Appcelerator Titanium Mobile

FLEXIBILITY

VS

PERFORMANCE

Olivier Morandi

MOBILE DEVELOPER CONFERENCE 24 – 25 MAGGIO 2012

WHYMCA
THE MOBILE REVOLUTION

$ whoami

Olivier Morandi

Software engineer


http://titaniumninja.com


olivier.morandi@gmail.com


@olivier_morandi


https://github.com/omorandi

Titanium Mobile

- Rapid development & prototyping tool
- Single high level language: JS
- Multiple deployment platforms (iOS, Android, ...)
- We can leverage an ever growing set of JS libraries and modules (web, node.js, etc.)
- Possibility to extend the framework with native modules

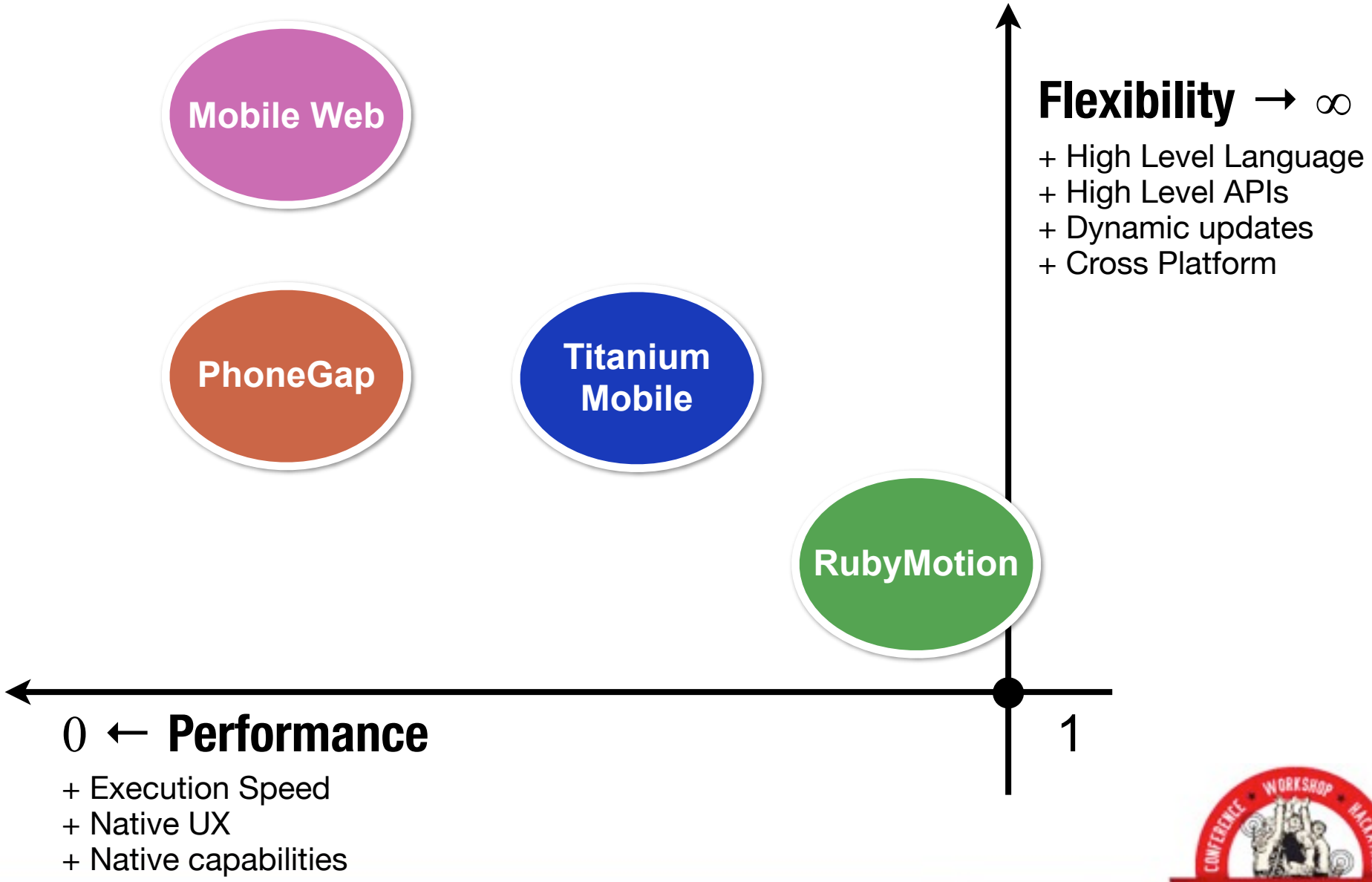
Titanium Apps are NATIVE

Can we also expect a native User Experience?

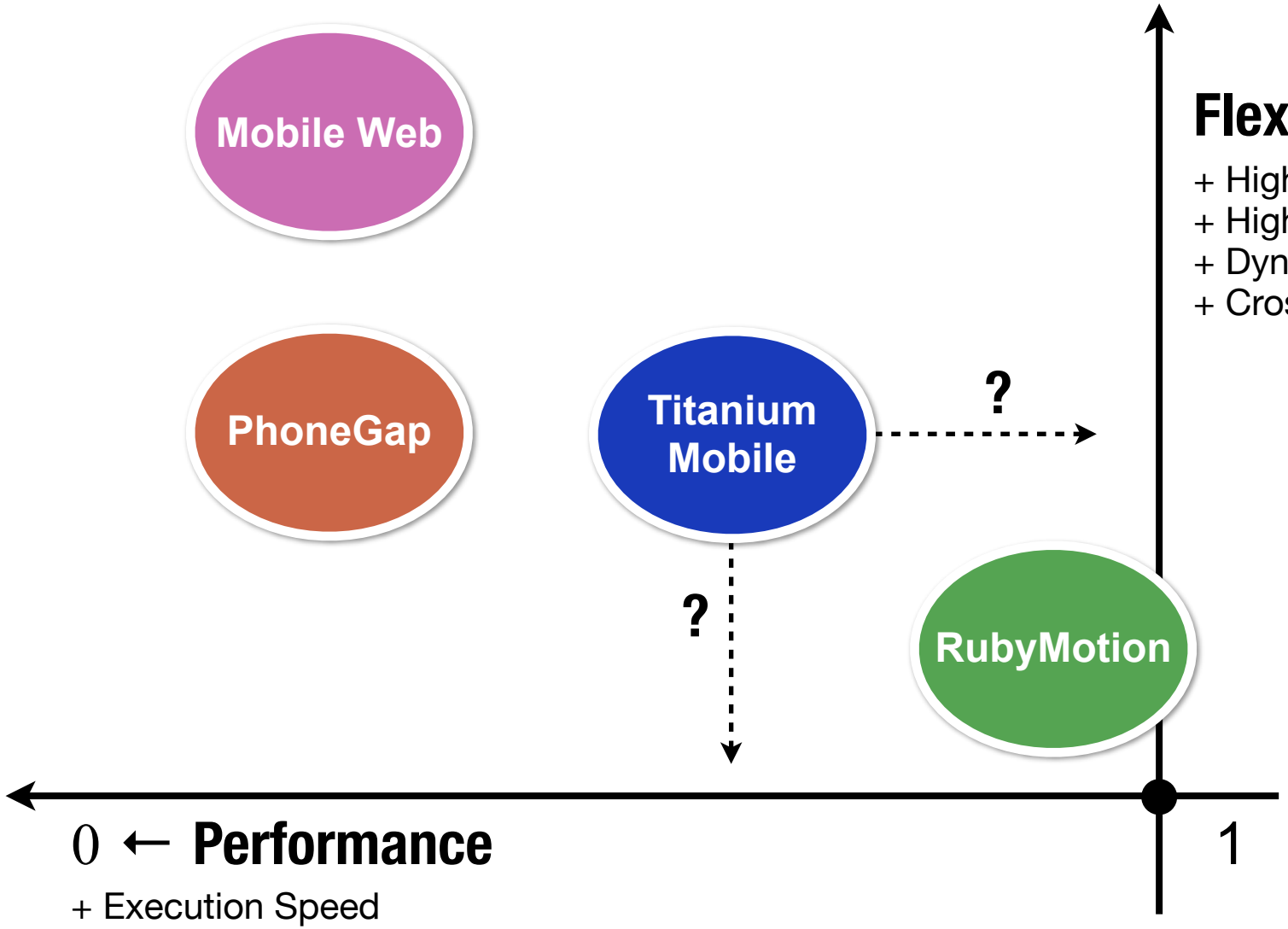
Titanium 2.0

- Improved stability
- Improved platform parity
- Improved performance
- Appcelerator Cloud Services

