tips & tricks

for iPhone application development

adrian kosmaczewski

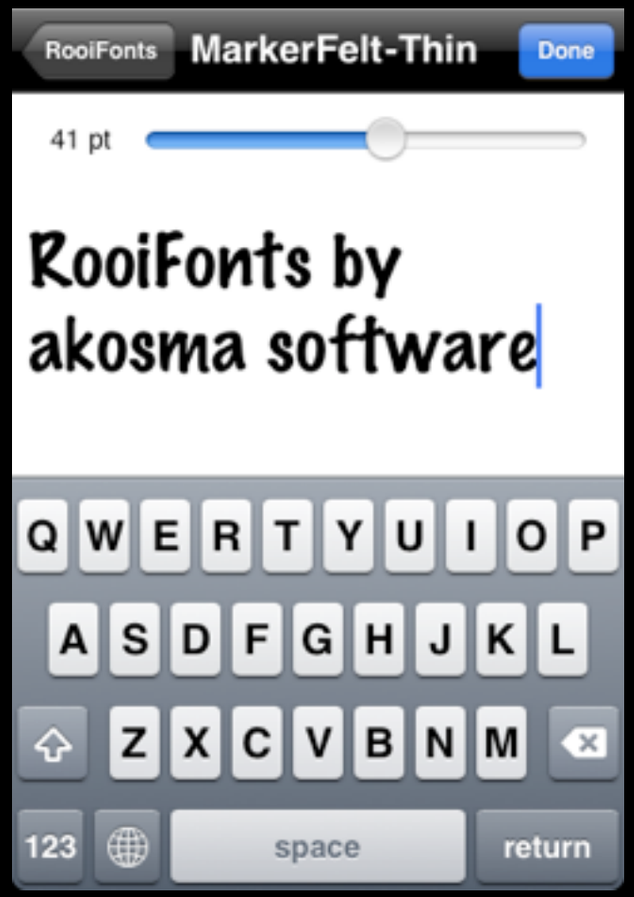
akosma software

akosma.com
akosma.tumblr.com
github.com/akosma
linkedin.com/in/akosma
facebook.com/akosma

@akosma

Orange
17:35
Current location:
Latitude: 46.510
Longitude: 6.659
En este lugar encont...
Lista de compras:
Leche
Huevos
Orégano
Pasta dentífrica
testing my latest iPhone app and its integration with Twitterrific, which doesn't look bad!
In a couple of year's time, with more RAM and CPU power, the iPhone OS will feature garbage...
Actually I got so used to retain, release and the Clang static analyzer that I

bluewoki by akosma
connected with Hermes
connect


RooiFonts
MarkerFelt-Thin
Done
41 pt
RooiFonts by akosma software
Q W E R T Y U I O P
A S D F G H J K L
Z X C V B N M
123
space
return

DeviceDNA
2C1641DC-FFA5-5055-A...
Unique Device Identifier
iPhone Simulator
Device Name
iPhone OS 3.0
iPhone OS Version
iPhone Simulator
Device Model
Select the values to send and press the message button
DeviceDNA © Copyright 2009 akosma software - All Rights Reserved

Police
Fire
Ambulance
AIR-GLACIERS
rega
Tel 143
Tox
Swiss Emergency Numbers

some questions

@perlausten?
@supermouse?
@williwu?

new to iPhone?

technologies?

J2EE, J2ME, .NET, RoR, others?

programming languages?

C / C++?
Java, C#?
Ruby, Python, Lua?
JavaScript?

