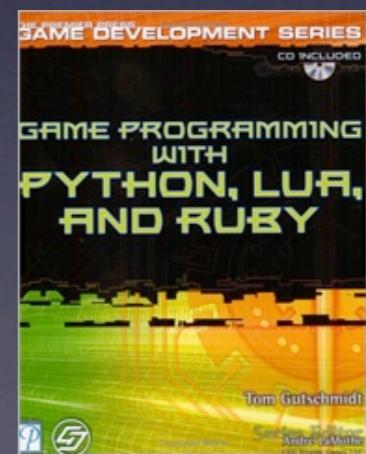
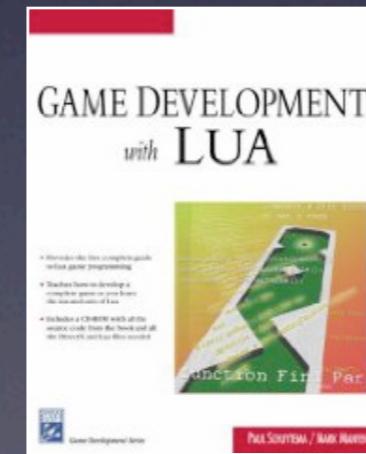
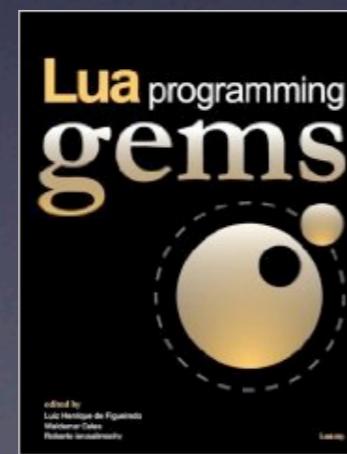
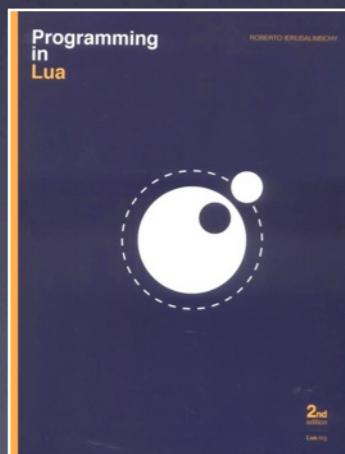
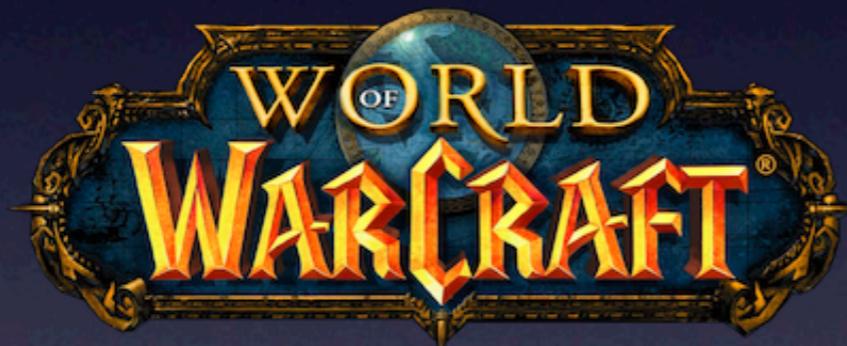
150,000

