

**Applications iPhone**

il est temps de s'y mettre !

HEIG-VD, Yverdon

Mercredi 16 juin, 18h00

---



**SEN**  **TE**

# Éléments clés dans le développement d'une application iPhone

Marc-Antoine Scheurer

`marco@sente.ch`

# Applications iPhone

- Application “web” ou “native”?
- Développer une application **native**
- Développer une **bonne** application native
- Développer pour d’autres appareils
- Liens utiles

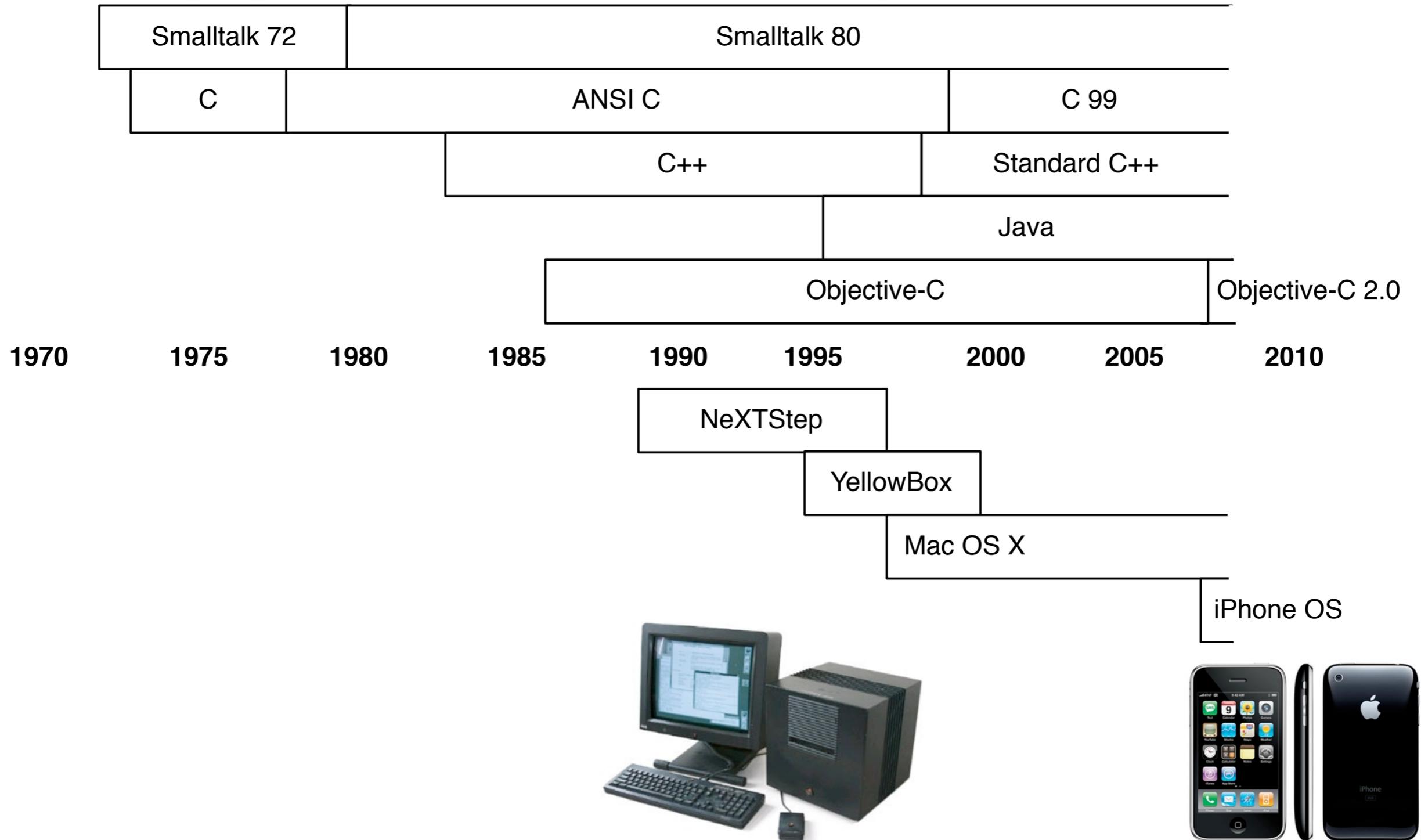
# “Web” ou “native”?