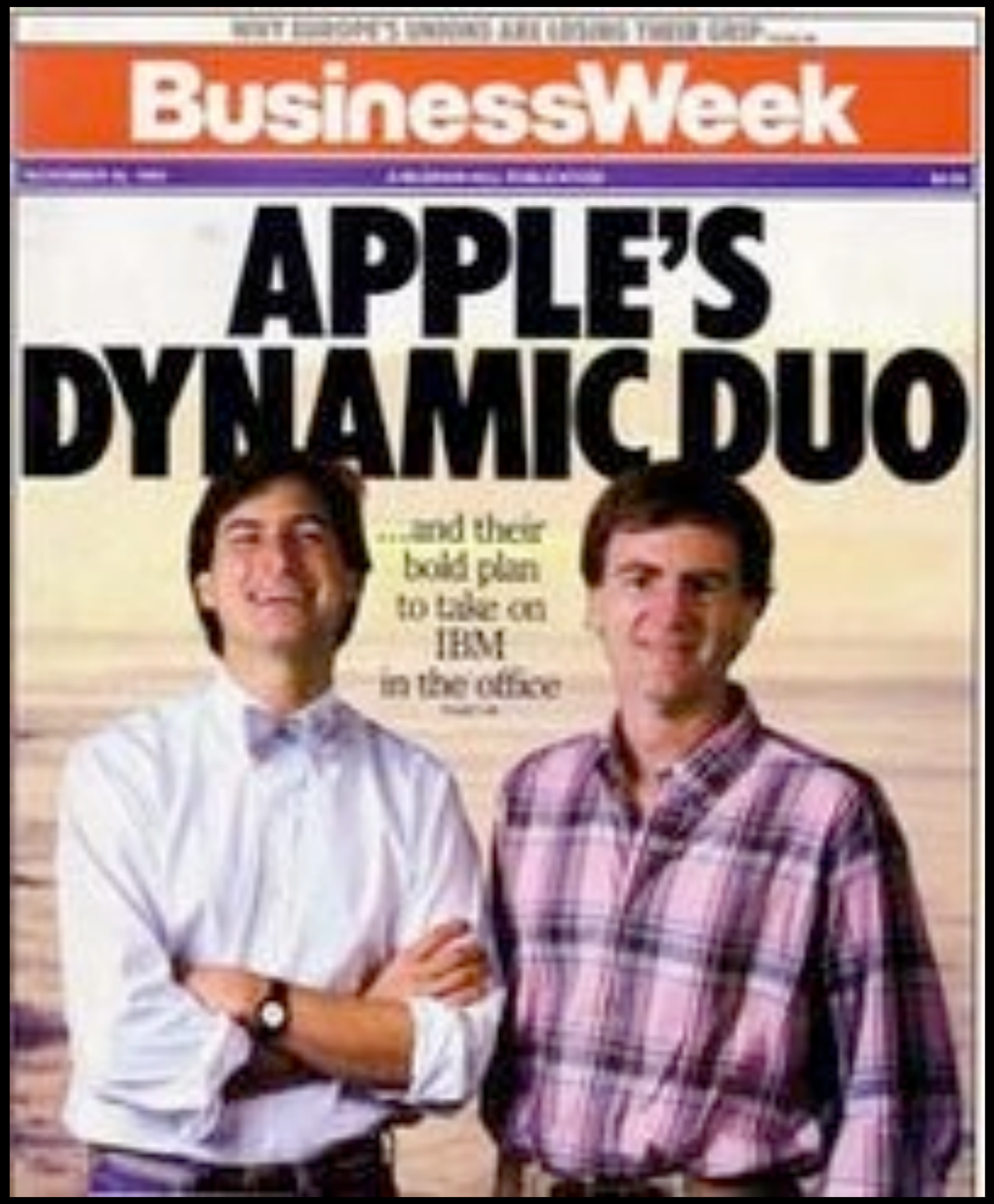
iPhone dev

≠

web dev

Cocoa &
Objective-C

born in the 80s


BusinessWeek
APPLE'S DYNAMIC DUO
...and their bold plan to take on IBM in the office

NeXT
NeXT

WorldWideWeb
Info
Navigate
Document
Find
Edit
Links
Style
Print... p
Page layout...
Windows
Services
Hide h
Quit q

Links
Mark all A
Mark selection M
Link to marked L
Link to New N
Unlink Z
Link to file...
Help

Tim's Home Page

My home page

altas

The World-Wide Web Virtual Library: Subject Catalogue

The WWW Virtual Library

This is a c
type ., an

Mail to ma
to add po
administra

See also

Aeronaut

Agricultu

Anthropology

Archaeology

Asian Studies

Astronomy and

Bio Sciences Separate list .

High-Energy Physics Information

CERN Welcome

CERN

European Laboratory for Part

Geneva, Switzerlan

About the Laboratory:

- on CERN info ,
- General information, divisions, groups and activities , scientific committees

Separate list.

CERN Experiments

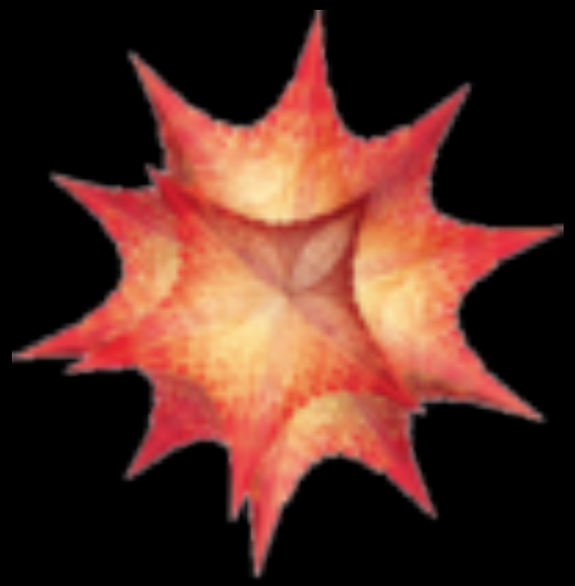
Experiments

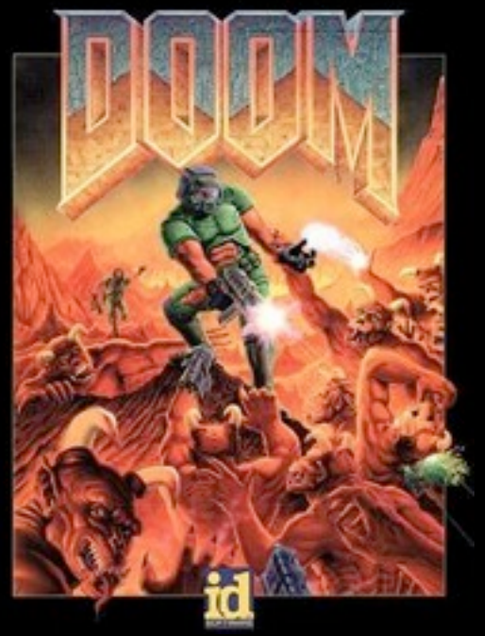
WWW Support for Experiments

ALEPH	LEP experiment
ALICE	A Large Ion Collider E LHC
ATLAS	A Toroidal LHC Appar
CHORUS	WA95 - Neutrino oscill CERN
CMS	Compact Muon Solenc