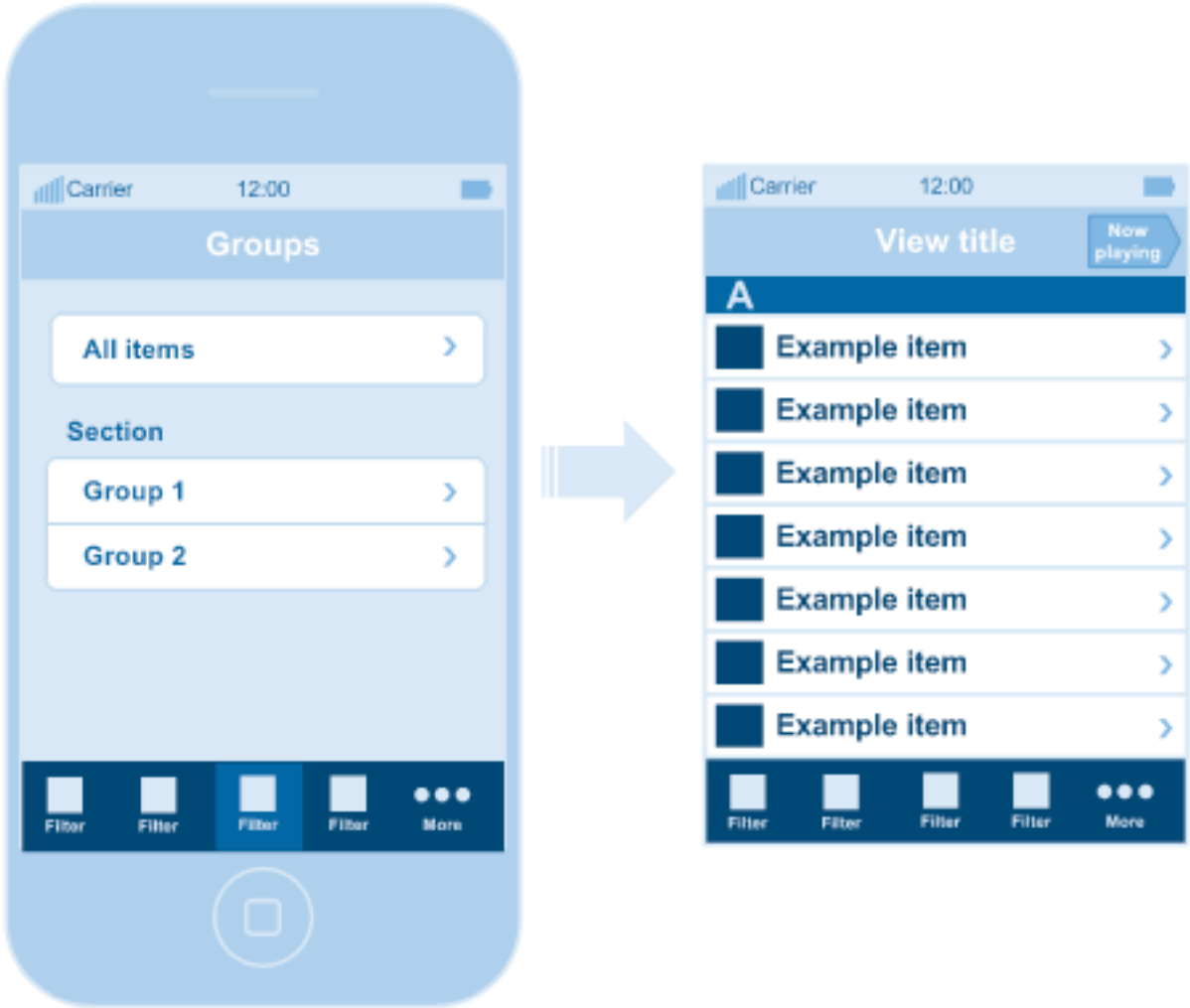
Some Mobile Dev Tools



Some Mobile Dev Tools



An example: Ti.UI.TableView

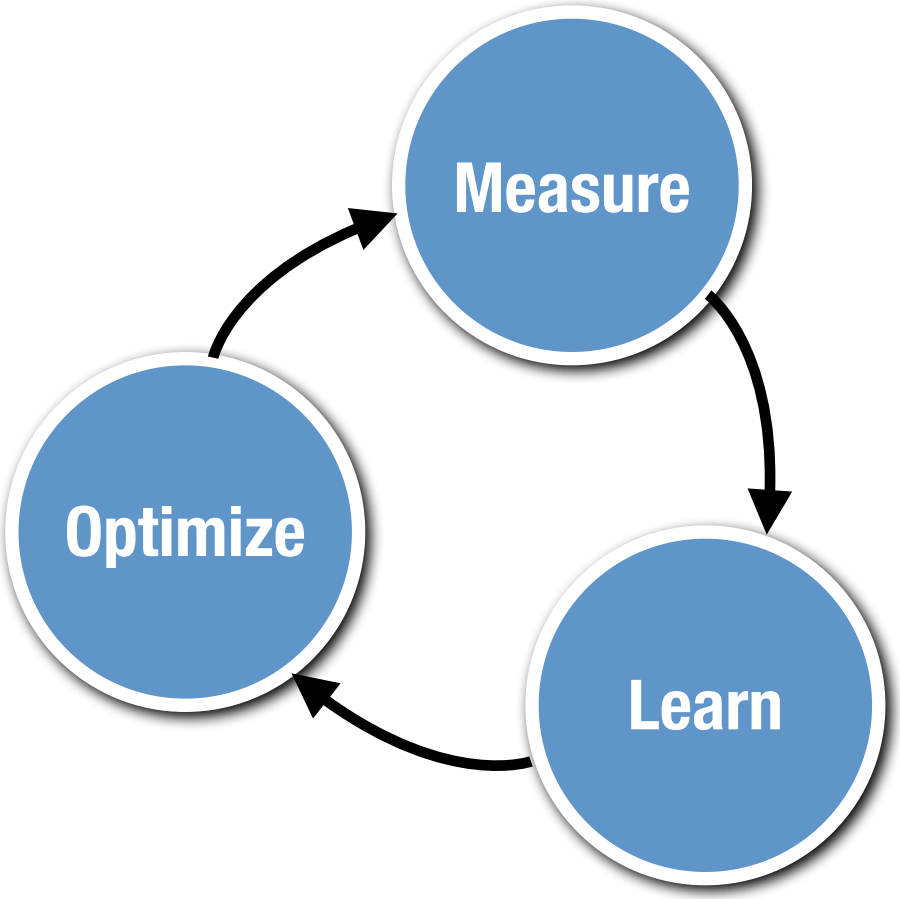


Common Problems

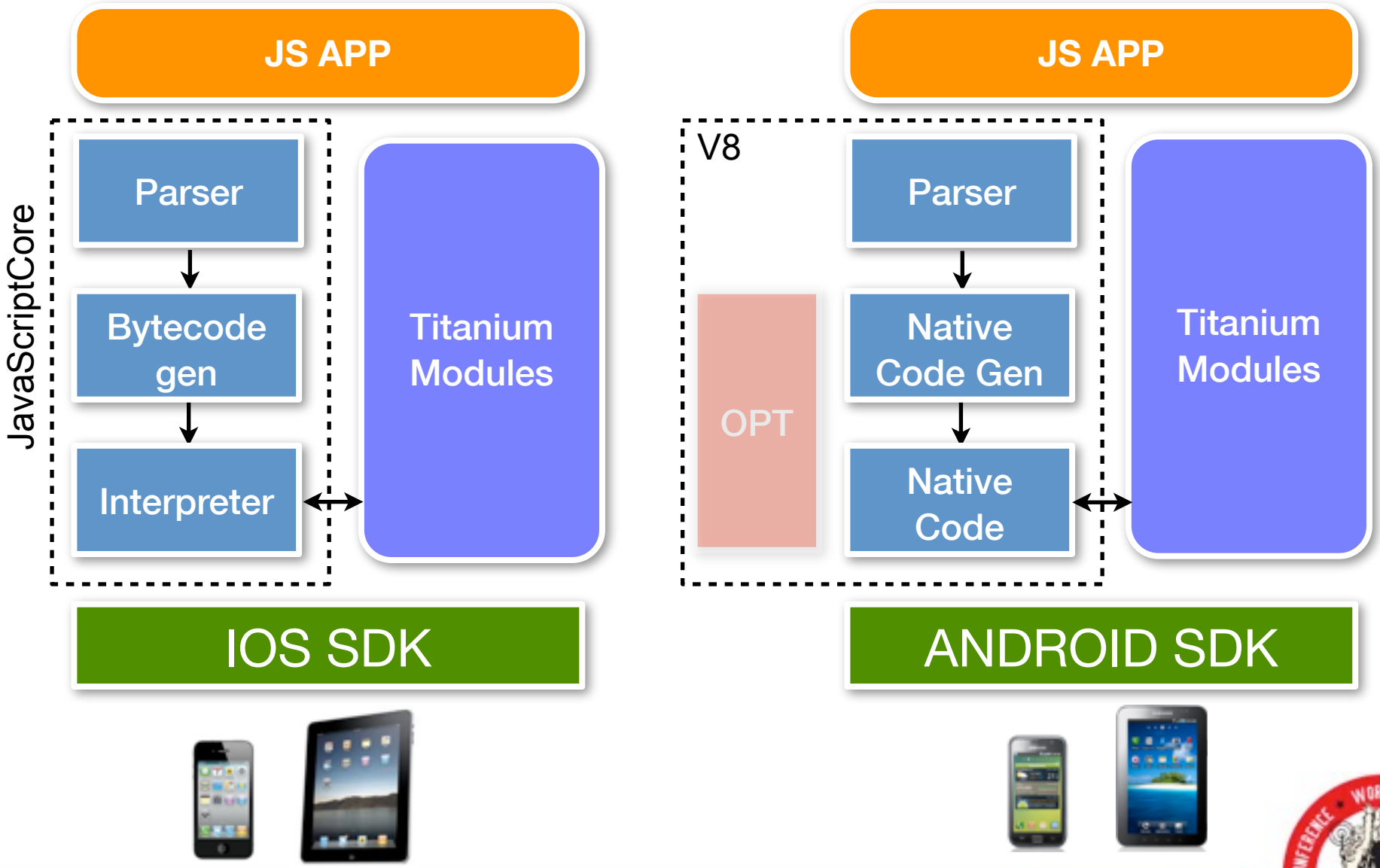
- Slow transitions
  - the table view takes ages to show up
- Choppy scrolling
  - Everybody wants super-slick animations, isn't it?

Towards Increased Performance

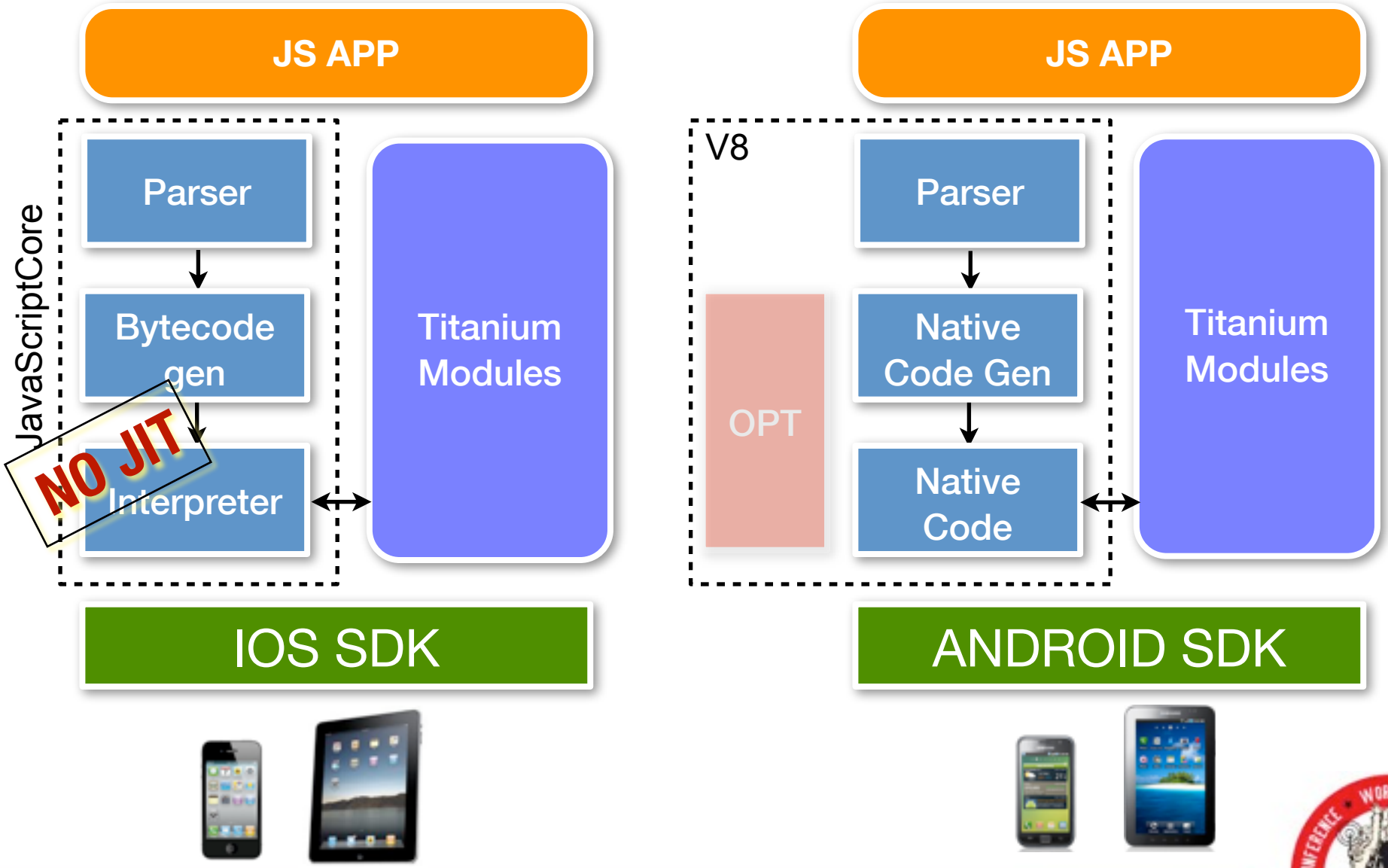
- Understand the tools, then:



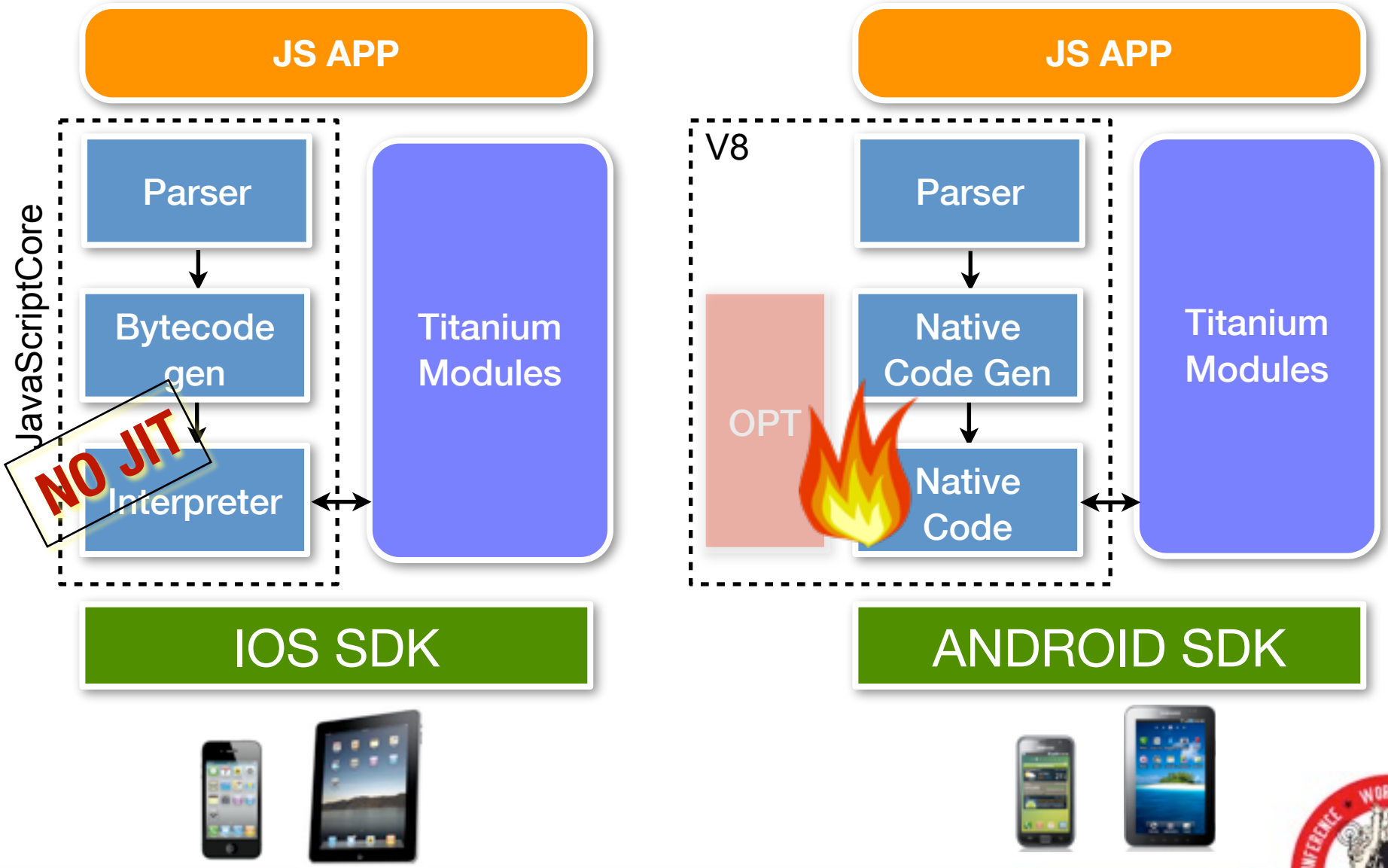
Runtime Environment



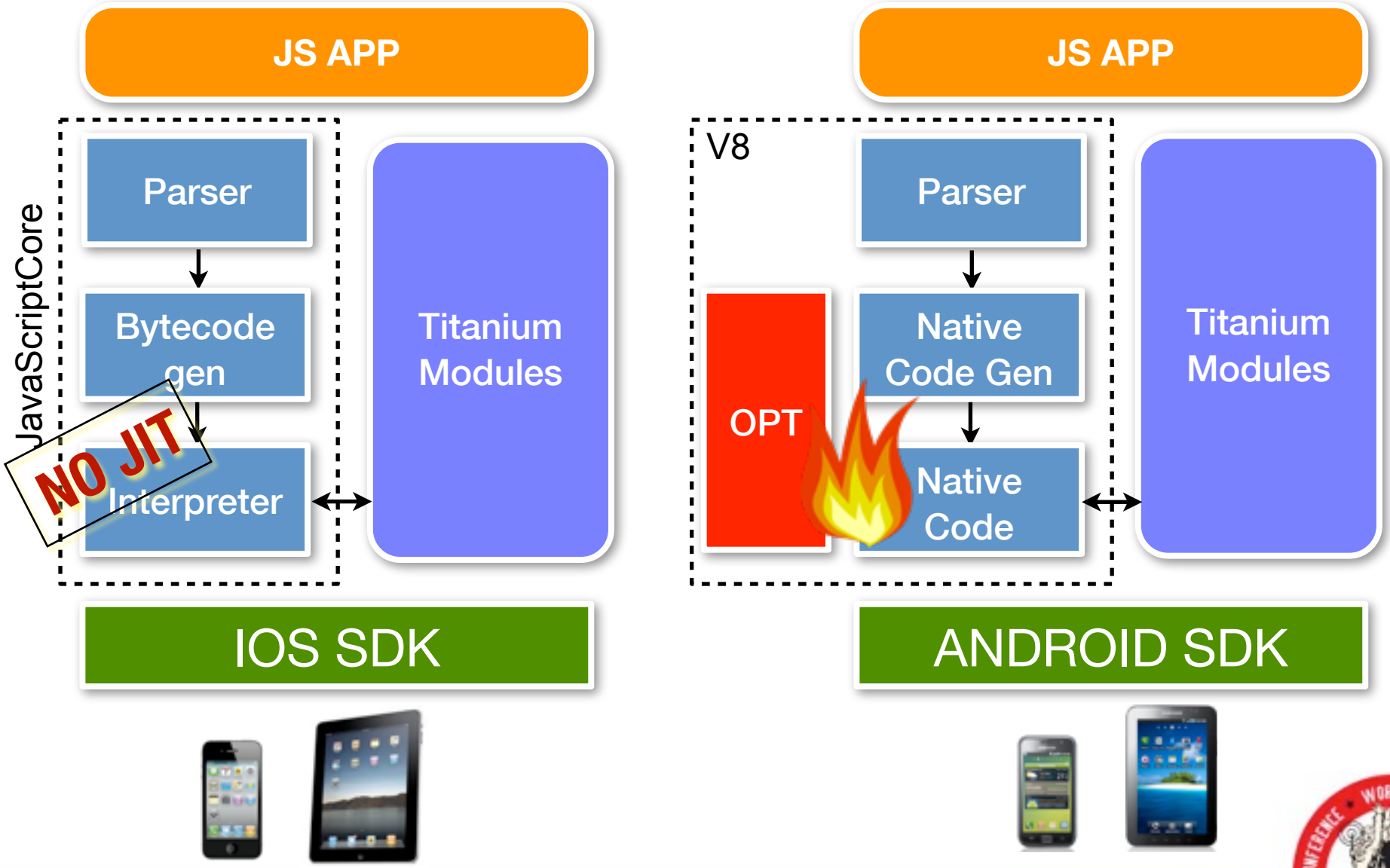
Runtime Environment



Runtime Environment



Runtime Environment



Measure: which runs faster?

```
function processRow(row) {
    return {title: row.info};
}
```

1

```
var rows = [];
for (var r = 0; r < array.length; r++) {
  rows.push({title: array[r].info});
}
```

2

```
var rows = [];
for (var r = 0; r < array.length; r++) {
  rows.push(processRow(array[r]));
}
```

3

```
var len = array.length;
var rows = [];
for (var r = 0; r < len; r++) {
  rows.push({title: array[r].info});
}
```

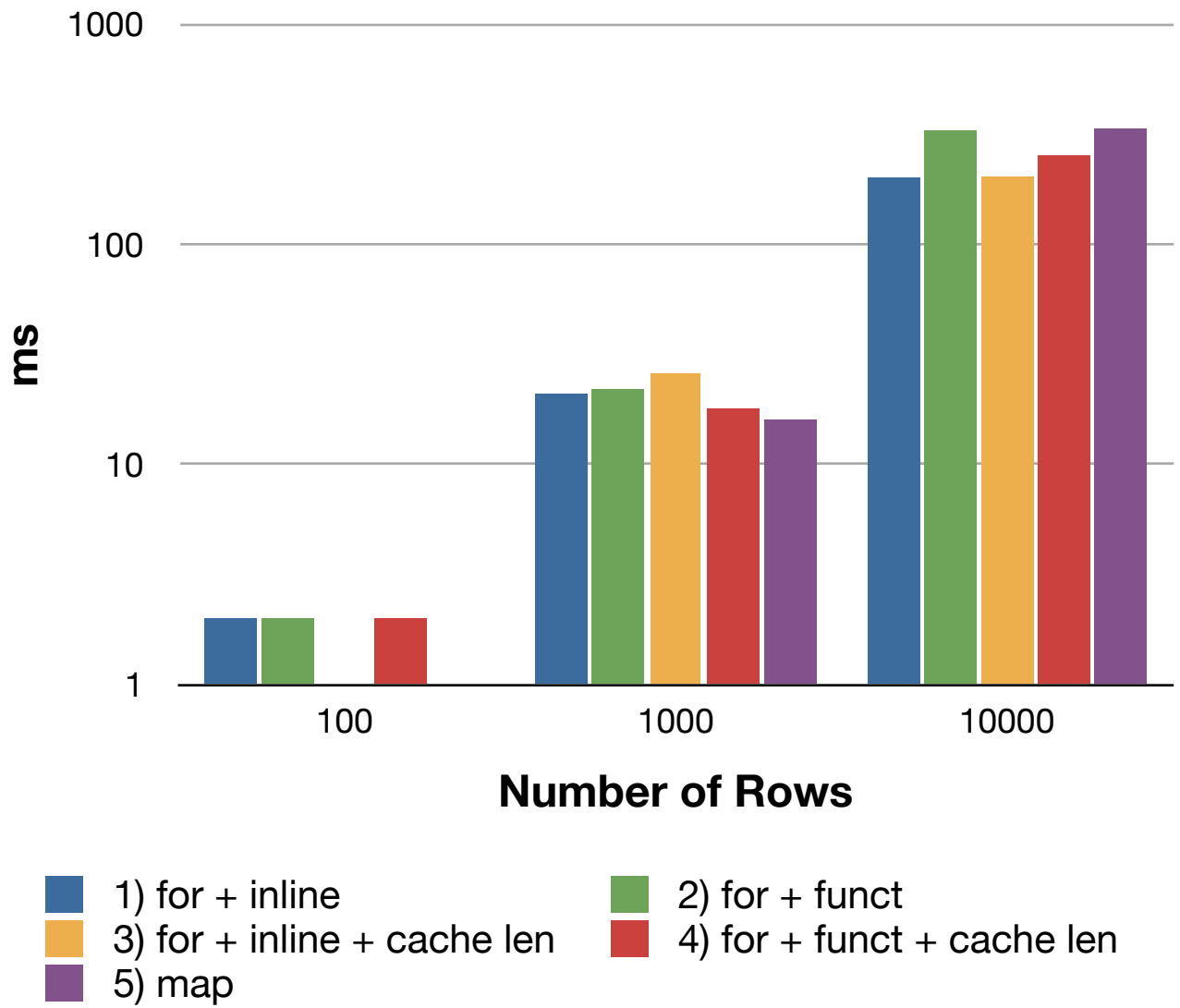
4

```
var len = array.length;
var rows = [];
for (var r = 0; r < len; r++) {
  rows.push(processRow(array[r]));
}
```

5

```
var rows = array.map(processRow);
```

Tests on iPhone 4



Which runs faster?

```
function processRow2(row) {
    return Ti.UI.createTableViewRow({title: row.info});
}
```

1

```
var rows = [];
for (var r = 0; r < array.length; r++) {
  rows.push(Ti.UI.createTableViewRow(
    {title: array[r].info}));
}
```

2

```
var rows = [];
for (var r = 0; r < array.length; r++) {
  rows.push(processRow2(array[r]));
}
```

3

```
var len = array.length;
var rows = [];
for (var r = 0; r < len; r++) {
  rows.push(Ti.UI.createTableViewRow(
    {title: array[r].info}));
}
```

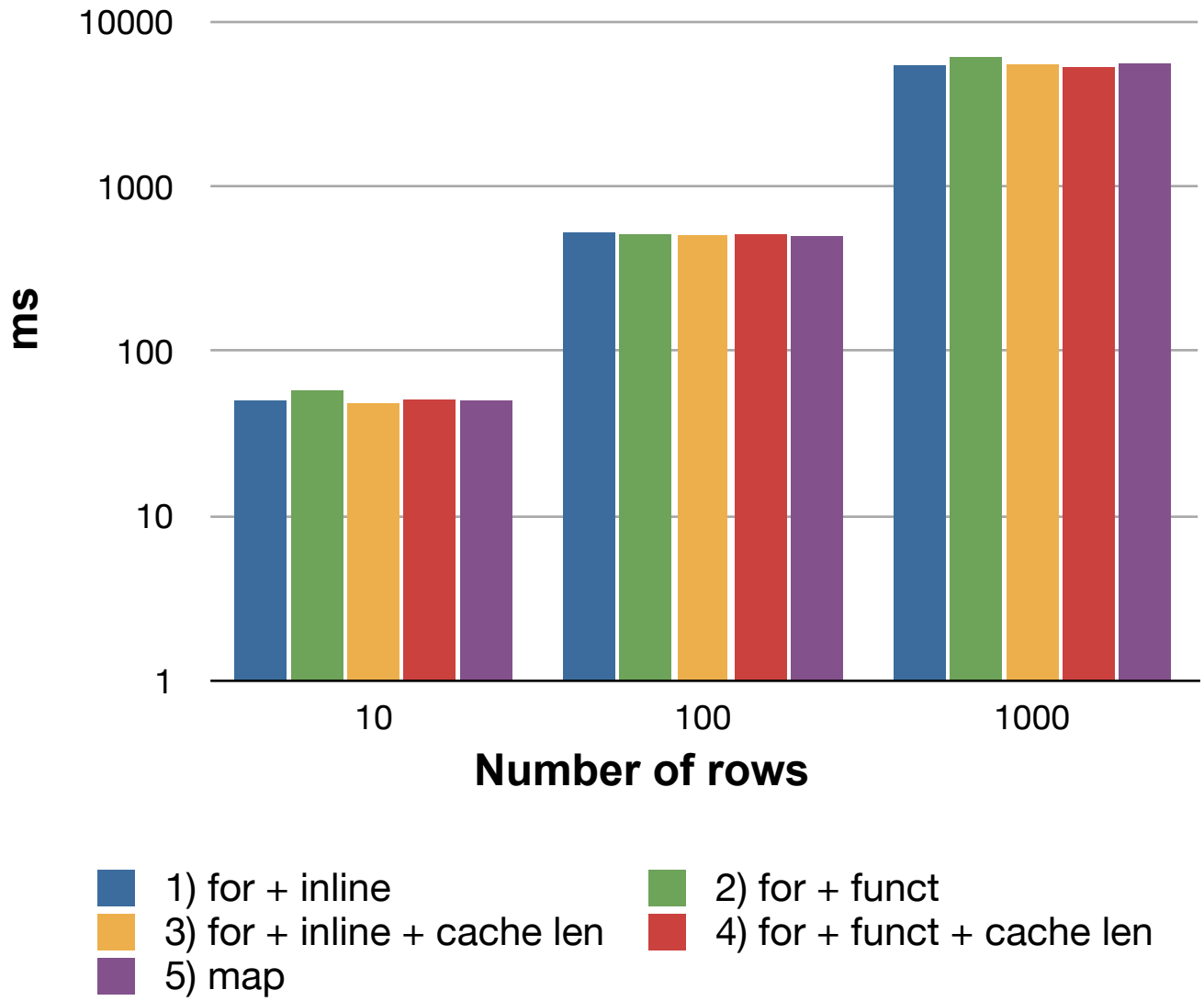
4

```
var len = array.length;
var rows = [];
for (var r = 0; r < len; r++) {
  rows.push(processRow2(array[r]));
}
```