	<b>Web</b>	<b>Native</b>
<b>Langages</b>	HTML, Javascript, CSS	Objective C, C, C++
<b>Outils</b>	Dashcode, modèles tiers (jQuery, iUI, WebApp.net)	SDK (Xcode, Interface Builder, Cocoa Touch, ...)
<b>Environnement</b>	WebKit	iPhone OS (Unix)
<b>Distribution</b>	Libre	App Store
<b>Avantage</b>	“Portabilité”	Meilleure intégration et expérience utilisateur

# Développer une application native

- Acheter un Mac
- Devenir un développeur officiel (100\$ / an)
- Télécharger le SDK
- Apprendre Objective C
- Écrire une application
- Soumettre à l'App Store
- Se faire connaître

# Apprendre Objective C



# Un message d'Alan Kay

(inventeur de Smalltalk et du terme “orienté-objet”)

“Folks --

Just a gentle reminder that [...] Smalltalk is not only NOT about its syntax or the class library, it is not even about classes. I'm sorry that I long ago coined the term "objects" for this topic because it gets many people to focus on the lesser idea.

The big idea is "messaging" --that is what the kernel of Smalltalk/Squeak is all about [...]

The key in making great and growable systems is much more to design how its modules communicate rather than what their internal properties and behaviors should be.

If you focus on just messaging --and realize that a good metasystem can late bind the various 2nd level architectures used in objects-- then much of the language-, UI-, and OS based discussions [...] are really quite moot.”

Alan Kay, email to the Squeak mailing list, 10-10-1998

# Messages

- `[aPerson setYearOfBirth:1965];`
- `[aPerson moveNorth:2 east:5];`

<code>aPerson</code>	<code>receveur</code>
<code>moveNorth:2 east:5</code>	<code>message</code>
<code>moveNorth: east:</code>	nom de méthode (sélecteur)
<code>2,5</code>	arguments

# Écrire une application

- Avoir une idée
- Prévoir du temps
  - par exemple: 2 semaines
- Développer
- Tester (au bureau, dehors, en mode “avion”)

# Soumettre à l'App Store

- Respecter les conditions
  - techniques
  - sociales / politiques
  - commerciales
- Description, mots-clés, copies d'écran
- Délai < 7 jours pour 95% des soumissions

# Se faire connaître

- Marketing et communication
  - Presse, blogs, Facebook, Twitter, etc.
- Hit parade des ventes sur l'App Store
  - Top 100, Top 25, Top 10
  - Moyenne mobile sur 3 jours
  - Maximiser l'impact initial

# Développer une bonne application native

- Maîtriser l'ergonomie
- Maîtriser les choix techniques
- Maîtriser les outils de développement

# Maîtriser l'ergonomie

- Se limiter
- Respecter les standards
- Faire un effort sur le graphisme
- Prototyper... ou non
  - Papier, Photoshop, Balsamiq, Omnigraffle

# Maîtriser les choix techniques

- Ressources limitées
  - CPU
  - Mémoire
  - Système de fichiers
  - Bande passante
  - Énergie