Atlas

Lotus




DOOM
id

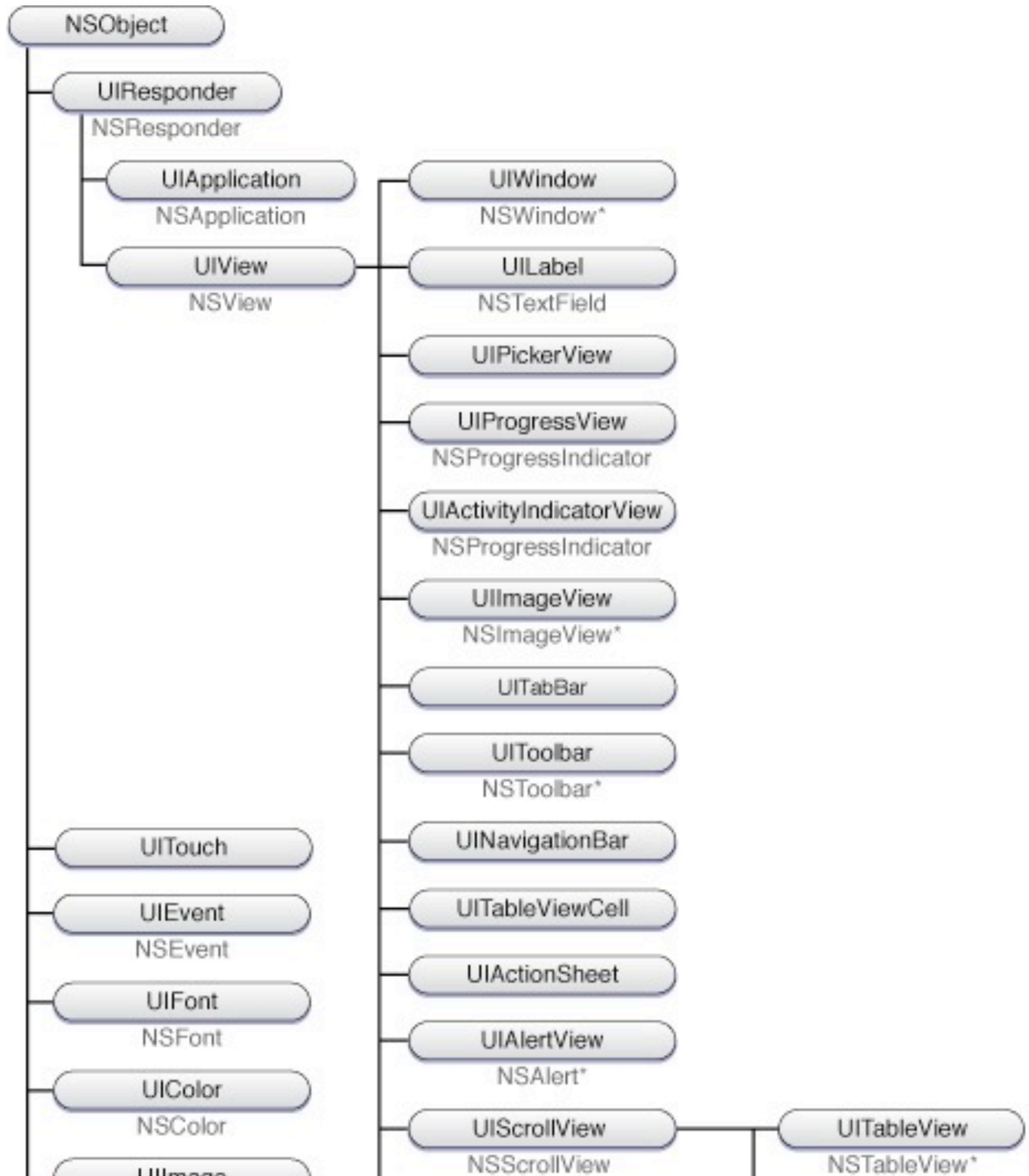
Objective-C

Objective-C	Java
@interface & @implementation	class
@protocol	interface
#import // files!	import // classes!
categories	n/a
id	n/a