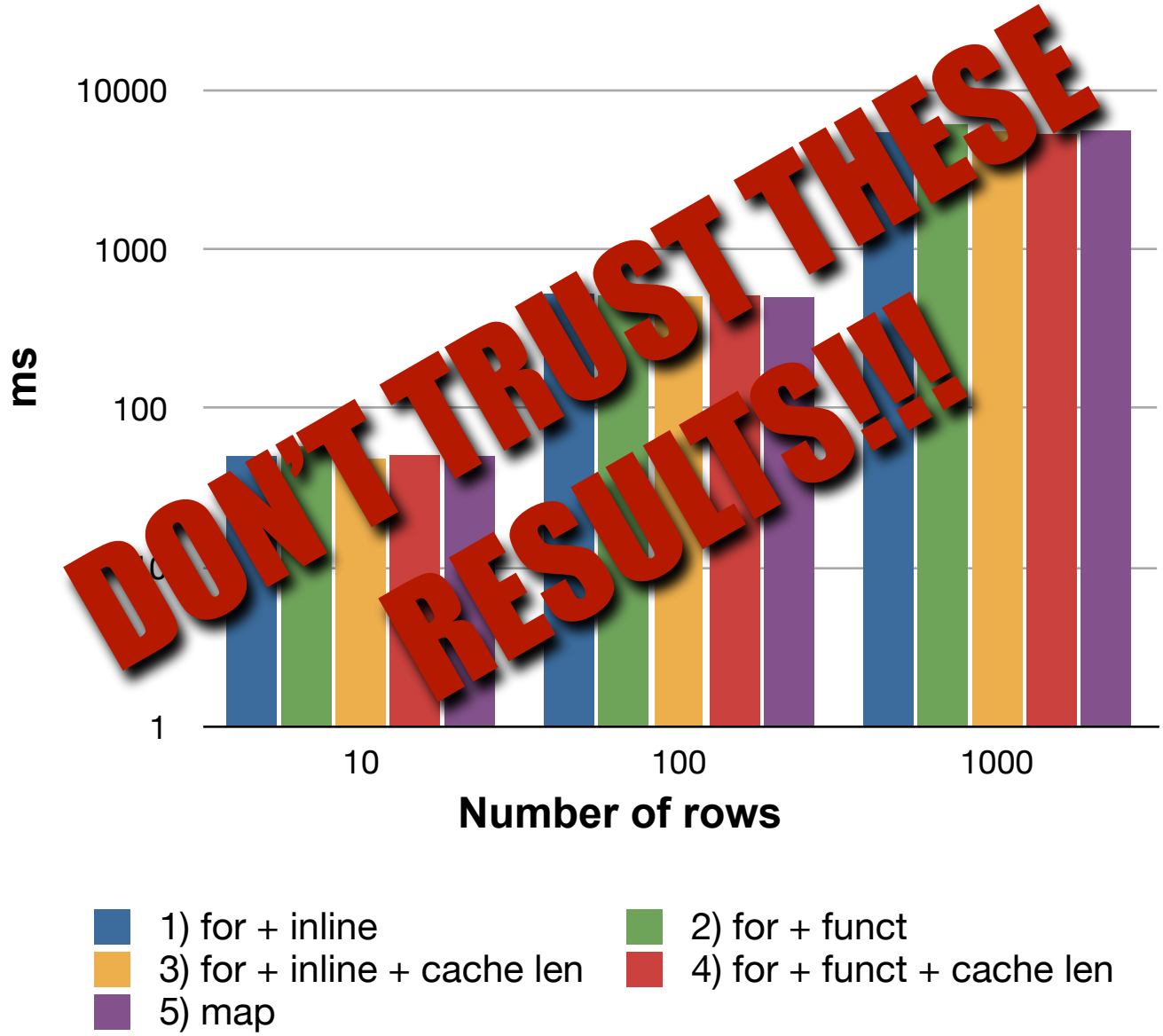
5

```
var rows = array.map(processRow2);
```

Tests on iPhone 4



Tests on iPhone 4



WHY?

- They’re not generalizable
- Your code won’t look like the one presented here
- **Do your own measurements**

Is table creation slow?

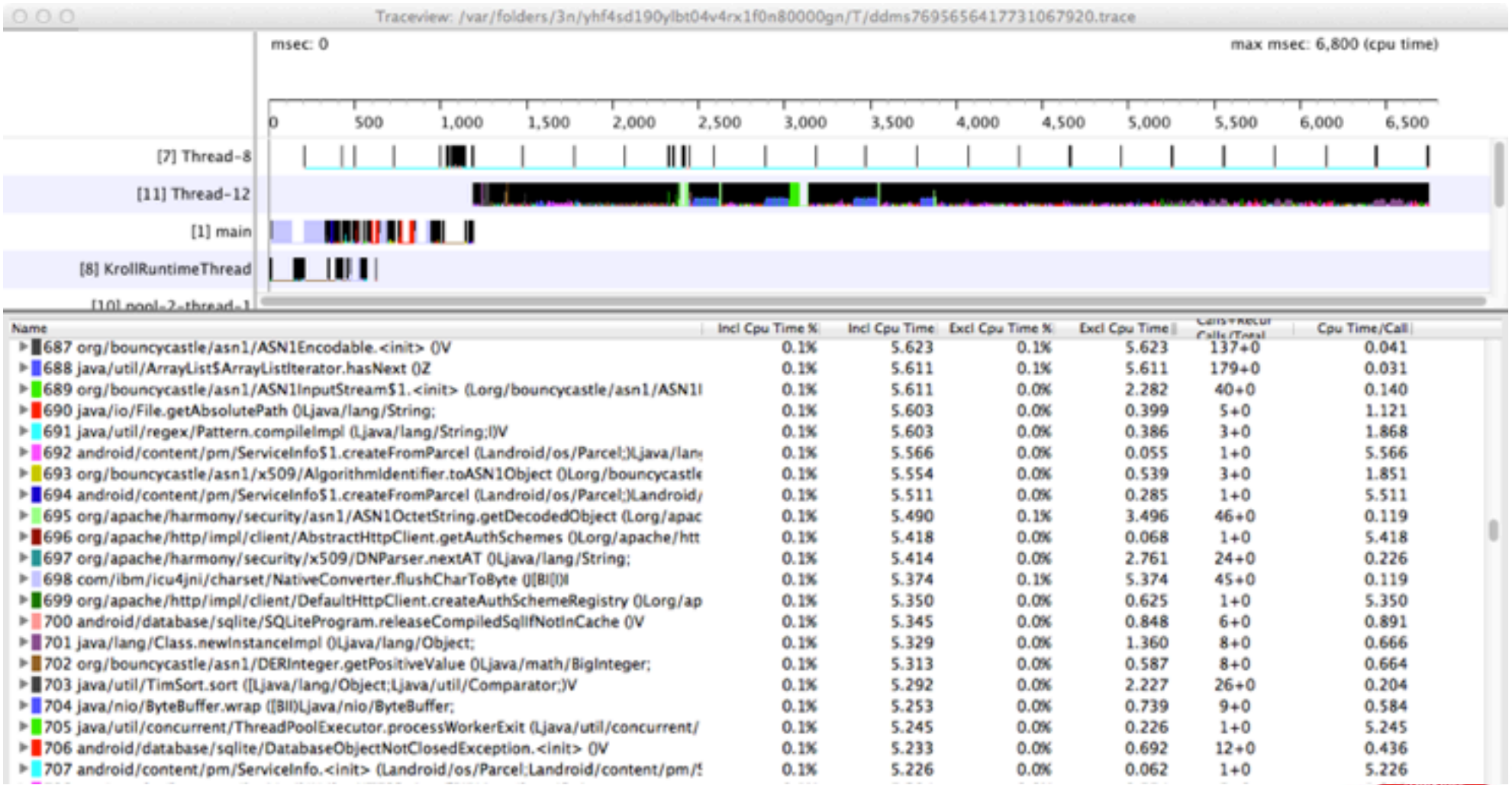
- **Obtaining some rough figures:**

```
var start = new Date().getTime();

//your code goes here

var end = new Date().getTime();
Ti.API.info('elapsed ms: ' + (end - start));
```

Are there more accurate tools?

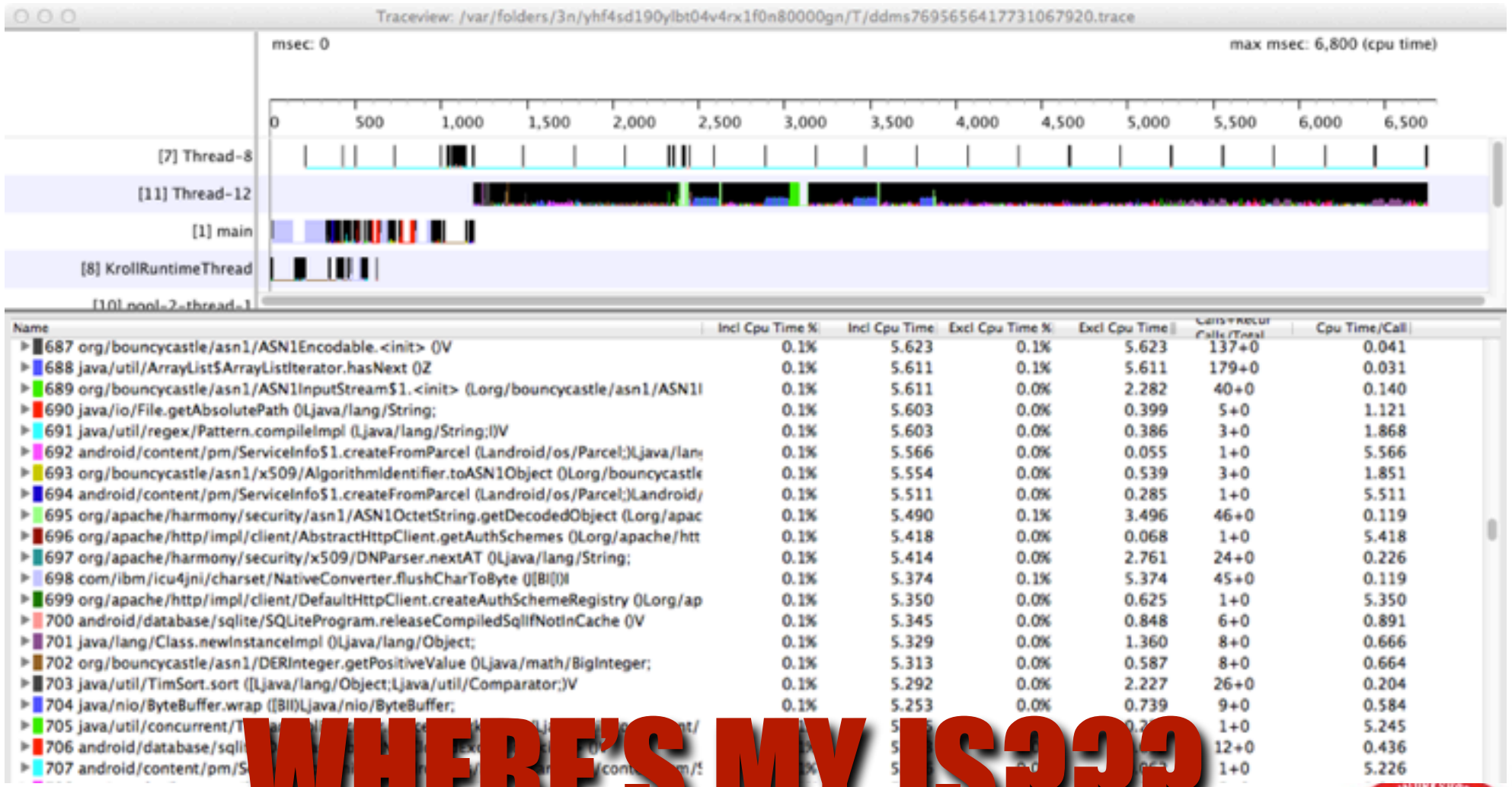
- mmm.... AFAIK NO!