# Ressources

	<b>3G</b>	<b>3GS</b>	<b>4</b>
<b>CPU</b> horloge L1 L2	ARM 11 412 MHz 32 KB -	ARM Cortex A8 600 MHz 64 KB 256 KB	A4 (Cortex A8?) 1 GHz? 64 KB 640 KB
<b>GPU</b> triangles / s	PowerVR MBX Lite 0.6 M	PowerVR SGX 5.2 M	A4 (PowerVR SGX) 10 M ?? 20 M ??
<b>Mémoire RAM</b> disponible	128 MB 40 MB	256 MB 150 MB	512 MB ?
<b>Mémoire Flash</b> disponible	8 - 32 GB 2 GB	16 - 32 GB 2 GB	16 - 32 GB 2 GB
<b>Écran</b>	480x320	480x320	960x640
<b>En plus</b>	GPS	Boussole	Gyroscope
<b>Batterie</b>	6h sur WiFi	9h sur WiFi	10h sur WiFi

# Énergie

- 3G, WiFi, Bluetooth, GPS: 2 W
- CPU + GPU: 800 mW
- Écran: 200 mW
- (Appareil 3G)

# Bande passante

- WiFi, 3G, Edge
- Appareil en déplacement (changement de cellule)
- Économiser le nombre de connections (latence)

# Maîtriser les outils de développement

- Objective C
- Cocoa Touch
- Xcode
- Interface Builder
- Simulateur
- Instruments
- Unit Testing

