



Javascript in iOS native apps

Jordi Giménez

jordi@mobilejazz.cat

Problem

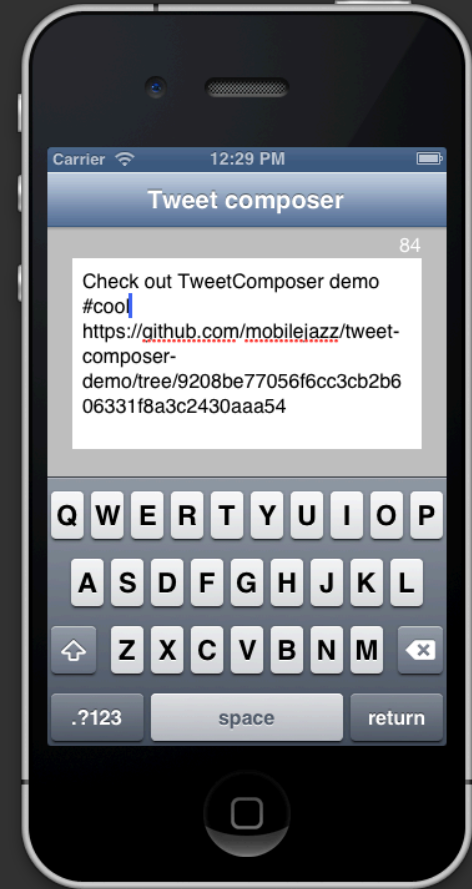
- Reusing code in several platforms
- Existing code to be integrated on our application
- HTML5 thing does not work as advertised
- Xcode is the only official way to compile code that can go to the AppStore => {C,C++,Objective-C}

Solution

- Apple developed WebKit: JavaScript engine in Safari
- **JavaScriptCore** is part of WebKit
 - Open-source library
 - Compilable in Xcode

Example

- A Twitter application usually provides a character counter.
- However, links need to be treated specially because they will be shortened.
- Twitter open-sourced their text processing library (twitter-text.js)



Nasty bits

- Create a JavaScript execution context

```
// create a JS context  
JSValueRef exception = NULL;  
context = JSGlobalContextCreate(NULL);
```

Nasty bits

- Load your library

```
// load twitter-text.js to a string
NSError * error = nil;
NSURL* url = [[NSBundle mainBundle] URLForResource:@"twitter-text" withExtension:@"js"];
NSString* source = [NSString stringWithContentsOfURL:url encoding:NSUTF8StringEncoding error:&error];
if(error != nil) return nil;
// evaluate the string
JSStringRef sourceJS = JSStringCreateWithUTF8CString([source UTF8String]);
JSEvaluateScript(context, sourceJS, NULL, NULL, 0, &exception);
if(exception != nil) return nil;
```

Nasty bits

- Create functions

```
NSString * script =
@"var text = arguments[0];"
@"var tcoLength = arguments[1];"
@"var urls = twttr.txt.extractUrls(text);"
@"var total = text.length;"
@"for(var i=0; i<urls.length; i++) {"
@"  var url = urls[i];"
@"  total -= url.length;"
@"  total += tcoLength;"
@"  if(url.indexOf('https://') == 0)" // HTTPS URL case
@"    total ++;"
@"}"
@"return total;";

JSStringRef scriptJS = JSStringCreateWithUTF8CString([script UTF8String]);
lengthAferShorteningFn = JSObjectMakeFunction(context, NULL, 0, NULL, scriptJS, NULL, 1, &exception);
JSStringRelease(scriptJS);
if(exception != nil) { [self logException:exception]; return 0; }
```

Nasty bits

- Call functions

```
JSValueRef exception = NULL;
const int NARGS = 2;
// arguments
JSValueRef arguments[NARGS];
arguments[0] = JSValueMakeString(context, JSStringCreateWithUTF8CString([text UTF8String]));
arguments[1] = JSValueMakeNumber(context, tcoUrlLength); // tcoLength

// run
JSValueRef resultJS = JSObjectCallAsFunction(context, lengthAferShorteningFn, NULL, NARGS, arguments, &exception);
if(exception != nil) { [self logException:exception]; return 0; }

// get result
double result = JSValueToNumber(context, resultJS, &exception);
if(exception != nil) { [self logException:exception]; return 0; }

return result;
```


References

- Sample code:
<https://github.com/mobilejazz/tweet-composer-demo>
- Objective-C wrapper with some added stuff (not tested)
<https://github.com/siuying/EvalJS>