Android DDMS + Traceview



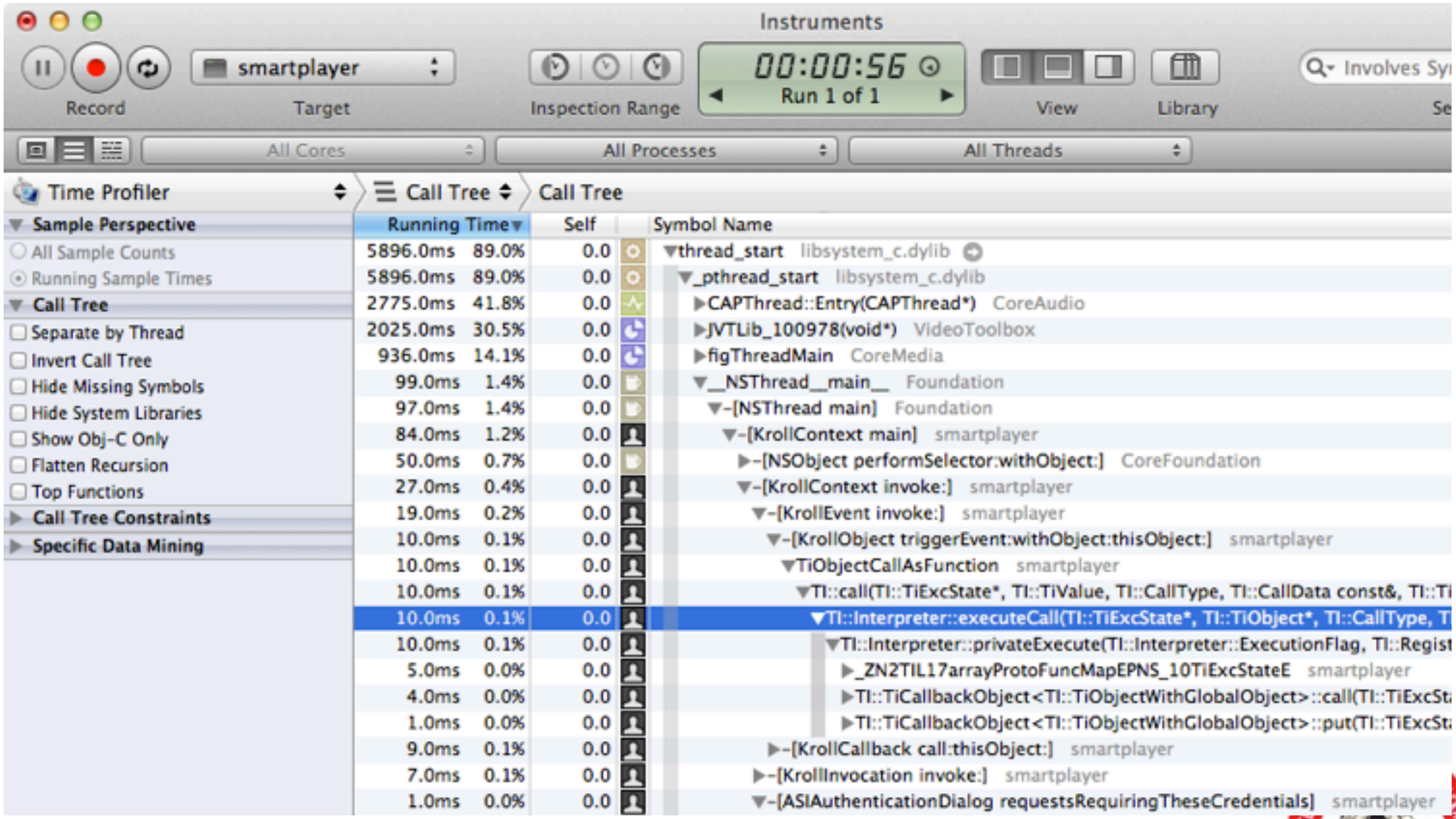
Name	Incl Cpu Time %	Incl Cpu Time	Excl Cpu Time %	Excl Cpu Time	Calls+Recur Calls/Total	Cpu Time/Call
687 org/bouncycastle/asn1/ASN1Encodable.<init> ()V	0.1%	5.623	0.1%	5.623	137+0	0.041
688 java/util/ArrayList$ArrayListIterator.hasNext ()Z	0.1%	5.611	0.1%	5.611	179+0	0.031
689 org/bouncycastle/asn1/ASN1InputStream$1.<init> (Lorg/bouncycastle/asn1/ASN1I	0.1%	5.611	0.0%	2.282	40+0	0.140
690 java/io/File.getAbsolutePath ()Ljava/lang/String;	0.1%	5.603	0.0%	0.399	5+0	1.121
691 java/util/regex/Pattern.compileImpl (Ljava/lang/String;I)V	0.1%	5.603	0.0%	0.386	3+0	1.868
692 android/content/pm/ServiceInfo$1.createFromParcel (Landroid/os/Parcel;)Ljava/lan	0.1%	5.566	0.0%	0.055	1+0	5.566
693 org/bouncycastle/asn1/x509/AlgorithmIdentifier.toASN1Object ()Lorg/bouncycastle	0.1%	5.554	0.0%	0.539	3+0	1.851
694 android/content/pm/ServiceInfo$1.createFromParcel (Landroid/os/Parcel;)Landroid/	0.1%	5.511	0.0%	0.285	1+0	5.511
695 org/apache/harmony/security/asn1/ASN1OctetString.getDecodedObject (Lorg/apac	0.1%	5.490	0.1%	3.496	46+0	0.119
696 org/apache/http/impl/client/AbstractHttpClient.getAuthSchemes ()Lorg/apache/htt	0.1%	5.418	0.0%	0.068	1+0	5.418
697 org/apache/harmony/security/x509/DNParser.nextAT ()Ljava/lang/String;	0.1%	5.414	0.0%	2.761	24+0	0.226
698 com/ibm/icu4jni/charset/NativeConverter.flushCharToByte ()[B[I)I	0.1%	5.374	0.1%	5.374	45+0	0.119
699 org/apache/http/impl/client/DefaultHttpClient.createAuthSchemeRegistry ()Lorg/ap	0.1%	5.350	0.0%	0.625	1+0	5.350
700 android/database/sqlite/SQLiteProgram.releaseCompiledSqlIfNotInCache ()V	0.1%	5.345	0.0%	0.848	6+0	0.891
701 java/lang/Class.newInstanceImpl ()Ljava/lang/Object;	0.1%	5.329	0.0%	1.360	8+0	0.666
702 org/bouncycastle/asn1/DERInteger.getPositiveValue ()Ljava/math/BigInteger;	0.1%	5.313	0.0%	0.587	8+0	0.664
703 java/util/TimSort.sort ([Ljava/lang/Object;Ljava/util/Comparator;)V	0.1%	5.292	0.0%	2.227	26+0	0.204
704 java/nio/ByteBuffer.wrap ([BII)Ljava/nio/ByteBuffer;	0.1%	5.253	0.0%	0.739	9+0	0.584
705 java/util/concurrent/ThreadPoolExecutor.processWorkerExit (Ljava/util/concurrent/	0.1%	5.245	0.0%	0.226	1+0	5.245
706 android/database/sqlite/DatabaseObjectNotClosedException.<init> ()V	0.1%	5.233	0.0%	0.692	12+0	0.436
707 android/content/pm/ServiceInfo.<init> (Landroid/os/Parcel;Landroid/content/pm/S	0.1%	5.226	0.0%	0.062	1+0	5.226

Android DDMS + Traceview

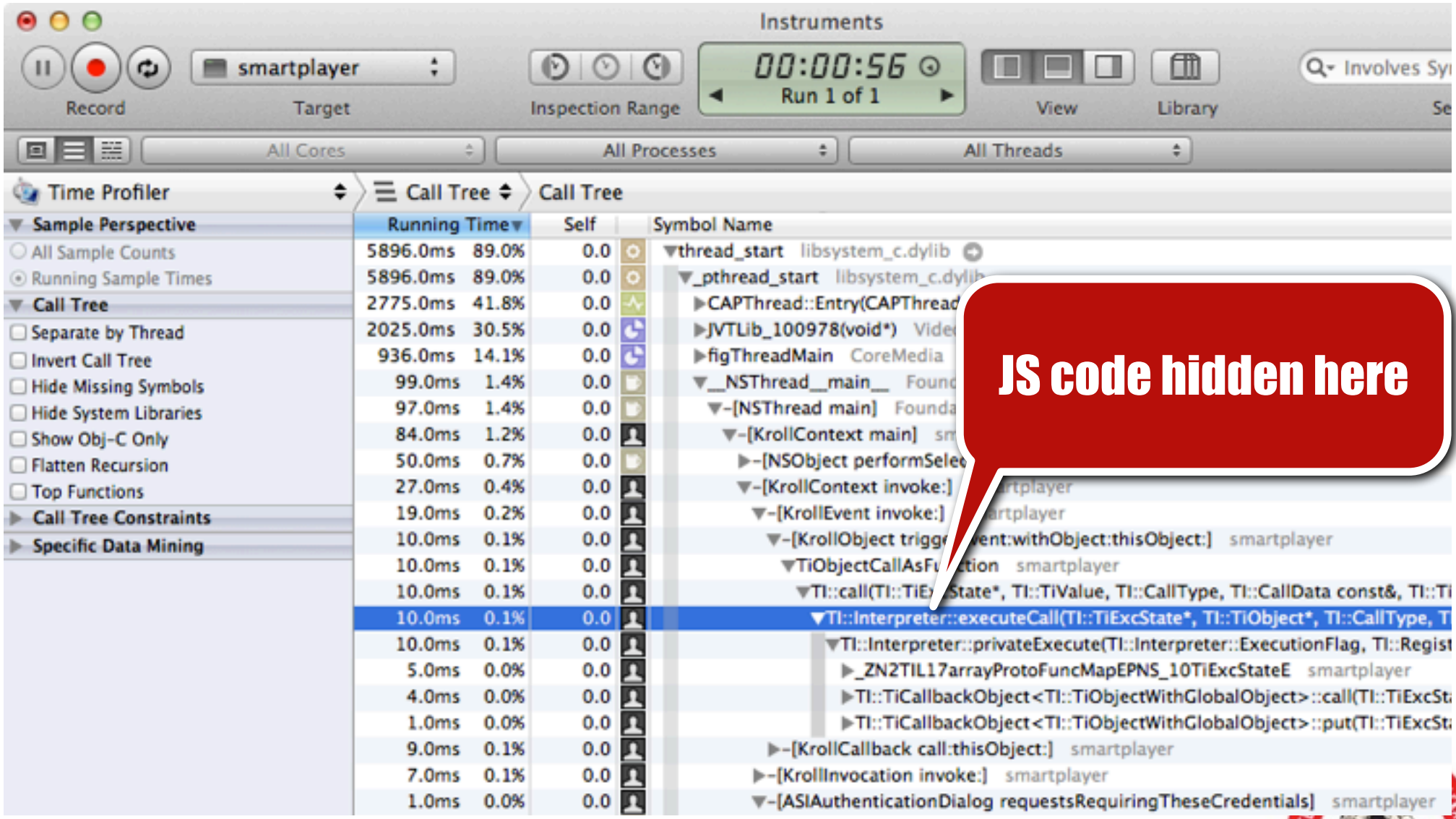


WHERE'S MY JS???

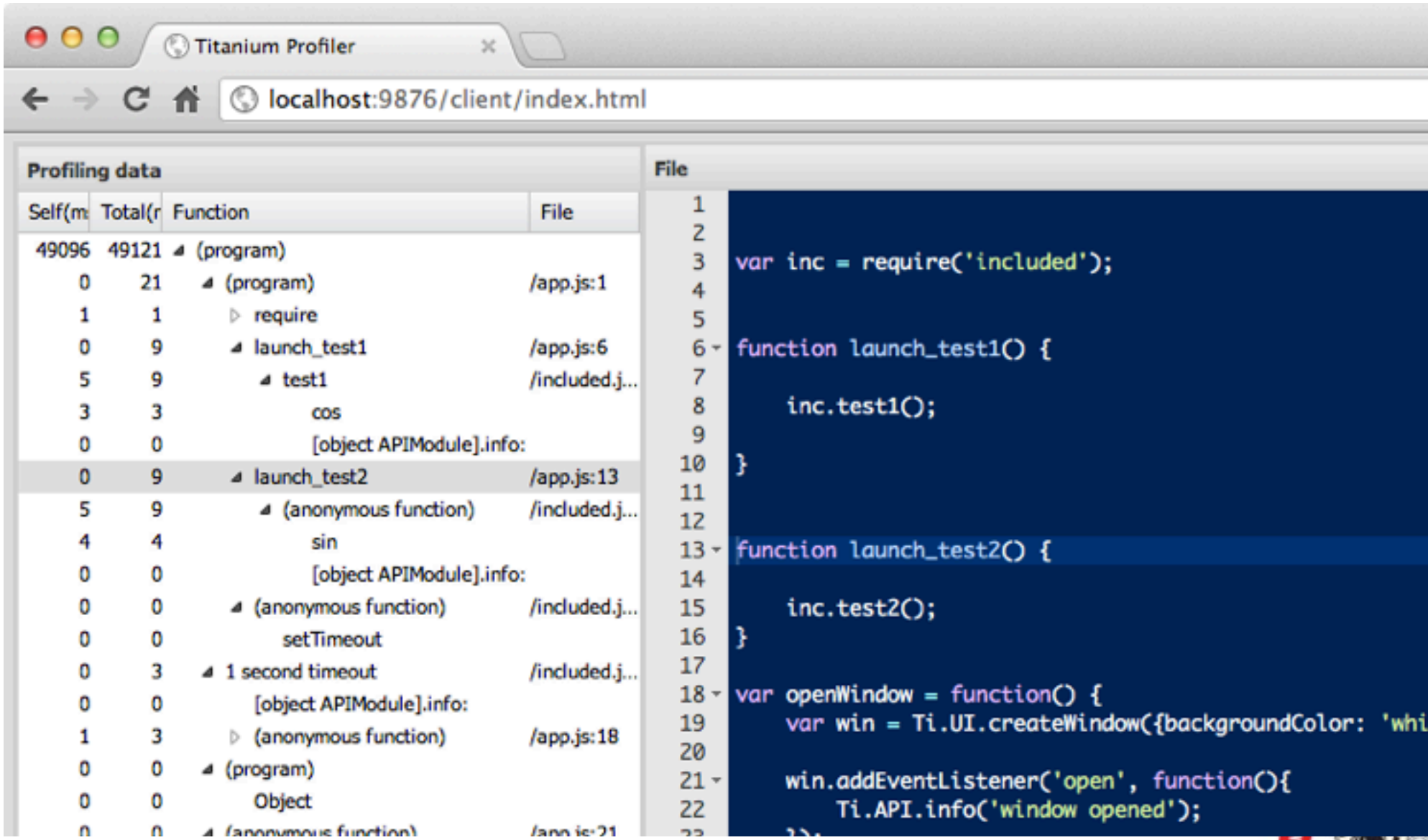
Xcode Instruments: profiler tool



Xcode Instruments: profiler tool



Titanium Profiler... still to come



Titanium Profiler (iOS only)