# Objective C

- Gestion manuelle de la mémoire par comptage de référence
- Optimisations

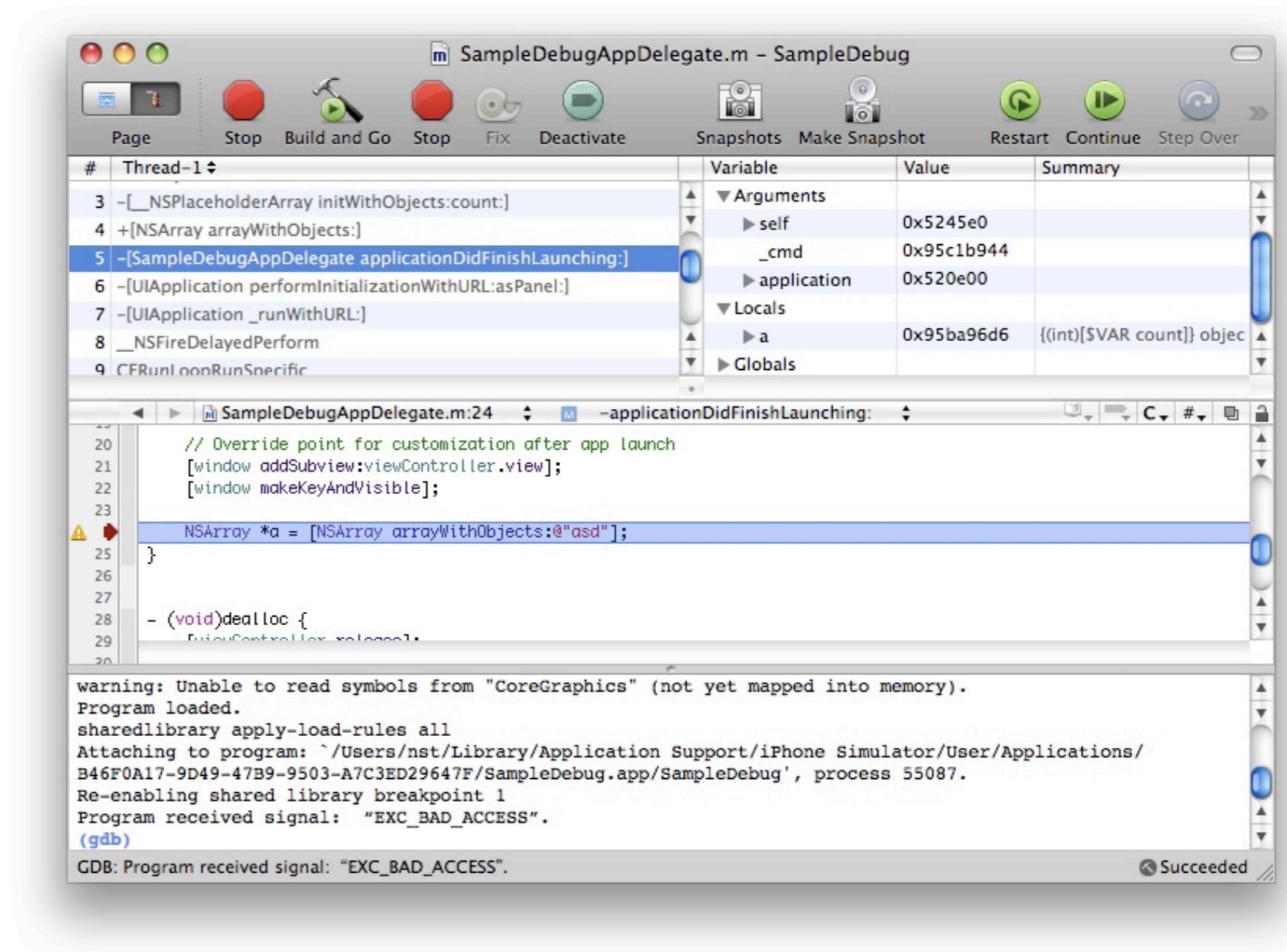
# Cocoa Touch, frameworks

- Foundation
- UIKit
- MapKit, CoreLocation
- CoreData
- OpenGL
- CoreGraphics, CoreAudio (C)
- ...



# Xcode

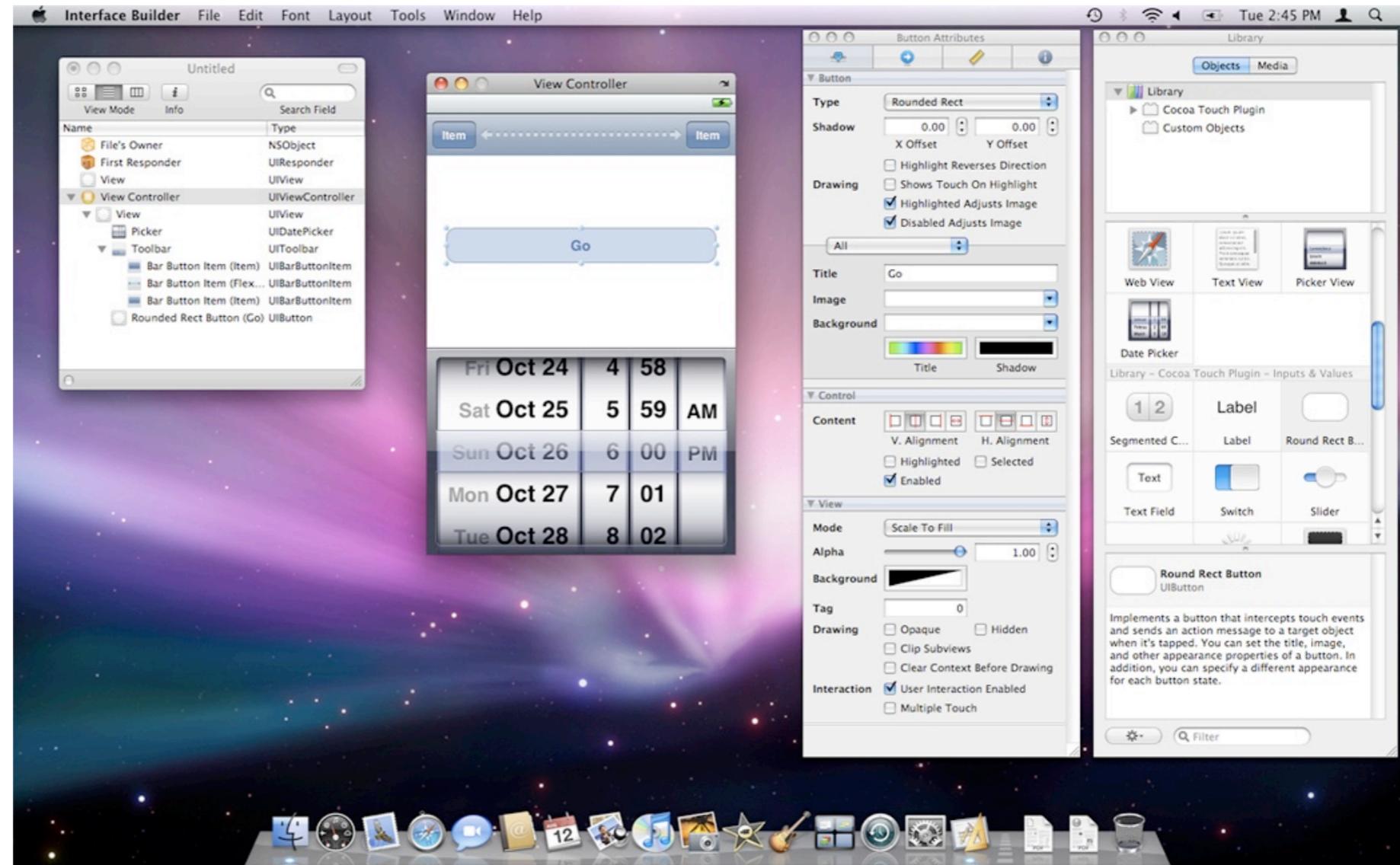
- Organisation
- Édition
- Compilation
- Debug
- Documentation