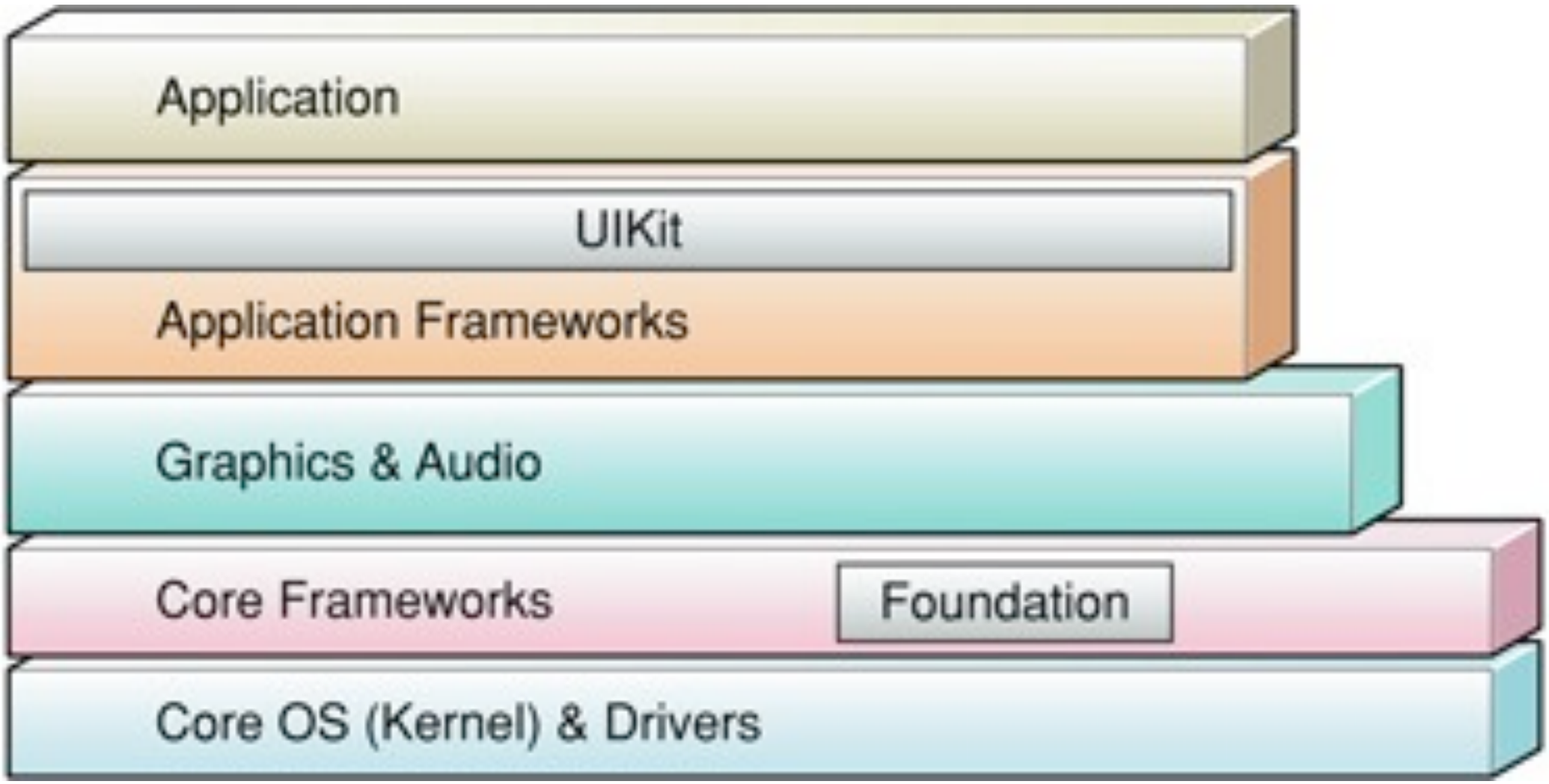
- Thin layer around C
- Message-dispatch runtime
- Static and dynamic (you choose)
- The “real” father of Java:
  - http://cs.gmu.edu/~sean/stuff/java-objc.html

- Single inheritance + interfaces (“@protocols”)
- @protocols can have optional methods
- Fields protected by default
- All methods are virtual and overridable
- Methods can be added to existing classes
- Full introspection / reflection
- Messages can be intercepted and forwarded
  - à la AOP!

Cocoa Touch


NSObject
UIResponder
NSResponder
UIApplication
NSApplication
UIView
NSView
UIWindow
NSWindow*
UILabel
NSTextField
UIPickerView
UIProgressView
NSProgressIndicator
UIActivityIndicatorView
NSProgressIndicator
UIImageView
NSImageView*
UITabBar
UIToolbar
NSToolbar*
UINavigationBar
UITableViewCell
UIActionSheet
UIAlertView
NSAlert*
UIScrollView
NSScrollView
UITableView
NSTableView*
UITouch
UIEvent
NSEvent
UIFont
NSFont
UIColor
NSColor

iPhone Specifics


Application
UIKit
Application Frameworks
Graphics & Audio
Core Frameworks
Foundation
Core OS (Kernel) & Drivers

128 or 256 MB RAM

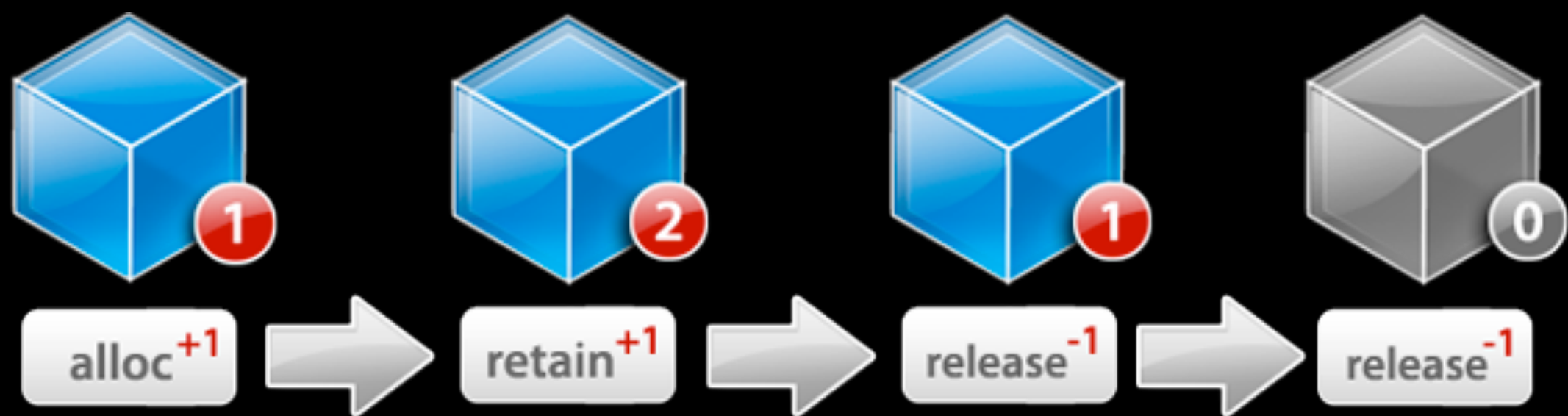
~70 MB for the OS!