Developers in the **Corona Community**

1,000,000,000+
app sessions
in 10 months



Lua: An Industry Standard



Breakthrough Simplicity

```

Phone SDK.
// Display "myImage.png"
// -----
// OpenGlestextureAppDelegate.m
// -----
#import "OpenGlestextureAppDelegate.h"
#import "EAGLView.h"
#import "OpenGlestextureViewController.h"

@implementation OpenGlestextureAppDelegate
@synthesize window=_window;
@synthesize viewController=_viewController;

- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
{
    // Override point for customization after application launch.
    self.window.rootViewController = self.viewController;
    return YES;
}

-(void)applicationDidBecomeActive:(UIApplication *)application
{
    /*
        Restart any tasks that were paused (or not yet started) while the application was
        inactive. If the application was previously in the background, optionally refresh the user
        interface.
    */
    [self.viewController drawFrame];
}

-(void)dealloc
{
    [_window release];
    [_viewController release];
    [super dealloc];
}

@end

// -----
// EAGLView.m
// -----
#import <QuartzCore/QuartzCore.h>
#import "EAGLView.h"

@interface EAGLView : UIView<PrivateMethod>
- (void)createFramebuffer;
- (void)deleteFramebuffer;
@end

@implementation EAGLView
@synthesize context;

// You must implement this method
+ (Class)layerClass
{
    return [CAEAGLLayer class];
}

//The EAGL view is stored in the nib file. When it's unarchived it's sent -initWithCoder:.
- (id)initWithCoder:(NSCoder *)coder
{
    self = [super initWithCoder:coder];
    if (self) {
        CAEAGLLayer *eaglLayer = (CAEAGLLayer *)self.layer;

        eaglLayer.opaque = TRUE;
        eaglLayer.drawableProperties = [NSDictionary dictionaryWithObjectsAndKeys:
                                       [NSNumber numberWithBool:FALSE],
                                       kEAGLDrawablePropertyRetainedBacking,
                                       kEAGLColorFormatRGB8,
                                       kEAGLDrawablePropertyColorFormat,
                                       nil];
    }
    return self;
}

- (void)dealloc
{
    [self deleteFramebuffer];
    [context release];
    [super dealloc];
}

- (void)setContext:(EAGLContext *)newContext
{
    if (context != newContext) {
        [self deleteFramebuffer];

        [context release];
        context = [newContext retain];
        [EAGLContext setCurrentContext:nil];
    }
}

- (void)createFramebuffer
{
    if (context && !defaultFramebuffer) {
        [EAGLContext setCurrentContext:context];
        // Create default framebuffer object.
        glGenFramebuffers(1, &defaultFramebuffer);
        glBindFramebuffer(GL_FRAMEBUFFER, defaultFramebuffer);

        // Create color render buffer and allocate backing store.
        glGenRenderbuffers(1, &colorRenderbuffer);
        glBindRenderbuffer(GL_RENDERBUFFER, colorRenderbuffer);
        [context renderbufferStorage:GL_RENDERBUFFER fromDrawable:(CAEAGLLayer *)self.layer];
        glGetRenderbufferParameteriv(GL_RENDERBUFFER, GL_RENDERBUFFER_WIDTH, &framebufferWidth);
        glGetRenderbufferParameteriv(GL_RENDERBUFFER, GL_RENDERBUFFER_HEIGHT, &framebufferHeight);

        glBindFramebuffer(GL_FRAMEBUFFER, GL_COLOR_ATTACHMENT0, GL_RENDERBUFFER, colorRenderbuffer);

        if (glCheckFramebufferStatus(GL_FRAMEBUFFER) != GL_FRAMEBUFFER_COMPLETE)
            NSLog(@"Failed to make complete framebuffer object %x",
                  glCheckFramebufferStatus(GL_FRAMEBUFFER));
    }
}

- (void)deleteFramebuffer
{
    if (context) {
        [EAGLContext setCurrentContext:context];
        if (defaultFramebuffer) {
            glDeleteFramebuffers(1, &defaultFramebuffer);
            defaultFramebuffer = 0;
        }

        if (colorRenderbuffer) {
            glDeleteRenderbuffers(1, &colorRenderbuffer);
            colorRenderbuffer = 0;
        }
    }
}