# Interface Builder

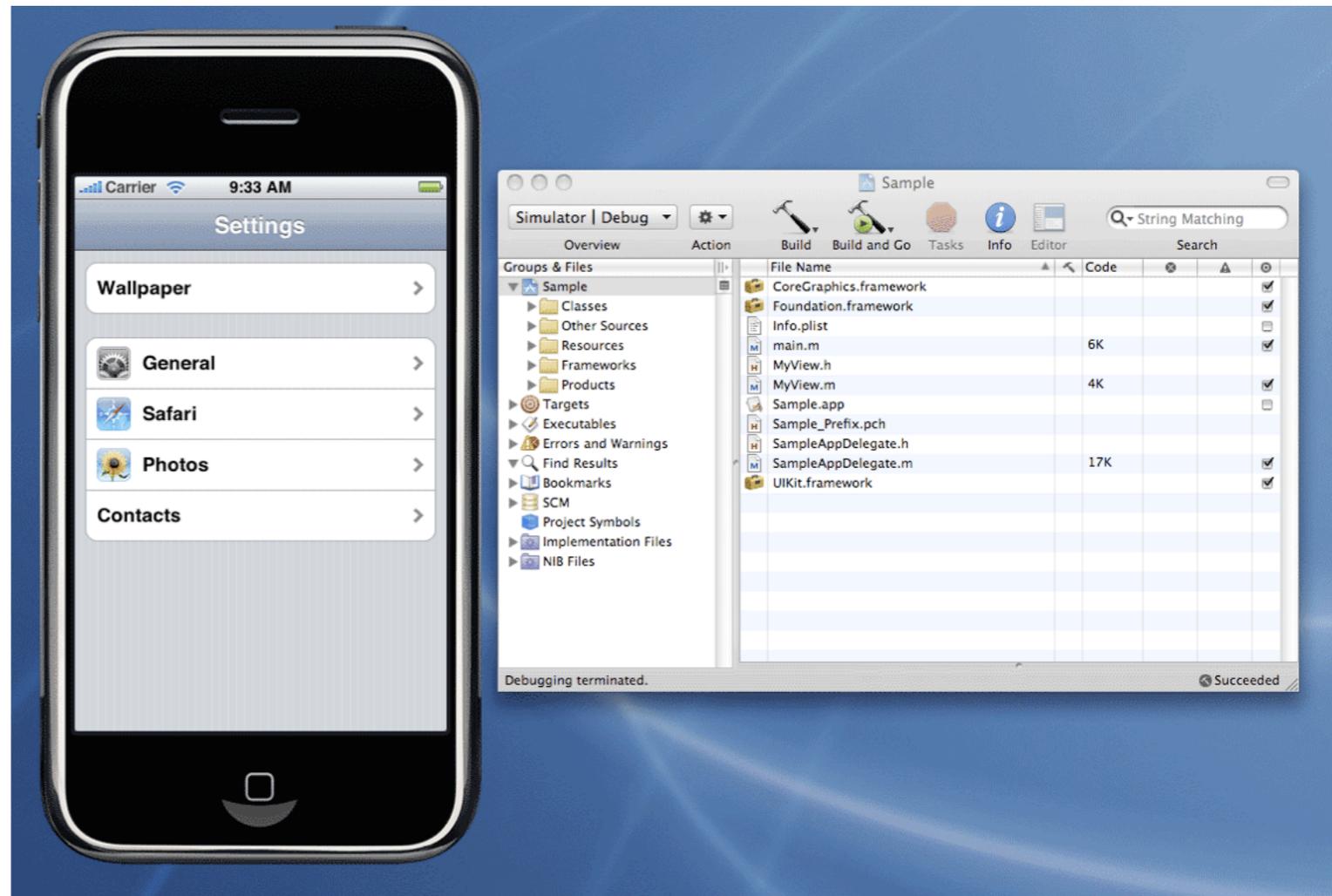
- Éditeur d'interface
- Pas un générateur de code
- Objets archivés par IB, dé-archivés par l'application