no swap file

(no virtual memory)

no garbage collector

(manual memory management)



http://cocoadevcentral.com/d/learn_objectivec/

no automatic objects
on the stack (C++)

```
// C++
// Memory freed when out of scope
std::string name(“Adrian”);

std::string *name = NULL;
name = new std::string(“Adrian”);
delete name;
```

hygiene

memory warnings

```
- (void)didReceiveMemoryWarning
{
    [super didReceiveMemoryWarning];
}
```

```
- (void)applicationDidReceiveMemoryWarning:(UIApplication *)application
{
    [[ImageCache sharedImageCache] removeAllImagesInMemory];
}
```

```
NSNotificationCenter *center = [NSNotificationCenter defaultCenter];
[center addObserver:self
          selector:@selector(whatever:)
              name:UIApplicationDidReceiveMemoryWarningNotification
            object:nil];
```

avoid autorelease

```
// Instead of
NSString *string = [NSString stringWithFormat:@"value = %d", intVariable];

// use
NSString *string = [[NSString alloc] initWithFormat:@"value = %d"
, intVariable];
...
[string release];
```

```
NSAutoreleasePool *pool = [[NSAutoreleasePool alloc] init];
for (id item in array)
{
    id anotherItem = [item createSomeAutoreleasedObject];
    [anotherItem doSomethingWithIt];
}
[pool release];
```

lazy-loading & reuse

```
- (void)tableView:(UITableView *)tableView
        didSelectRowAtIndexPath:(NSIndexPath *)indexPath
{
    Item *item = [items objectAtIndex:indexPath.row];
    if (detailController == nil)
    {
        detailController = [[DetailController alloc] init];
    }
    detailController.item = item;
    [self.navigationController
        pushViewController:detailController
                   animated:YES];
}
```

custom table cells & reuse

```
- (UITableViewCell *)tableView:(UITableView *)tableView
        cellForRowAtIndexPath:(NSIndexPath *)indexPath
{
    Item *item = [items objectAtIndex:indexPath.row];
    static NSString *identifier = @"ItemCell";

    ItemCell *cell = (ItemCell *)[tableView
          dequeueReusableCellWithIdentifier:identifier];

    if (cell == nil)
    {
        cell = [[[ItemCell alloc] initWithIdentifier:identifier]
                                                    autorelease];
    }
    cell.item = item;
    return cell;
}
```

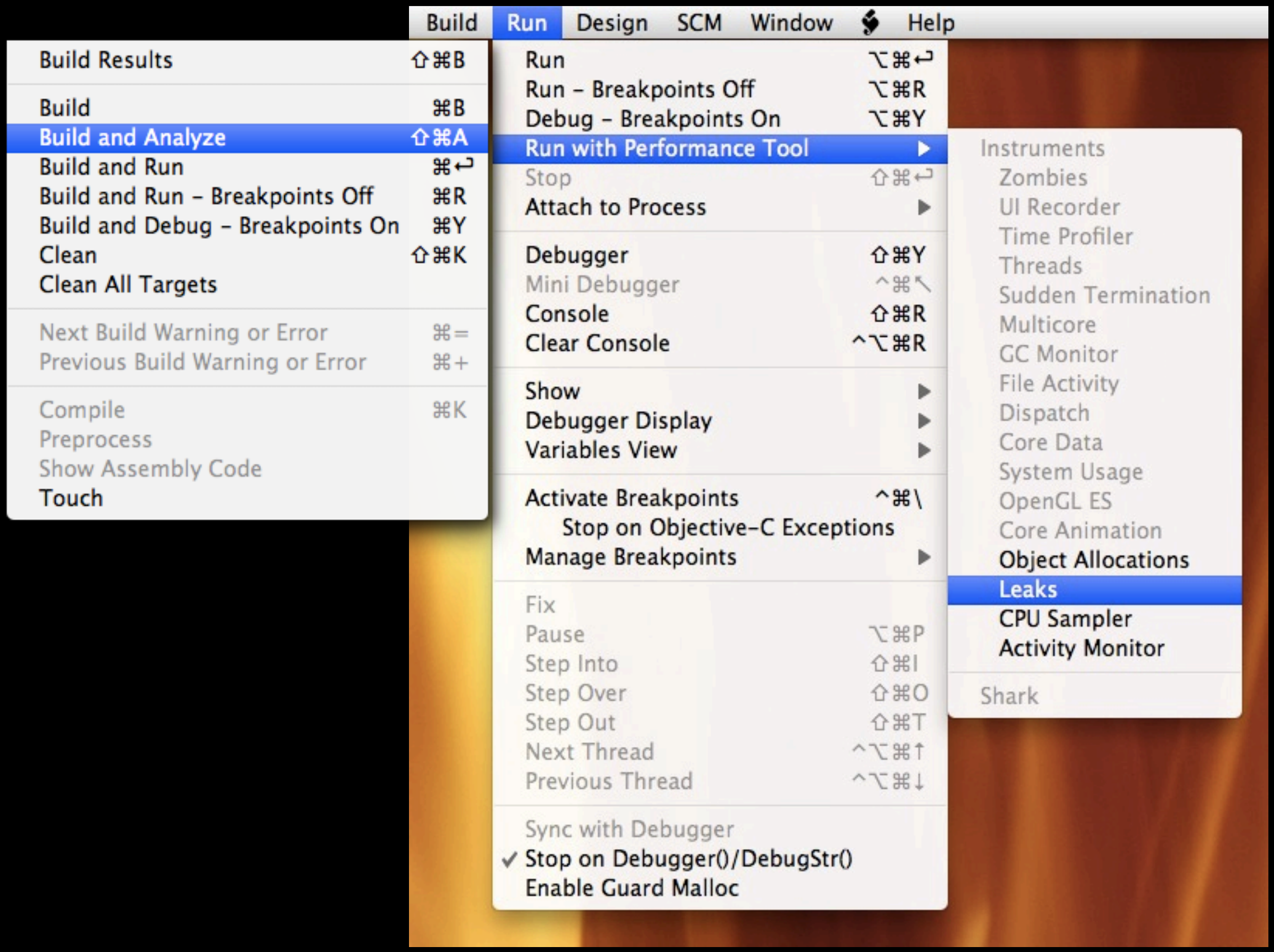
override setters properly

```
- (void)setItems:(NSArray *)obj
{
    if (obj == items)
    {
        return;
    }
    [items release];
    items = nil;
    items = [obj retain];

    if (items != nil)
    {
        // create the internal structure of the cell
        // if not present, and change the widget values
    }
}

- (void)setName:(NSString *)obj
{
    [name release];
    name = nil;
    name = [obj copy];   // I always copy NSStrings!

    if (name != nil)
    {
        // create the internal structure of the cell
        // if not present, and change the widget values
    }
}
```