- Custom version of the Ti JavaScriptCore library
- Work in progress
- Some info here

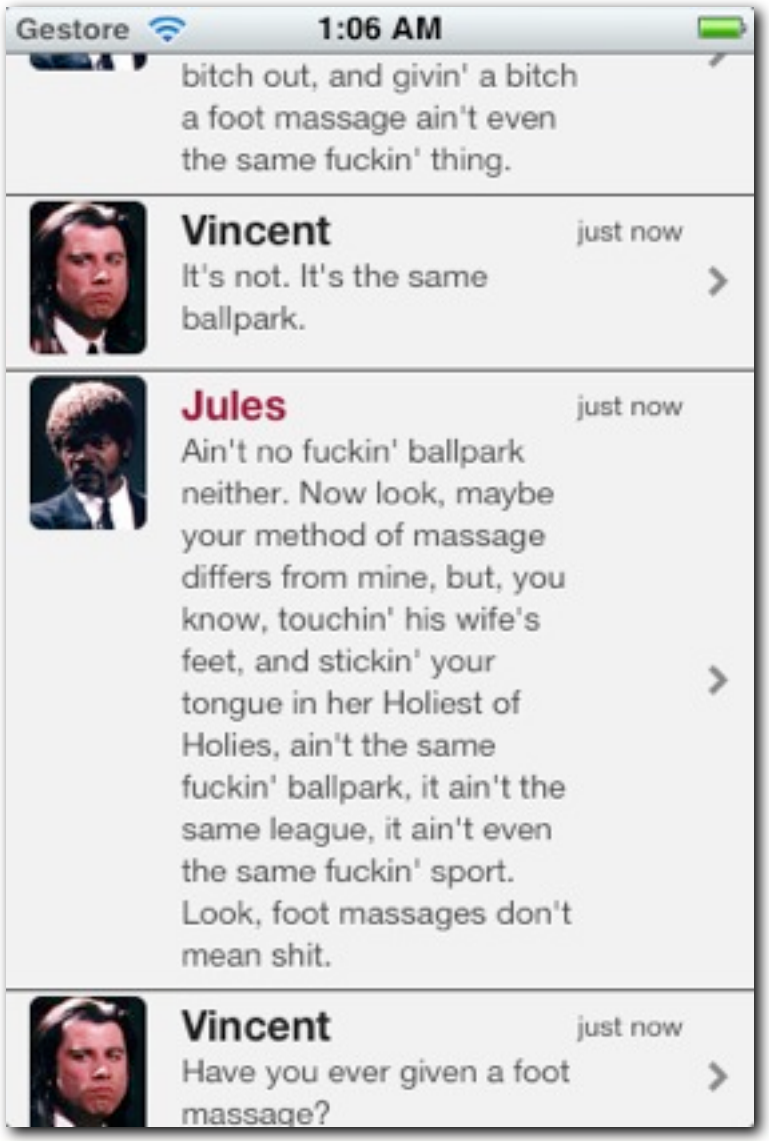
  http://titaniumninja.com/profiling-ti-mobile-apps-is-it-possible/
- Will be possibly integrated in Ti Mobile
- Stay tuned on

  http://titaniumninja.com/

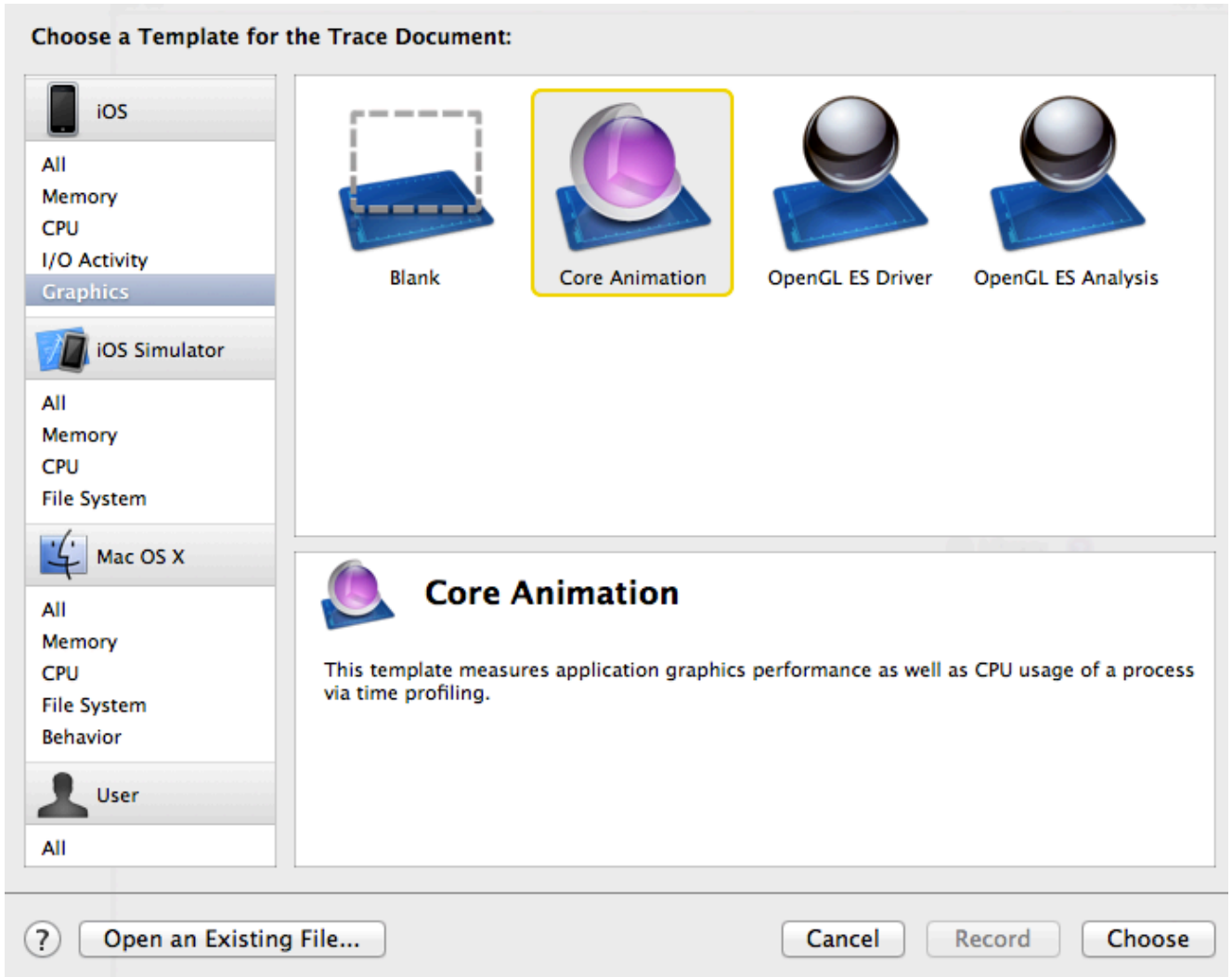
Is table creation REALLY slow?

- Try optimizing the row creation loop body
- Implement Lazy loading
  - Initialize the table with fewer rows and show it immediately
  - Meantime continue creating the remaining rows and update the table with the complete row set

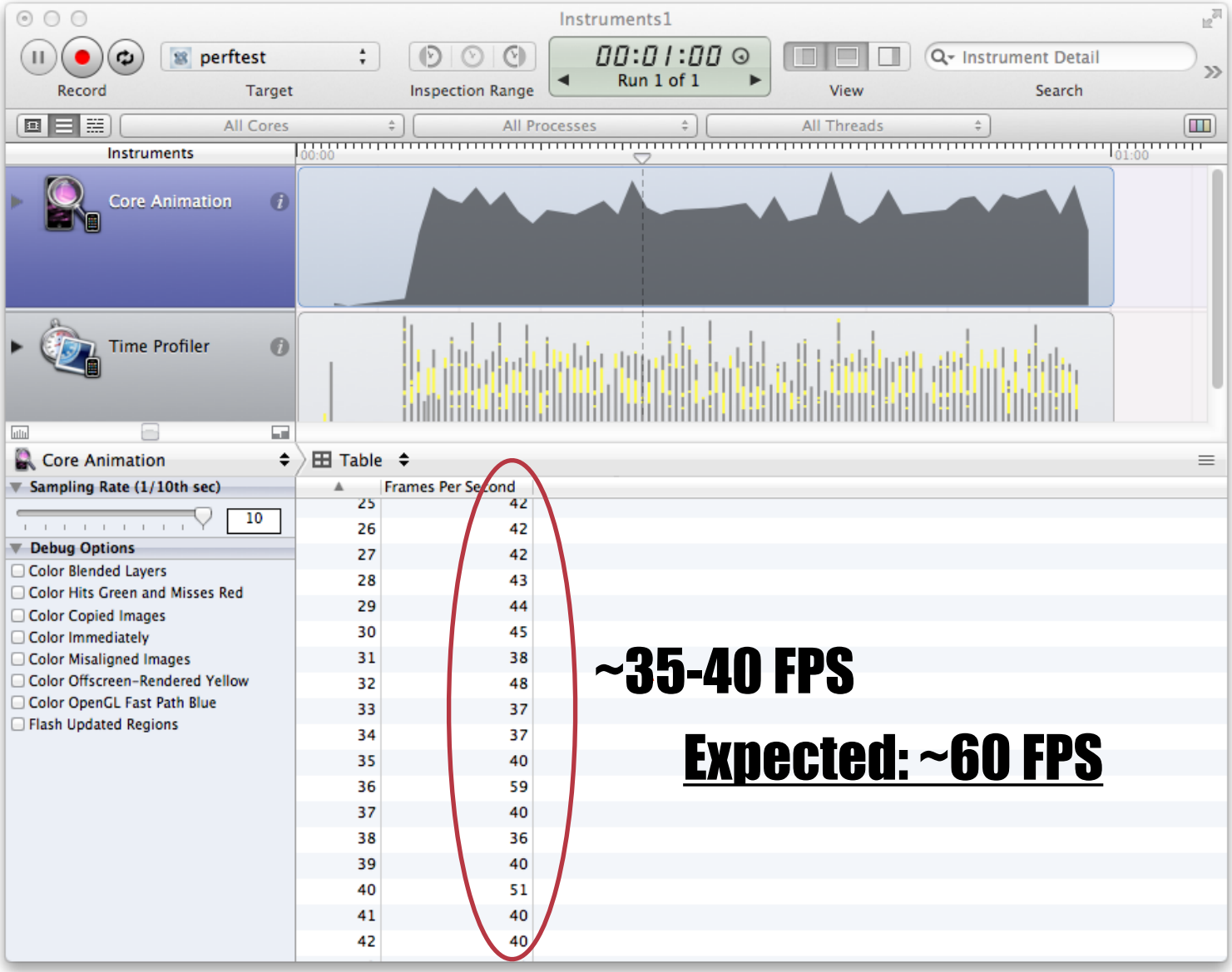
Wanna smooth scrolling?