- (BOOL)presentFramebuffer
{
    BOOL success = FALSE;

    if (context) {
        [EAGLContext setCurrentContext:context];
        glBindRenderbuffer(GL_RENDERBUFFER, colorRenderbuffer);
        success = [context presentRenderbuffer:GL_RENDERBUFFER];
    }

    return success;
}

- (void)layoutSubviews
{
    // The framebuffer will be re-created at the beginning of the next setFramebuffer
    // method call.
    [self deleteFramebuffer];
}

// -----
// OpenGlestextureViewController.m
// -----
#import <QuartzCore/QuartzCore.h>
#import "OpenGlestextureViewController.h"
#import "EAGLView.h"

@interface OpenGlestextureViewController : UIViewController
@property (nonatomic, retain) EAGLContext *context;
@property (nonatomic, assign) CADisplayLink *displayLink;
- (void) loadTexture;
@end

@implementation OpenGlestextureViewController
@synthesize animating, context, displayLink;

- (void)awakeFromNib
{
    EAGLContext *aContext = [[EAGLContext alloc] initWithAPI:kEAGLRenderingAPIOpenGL];
    if (!aContext)
        NSLog(@"Failed to create ES context");
    else if (![EAGLContext setCurrentContext:aContext])
        NSLog(@"Failed to set ES context current");

    self.context = aContext;
}

- (void)loadTexture
{
    [aContext release];
    [(EAGLView *)self.view setCurrentContext:context];
    [(EAGLView *)self.view setFramebuffer];
    [self loadTexture];
}

- (void)loadTexture
{
    glEnable(GL_TEXTURE_2D);
    glEnable(GL_BLEND);
    glBlendFunc(GL_SRC_ALPHA, GL_ONE_MINUS_SRC_ALPHA);

    glGenTextures(1, &textureID);
    glBindTexture(GL_TEXTURE_2D, textureID);
    glTexParameteri(GL_TEXTURE_2D,GL_TEXTURE_MIN_FILTER,GL_LINEAR);
    glTexParameteri(GL_TEXTURE_2D,GL_TEXTURE_MAG_FILTER,GL_LINEAR);

    NSString *path = [[NSBundle mainBundle] pathForResource:@"myImage" ofType:@"png"];
    NSData *texData = [[NSData alloc] initWithContentsOfFile:path];
    UIImage *image = [[UIImage alloc] initWithData:texData];

    GLuint width = CGImageGetWidth(image.CGImage);
    GLuint height = CGImageGetHeight(image.CGImage);
    CGColorSpaceRef colorSpace = CGColorSpaceCreateDeviceRGB();
    void *imageData = malloc( height * 4 );
    CGContextRef image_context = CGBitmapContextCreate( imageData, width, height, 8, 4 * width, colorSpace, kCGImageAlphaPremultipliedLast | kCGBitmapByteOrder32Big );
    CGColorSpaceRelease( colorSpace );
    CGContextClearRect( image_context, CGRectMake( 0, 0, width, height ) );
    CGContextTranslateCTM( image_context, 0, height - height );
    CGContextDrawImage( image_context, CGRectMake( 0, 0, width, height ), image.CGImage );

    glTexImage2D(GL_TEXTURE_2D, 0, GL_RGBA, width, height, 0, GL_RGBA, GL_UNSIGNED_BYTE,
                 imageData);
}

- (void)drawFrame
{
    [EAGLView setCurrentContext:nil];
    [context release];
    [super dealloc];
}

- (void)viewDidUnload
{
    [super viewDidUnload];
    // Tear down context.
    if ([EAGLContext setCurrentContext] == context)
        [EAGLContext setCurrentContext:nil];
    self.context = nil;
}

- (void)drawFrame
{
    [EAGLView setCurrentContext:context];
    // Replace the implementation of this method to do your own custom drawing.
    static const GLfloat squareVertices[] = {
        -0.5f, -0.33f,
        0.5f, -0.33f,
        -0.5f, 0.33f,
        0.5f, 0.33f,
    };
    static const GLfloat texCoords[] = {
        0.0, 1.0,
        1.0, 1.0,
        0.0, 0.0,
        1.0, 0.0
    };

    glClearColor(0.0f, 0.0f, 0.0f, 1.0f);
    glClear(GL_COLOR_BUFFER_BIT);

    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    glMatrixMode(GL_MODELVIEW);
    glLoadIdentity();

    glVertexPointer(2, GL_FLOAT, 0, squareVertices);
    glEnableClientState(GL_VERTEX_ARRAY);
    glTexCoordPointer(2, GL_FLOAT, 0, texCoords);
    glEnableClientState(GL_TEXTURE_COORD_ARRAY);

    glDrawArrays(GL_TRIANGLE_STRIP, 0, 4);
}

- (void)presentFramebuffer
{
    [(EAGLView *)self.view presentFramebuffer];
}