# Simulateur

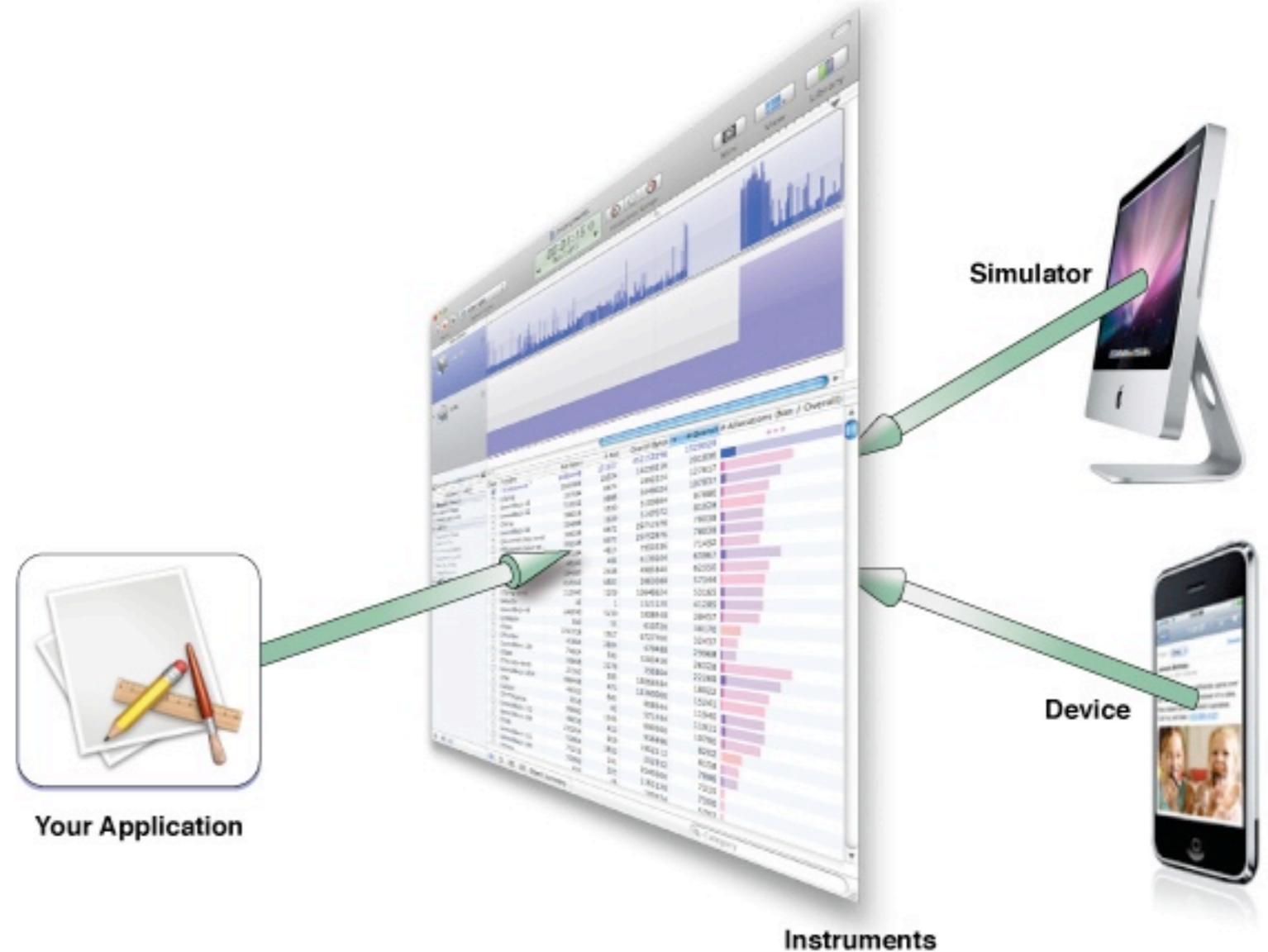
- Simulateur logiciel  $\neq$  émulateur matériel
- Code exécuté sur i386 et non sur ARM
- 2.5 GHz (Mac) et non 800 MHz (3G)
- Mémoire illimitée
- ... tester fréquemment sur l'appareil