Instruments: Core Animation



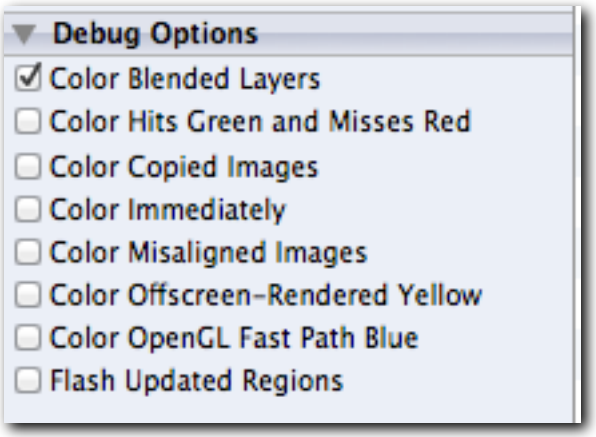
Tables with complex rows are slow

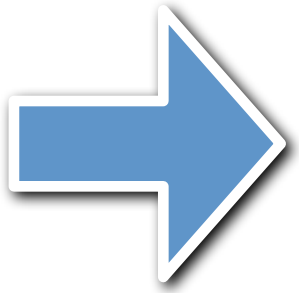


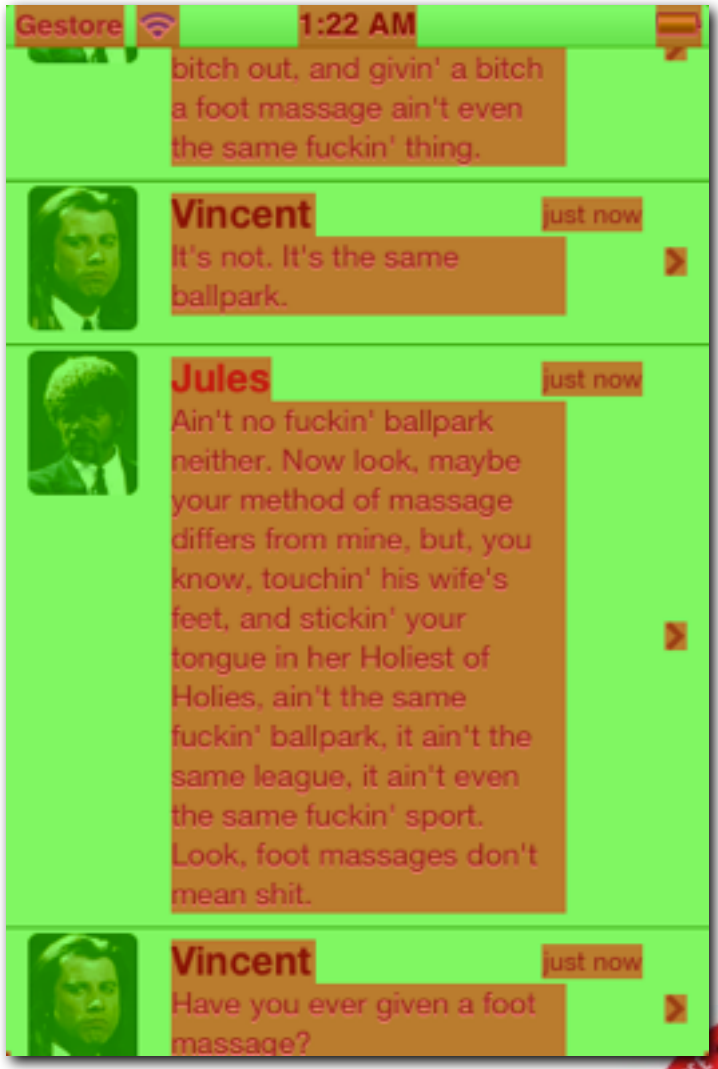
~35-40 FPS

Expected: ~60 FPS

1. Blended layers







These labels are transparent, even if we set the background color, or a background image

What can we do?

```
-(UILabel*)label
{
    if (label==nil)
    {
        label = [[UILabel alloc] initWithFrame:CGRectZero];
        label.backgroundColor = [UIColor whiteColor];
        label.numberOfLines = 0;
        [self addSubview:label];
        label.opaque = YES;
        self.opaque = YES;
        self.backgroundColor = [UIColor whiteColor];
    }
    return label;
}
```

TiUILabel.m:104

Discussion (Cfr. UIView Reference)

"The **opaque** property provides a hint to the drawing system as to how it should treat the view. If set to `YES`, the drawing system treats the view as fully opaque, which allows the drawing system to optimize some drawing operations and improve performance"

What can we do?

```
-(UILabel*)label
{
    if (label==nil)
    {
        label = [[UILabel alloc] initWithFrame:CGRectZero];
        label.backgroundColor = [UIColor whiteColor];
        label.numberOfLines = 0;
        [self addSubview:label];
        label.opaque = YES;
        self.opaque = YES;
        self.backgroundColor = [UIColor whiteColor];
    }
    return label;
}
```

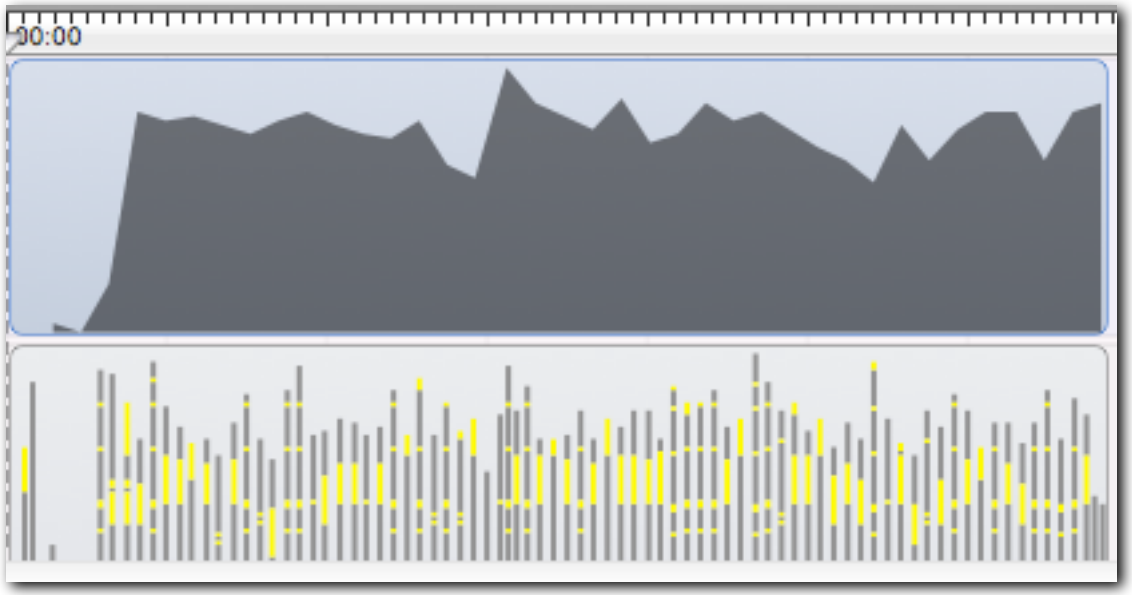
NOT A FIX!!!

TiUILabel.m:104

Discussion (Cfr. UIView Reference)

"The **opaque** property provides a hint to the drawing system as to how it should treat the view. If set to `YES`, the drawing system treats the view as fully opaque, which allows the drawing system to optimize some drawing operations and improve performance"

Result

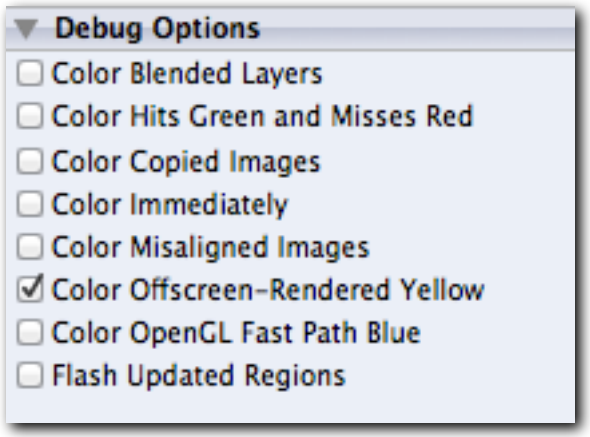


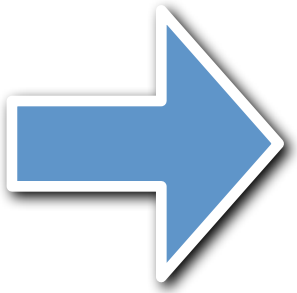
Table

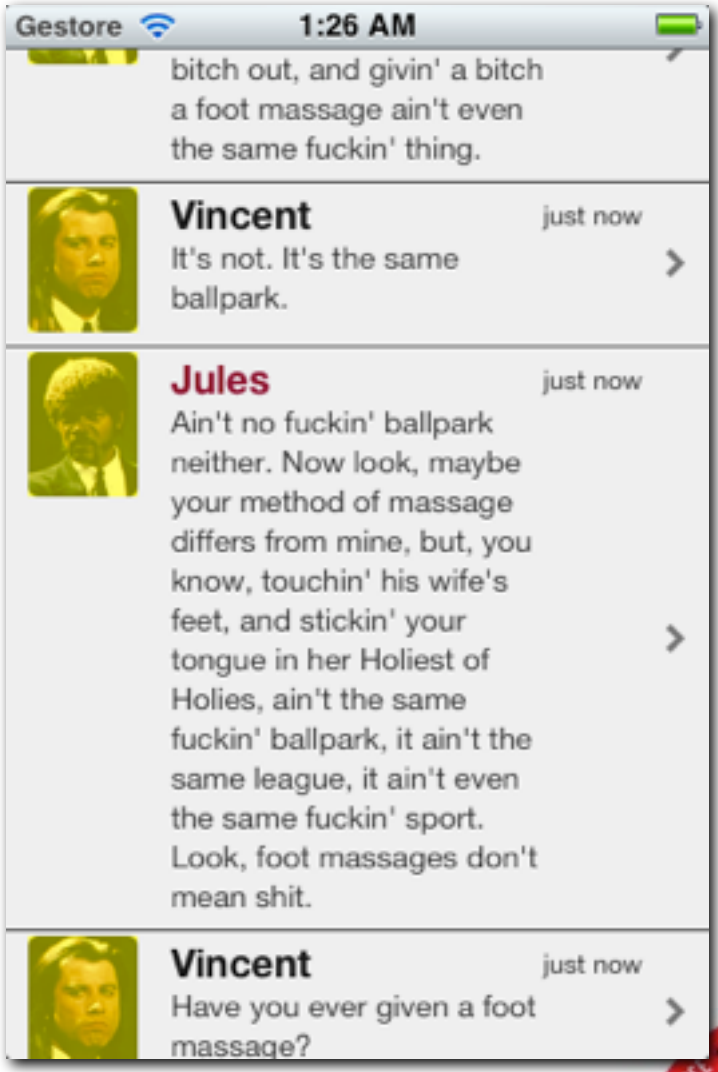
	Frames Per Second
4	50
5	48
6	49
7	49
8	45
9	48
10	50
11	47
12	45
13	44
14	48
15	38
16	35
17	60
18	52
19	49
20	46
21	53