beware of delegation

```
- (void)dealloc
{
    // widget might be retained by someone else!
    widget.delegate = nil;
    [widget release];
    [super dealloc];
}
```

instruments & static analysis


Build
Run
Design
SCM
Window
Help
Build Results ⇧⌘B
Build ⌘B
Build and Analyze ⇧⌘A
Build and Run ⌘↩
Build and Run – Breakpoints Off ⌘R
Build and Debug – Breakpoints On ⌘Y
Clean ⇧⌘K
Clean All Targets
Next Build Warning or Error ⌘=
Previous Build Warning or Error ⌘+
Compile ⌘K
Preprocess
Show Assembly Code
Touch
Run ⌥⌘↩
Run – Breakpoints Off ⌥⌘R
Debug – Breakpoints On ⌥⌘Y
Run with Performance Tool
Stop ⇧⌘↩
Attach to Process
Debugger ⇧⌘Y
Mini Debugger ^⌘↖
Console ⇧⌘R
Clear Console ^⌥⌘R
Show
Debugger Display
Variables View
Activate Breakpoints ^⌘\
Stop on Objective-C Exceptions
Manage Breakpoints
Fix
Pause ⌥⌘P
Step Into ⇧⌘I
Step Over ⇧⌘O
Step Out ⇧⌘T
Next Thread ^⌥⌘↑
Previous Thread ^⌥⌘↓
Sync with Debugger
✓ Stop on Debugger()/DebugStr()
Enable Guard Malloc
Instruments
Zombies
UI Recorder
Time Profiler
Threads
Sudden Termination
Multicore
GC Monitor
File Activity
Dispatch
Core Data
System Usage
OpenGL ES
Core Animation
Object Allocations
Leaks
CPU Sampler
Activity Monitor
Shark

warnings as errors

Target "Test" Info

General | Build | Rules | Properties | Comments

Configuration: Active (Debug)

warning

Show: All Settings

Setting	Value
Multiple Definition Types for Selector	☐
Nonvirtual Destructor	☐
Other Warning Flags	
Overloaded Virtual Functions	☐
Pedantic Warnings	☐
Pointer Sign Comparison	☑
Prototype Conversion	☐
Sign Comparison	☐
Strict Selector Matching	☐
Treat Missing Function Prototypes as Errors	☐
Treat Nonconformant Code Errors as Warnings	☐
Treat Warnings as Errors	☑
Typecheck Calls to printf/scanf	☑
Undeclared Selector	☐
Uninitialized Automatic Variables	☐
Unknown Pragma	☐
Unused Functions	☐
Unused Labels	☐
Unused Parameters	☐
Unused Values	☐

Enabling this option causes all warnings to be treated as errors.
[GCC_TREAT_WARNINGS_AS_ERRORS, -Werror]

Based On: Nothing

code organization

(cheaper maintenance)