# Instruments

- Mesure des performances
- CPU
- Mémoire
- CoreData
- OpenGL
- etc.



# Unit Testing

- OCUUnit
- Écrit par Sente en 1998 d'après l'article Smalltalk de Kent Beck
- Détection et exécution automatique des tests
- Intégré dans Xcode par Apple en 2005

```
16 - (void)testMyAddition {  
17     STAssertEquals (1 + 1, 2, @"");  
18     STAssertEquals (1 + 1, 3, @"");  
19 }  
20
```

error: -[MyTests testMyAddition]: '2' should be equal to '3':

# Développer pour d'autres appareils

- iPad
- Android
- Blackberry
- Nokia
- Windows mobile
- SDK alternatifs multiplateformes: interdits

# iPad

- Ergonomie différente
- Ressources différentes
- Mêmes outils et techniques
- Développement spécialisé

# Autres

Android	Java SDK, WebKit
Nokia	Qt (C++), WebKit, OVI SDK, Flash lite
Blackberry	Java ME
Windows Mobile	Visual C++, Visual C#, Visual Basic

# Multiplateformes

- PhoneGap, autorisé?  
<http://www.phonegap.com/>
- Rhodes, vraisemblablement non autorisé  
<http://rhomobile.com/products/rhodes/>
- Titanium, autorisé?  
<http://www.appcelerator.com/>
- ...

# Liens utiles

- **Apple**

<http://developer.apple.com/iphone>

- **Forum développeurs**

<http://www.iphonedevsdk.com/>

<http://stackoverflow.com>

- **Applications web**

**iUI** <http://code.google.com/p/iui/>

**Webapp.net** <http://webapp-net.com/>

**jQueryTouch** <http://jqtouch.com/>

- **Prototypage**

**Balsamiq** <http://www.balsamiq.com/>

**OmniGraffle** <http://www.omnigroup.com/products/omnigraffle/>

- **Cours pour développeurs**

**Stanford iPhone Application Development** (anglais, gratuit) sur **iTunesU**

<http://deimos3.apple.com/WebObjects/Core.woa/Browse/itunes.stanford.edu.3124430053.03124430055>

**Sen:te** (français, payant)

<http://www.iphone-class.com>