~45-50 FPS

2. Offscreen rendered layers

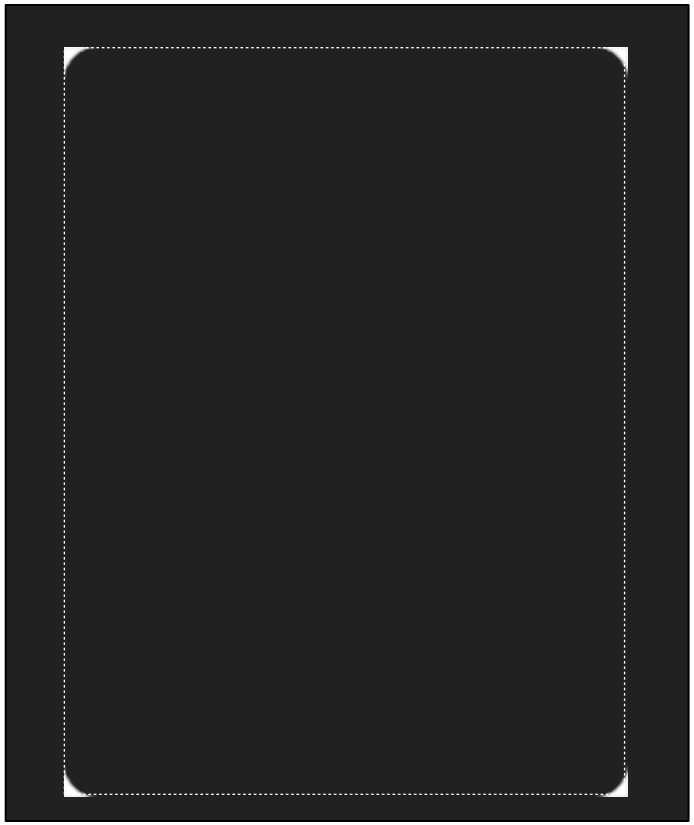






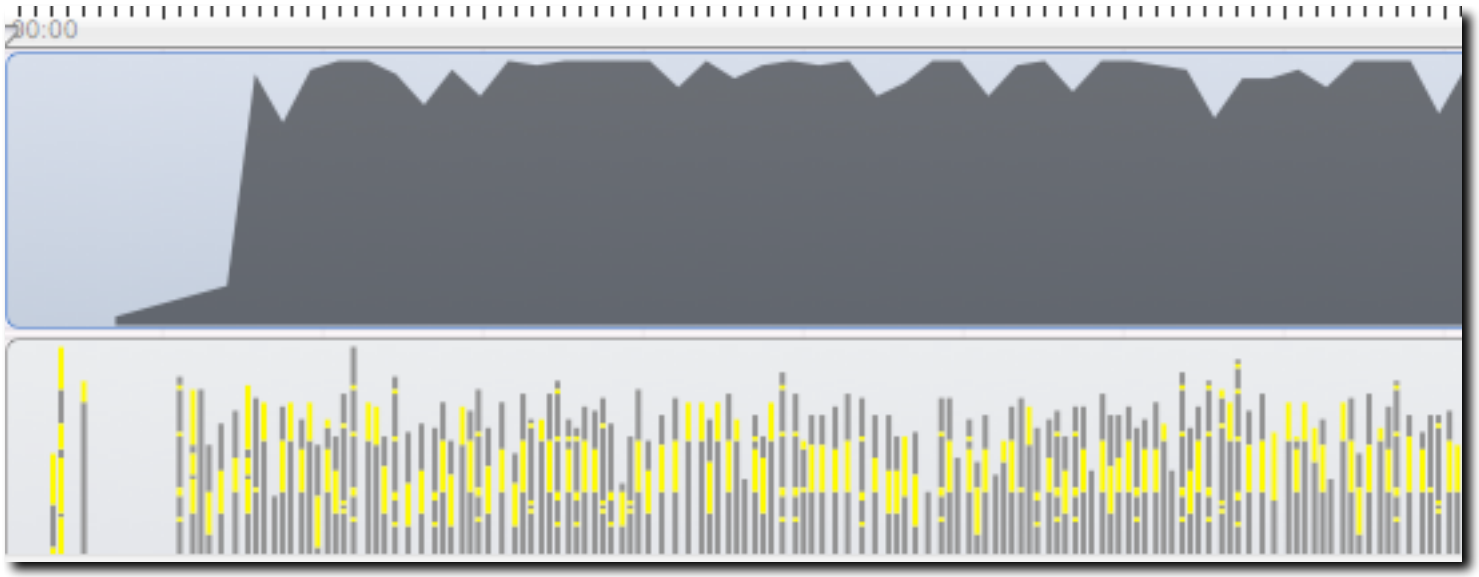
Solution

- Don't use rounded corners (`borderRadius`)
- Use an image mask instead



```
var img = Ti.UI.createMaskedImage({
    mask : 'mask.png',// alpha mask
    image : myPicture,//image to mask
    mode : Ti.UI.iOS.BLEND_MODE_SCREEN
});
```

What about FPSs?



Table

	Frames Per Second
10	60
11	57
12	50
13	58
14	52
15	60
16	59
17	60
18	60
19	60
20	60
21	54
22	60
23	56
24	59
25	60
26	59

~55-60 FPS

(Just using image masks)

Not satisfied? Going fully native

- Let's create a custom optimized UITableView
- We'll expose it through a native module

API for the custom view

- API
  - `createMessagesView(properties);`
  - `setMessages([]messages);`
  - `insert(message);`
  - `addEventListener('click', callback);`
- Config

```
{
    rowBackgroundColor: '#efefef',
    nameColorIfMe: '#191919',
    nameColorIfOther: '#8f032c',
    …
}
```

API for the custom view

- API
  - `createMessagesView(properties);`
  - `setMessages([]messages);`
  - `insert(message);`
  - `addEventListener('click', callback);`
- Config

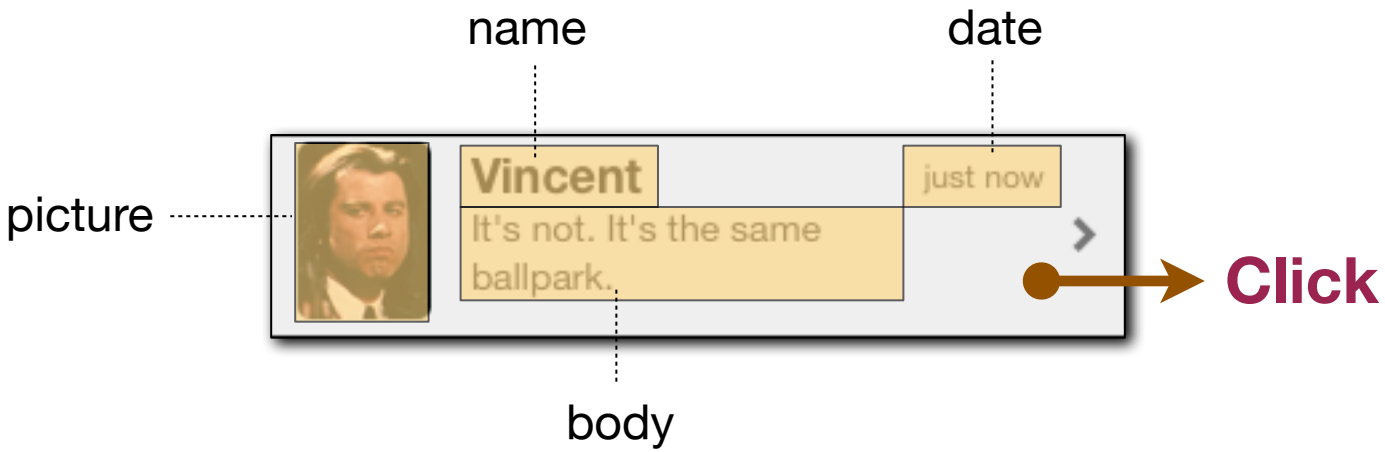
```
{
    rowBackgroundColor: '#efefef',
    nameColorIfMe: '#191919',
    nameColorIfOther: '#8f032c',
    …
}
```

DECREASED FLEXIBILITY

Message model

```
{
    isMe: true,
    name: "Vincent",
    picture: "vin.png",
    body: "It's not. It's the same ballpark.",
    date: 12394838299 //unix timestamp
},
```

A single row


name
date
picture
Vincent
just now
It's not. It's the same ballpark.
Click
body

Traditional UITableViewCell

```
- (UITableViewCell *)tableView:(UITableView *)tableView cellForRowAtIndexPath:(NSIndexPath
*)indexPath
{
    UITableViewCell *cell = [tableView dequeueReusableCellWithIdentifier:@"MessageCell"];

    if (cell == nil) {
        cell = [[[UITableViewCell alloc] initWithStyle:UITableViewCellStyleDefault
               reuseIdentifier:@"MessageCell"] autorelease];

        UILabel *nameLabel = [[UILabel alloc] initWithFrame:CGRectMake(75.0, 25.0, 120.0, 18.0)];
        nameLabel.font = [UIFont boldSystemFontOfSize:16.0];
        nameLabel.textAlignment = UITextAlignmentLeft;
          nameLabel.backgroundColor = rowBackgroundColor;

        [cell.contentView addSubview:nameLabel];

    } else {
        //...
    }

    Message *msg = [messages objectAtIndex:(indexPath.row)];
    BOOL isMe = [TiUtils boolValue:[dict objectForKey:@"isMe"]];
    nameLabel.textColor = isMe ? nameColorIfMe : nameColorIfOther;
    //...
```

Not fast enough? Use FastCells

Low level drawing primitives

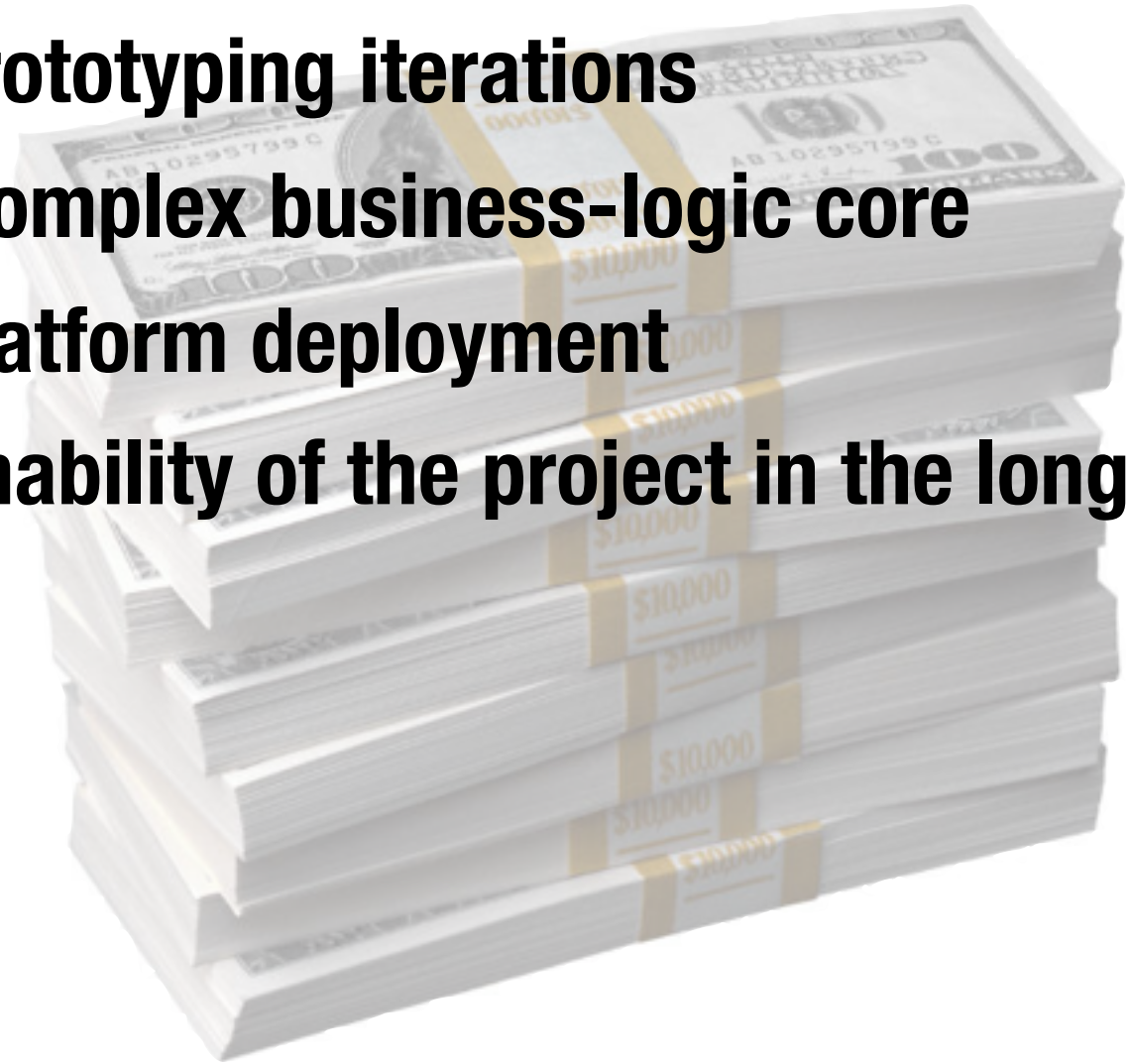
```
- (void) drawContentView:(CGRect)rect {
    if (message == nil)
        return;

    CGContextRef context = UIGraphicsGetCurrentContext();

    CGContextClearRect(context, rect);

    CGFloat max_width = self.contentView.frame.size.width;


    if (message.isMe)
        [nameColorIfMe set];
    else
        [nameColorIfOther set];

    [message.name drawInRect:CGRectMake(userInfoLeft, userInfoTop, rectWidth(userInfoLeft,
     userInfoRight, max_width), userInfoHeight) withFont:bigBoldFont
     lineBreakMode:UILineBreakModeTailTruncation];

     //...
}
```

Concluding remarks on Titanium

- Is it worth it?
- Does it really speeds up development?
- Is developing with Titanium “better” or “worse” than making parallel per-platform developments?

Where's the value?

- Quick prototyping iterations
- Single complex business-logic core
- Cross platform deployment
- Maintainability of the project in the long term



Some interesting reads

- JavaScriptCore, the WebKit JS implementation - http://wingolog.org/archives/2011/10/28/javascriptcore-the-webkit-js-implementation
- v8: a tale of two compilers - http://wingolog.org/archives/2011/07/05/v8-a-tale-of-two-compilers
- JSC v/s V8 performance on ARM - http://xc0ffee.wordpress.com/2011/09/07/webkit-gtk-jsc-vs-v8-performance-on-arm/
- The Future of JavaScript Engines: Replace Them With JavaScript Compilers - http://shorestreet.com/node/43
- Optimisation: don't waste your time - http://www.scirra.com/blog/83/optimisation-dont-waste-your-time
- Guidelines for JavaScript Programs: Are They Still Necessary? - http://www.inf.u-szeged.hu/~akiss/pub/pdf/herczeg_guidelines.pdf

Thank YOU!

Any question?