iPhone ≠ computer

http://www.flickr.com/photos/oskay/365607662/

http://www.flickr.com/photos/blakespot/2379207825/

http://www.flickr.com/photos/oskay/365607591/

books

Developer
Connection
Recommended Title

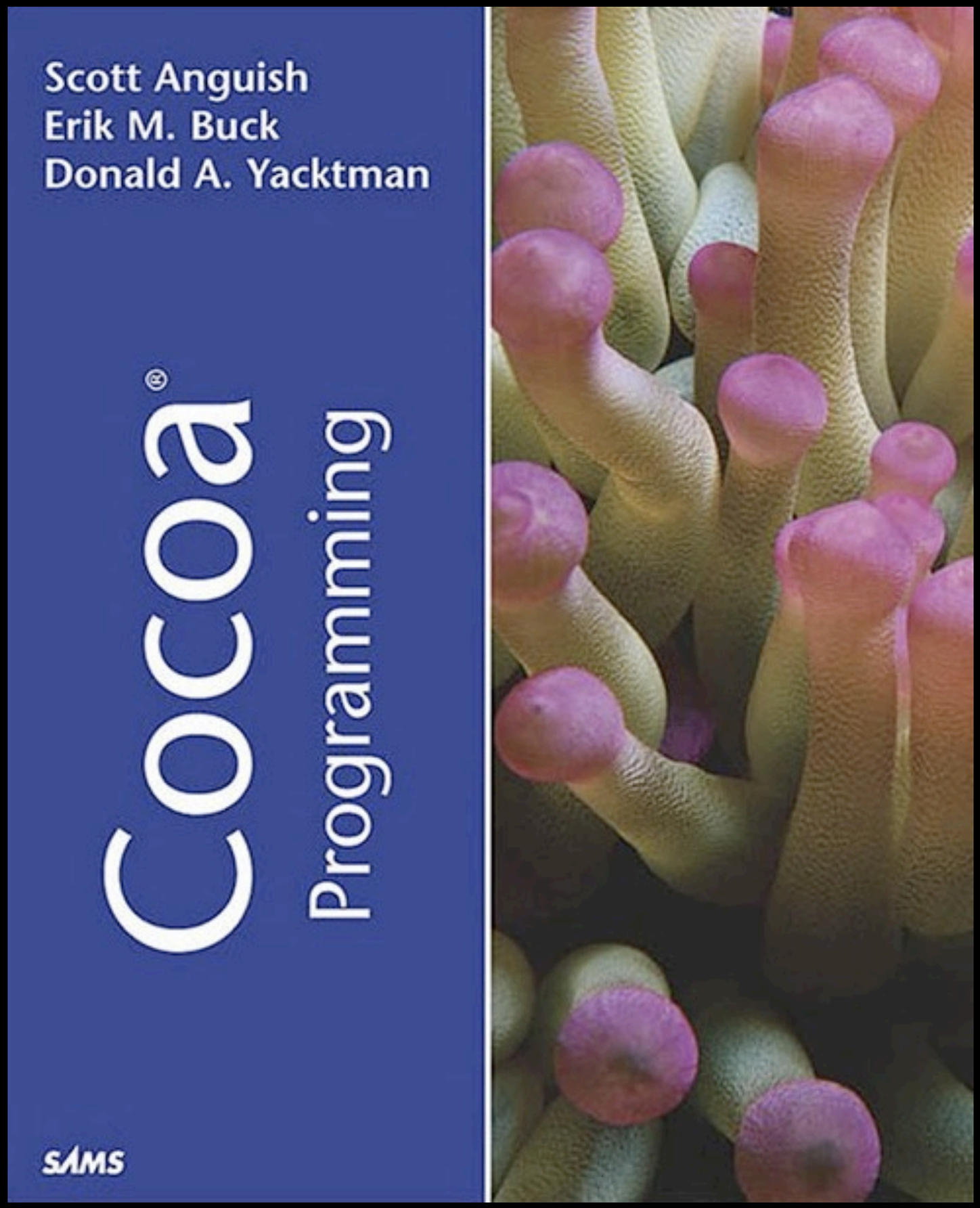
2nd Edition
Covers Mac OS X 10.2

Learning

Cocoa *with* Objective-C

O'REILLY

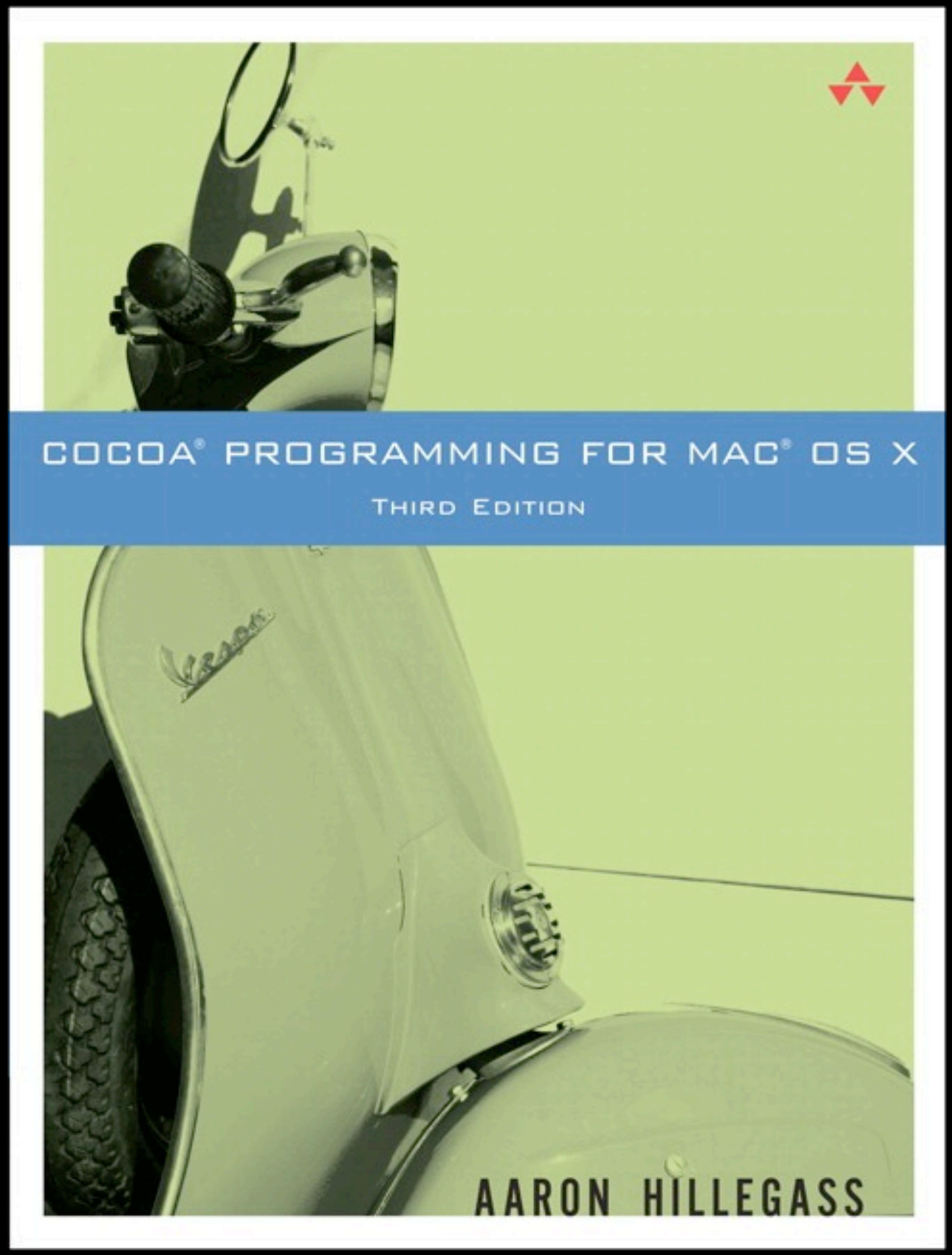
James Duncan Davidson & Apple Computer, Inc.