```

display.newImage("myImage.png")

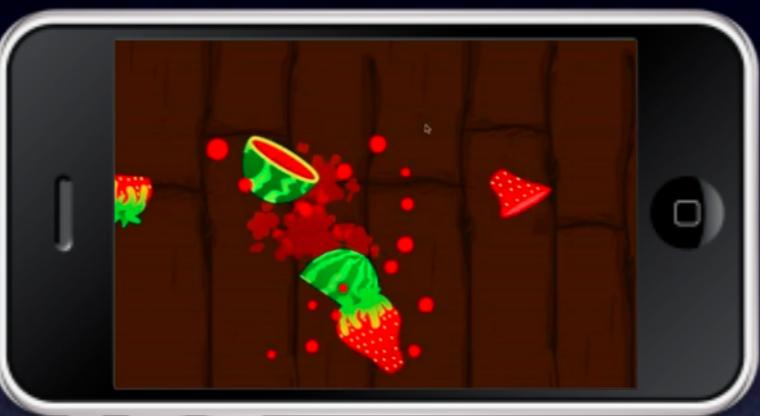


Develop 10x Faster

“Angry Birds”



“Fruit Ninja”



“Tiny Wings”



36 hours

code+graphics+sound
(complete 2 level game)

14 hours

code+graphics+sound
(gameplay only)

12 hours

code+graphics+sound
(gameplay only)

“Developing directly in Xcode would have been at least **5x** more code than Corona”

– Unicorn Labs, Top 20 iPad eBook

Expand your reach.

Corona SDK builds apps for all the major platforms. [Get Started ▾](#)

iOS



android



kindle fire



nook



Tons of Features/APIs

- Content scaling
- Multi-resolution images
- Simulator (instant refresh)
- Cloud services
- And more!



Not Just for Games



HTML5 + OpenGL

Cross-device/Skinnable UI

Branded Apps



#1 on All Major Markets

Reach for the sky.

Corona apps have reached
the top in all major markets.

[Build the next mobile hit ▶](#)

Bubble Ball
(Nay Games, LLC.)

#1
Apple App Store
(January 2011)



Blast Monkeys
(Yobonja)

#1
Android Market
(July 2011)



Corona Indie Bundle
(Various)

#1
Amazon Appstore
(December 2011)



Corona® SDK CODE LESS. PLAY MORE.



Your app is next!



Anyone Can Be #1... with Corona



D All Things Digital. News, and

Mobilized by Isa Fried

About Isa
Isa Fried has joined All Things Digital to cover mobile news and reviews, including tablets, smartphones and even some phones of average intelligence. Read more → [@isa_fried](#) [Follow Isa on Twitter](#)

Ethics Statement
Here is a statement of my ethics and coverage policies. It is more than most of you want to see, but, in the age of judgment of the media, I am saying it all out.

JOB LISTINGS
[VIEW ALL POST A JOB](#)
Manager, Engineering T-Mobile App Development | iPhone

News, and

abcNEWS / Technology

Home Video News Politics World Health Entertainment Money Tech Travel Photo News

BEST AC/TECH: Bestobiles | Tech This Out! | Various iPhone | One-Way Mission to Mars? | iPhone Apps for Music & More

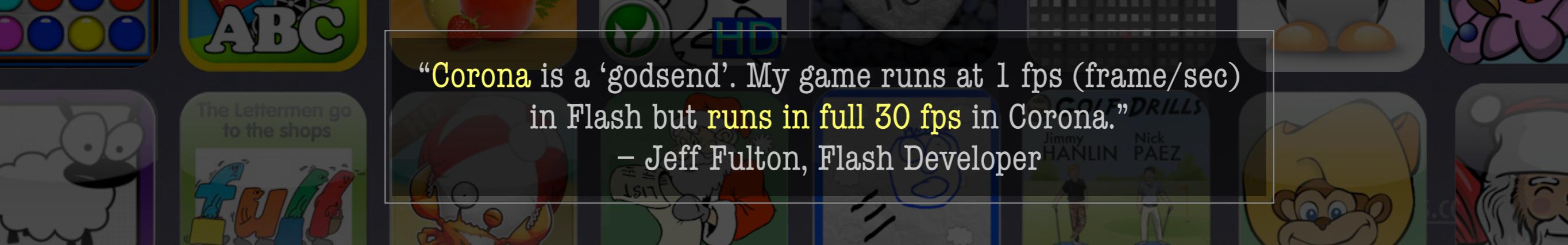
WATCH Video — [WATCH](#) Dismayed Value Falls in Pioneering iPhone

[WATCH](#) Angry Birds Invade in iPhone Games

[WATCH](#) iPhone 4S: The New iPhone 4

[WATCH](#) iPhone

Take a Look at What's Possible



"Corona is a 'godsend'. My game runs at 1 fps (frame/sec) in Flash but runs in full 30 fps in Corona."
– Jeff Fulton, Flash Developer