Scott Anguish
Erik M. Buck
Donald A. Yacktman
Cocoa®
Programming
SAMS

Programming in Objective-C

A complete introduction to the Objective-C language

Stephen G. Kochan


COCOA® PROGRAMMING FOR MAC® OS X
THIRD EDITION
AARON HILLEGASS

Updated and revised for SDK 3

A complete course in iPhone and iPod touch programming

Beginning iPhone 3 Development

Exploring the iPhone SDK

Dave Mark | Jeff LaMarche

Apress®

Learn the Coding Secrets of Master iPhone Designers and Developers

iPhone Cool Projects

Gary Bennett | Wolfgang Ante | Mike Ash | Benjamin Jackson
Neil Mix | Steven Peterson | Matthew "Canis" Rosenfeld

Apress®

Developing and Designing Cocoa Touch Applications

Programming the

iPhone User Experience

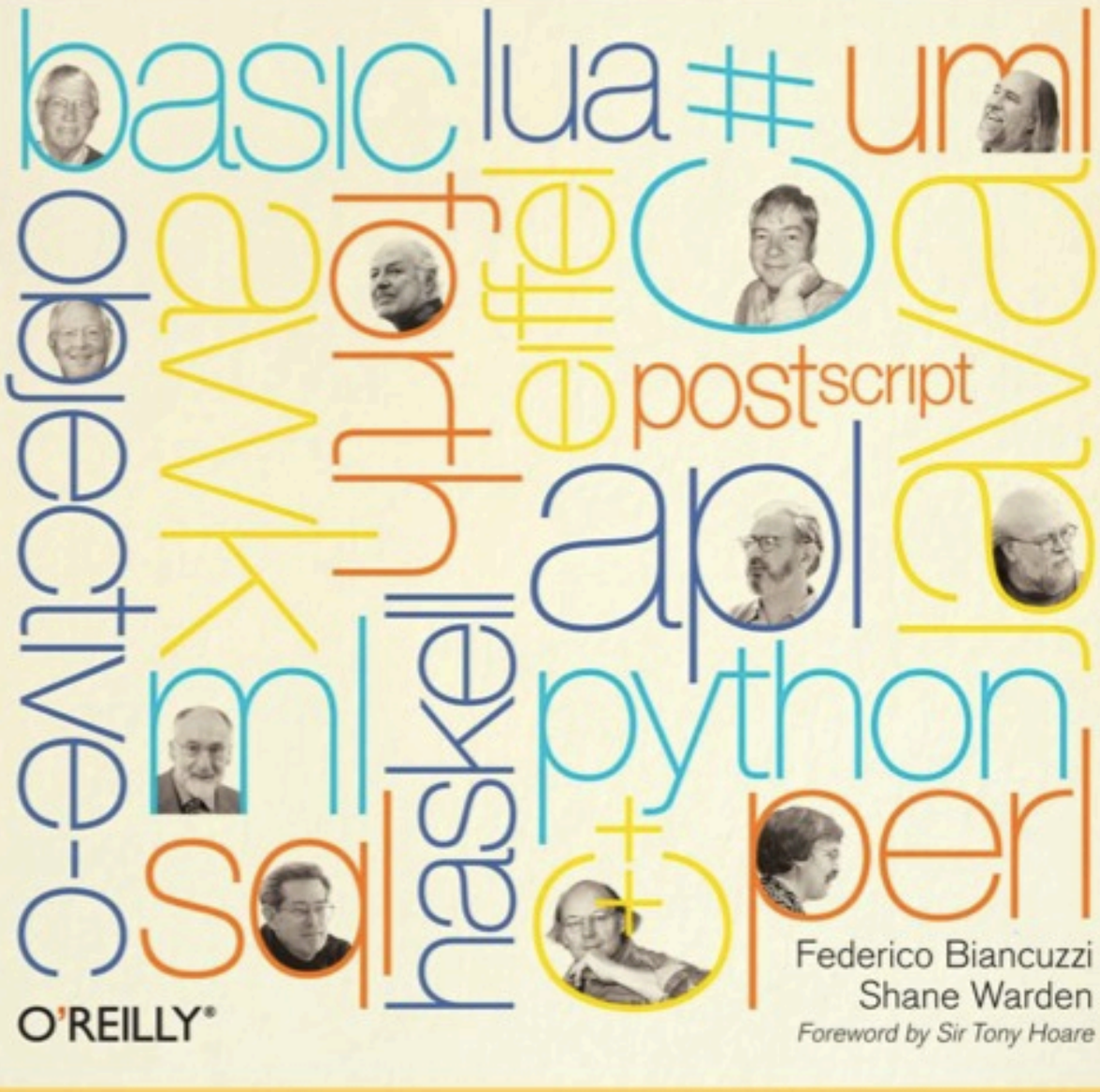
O'REILLY®

Toby Boudreaux

/Theory/In/Practice

Masterminds of Programming

Conversations with the Creators of Major Programming Languages



Federico Biancuzzi
Shane Warden

Foreword by Sir Tony Hoare

O'REILLY®

thanks!

questions?