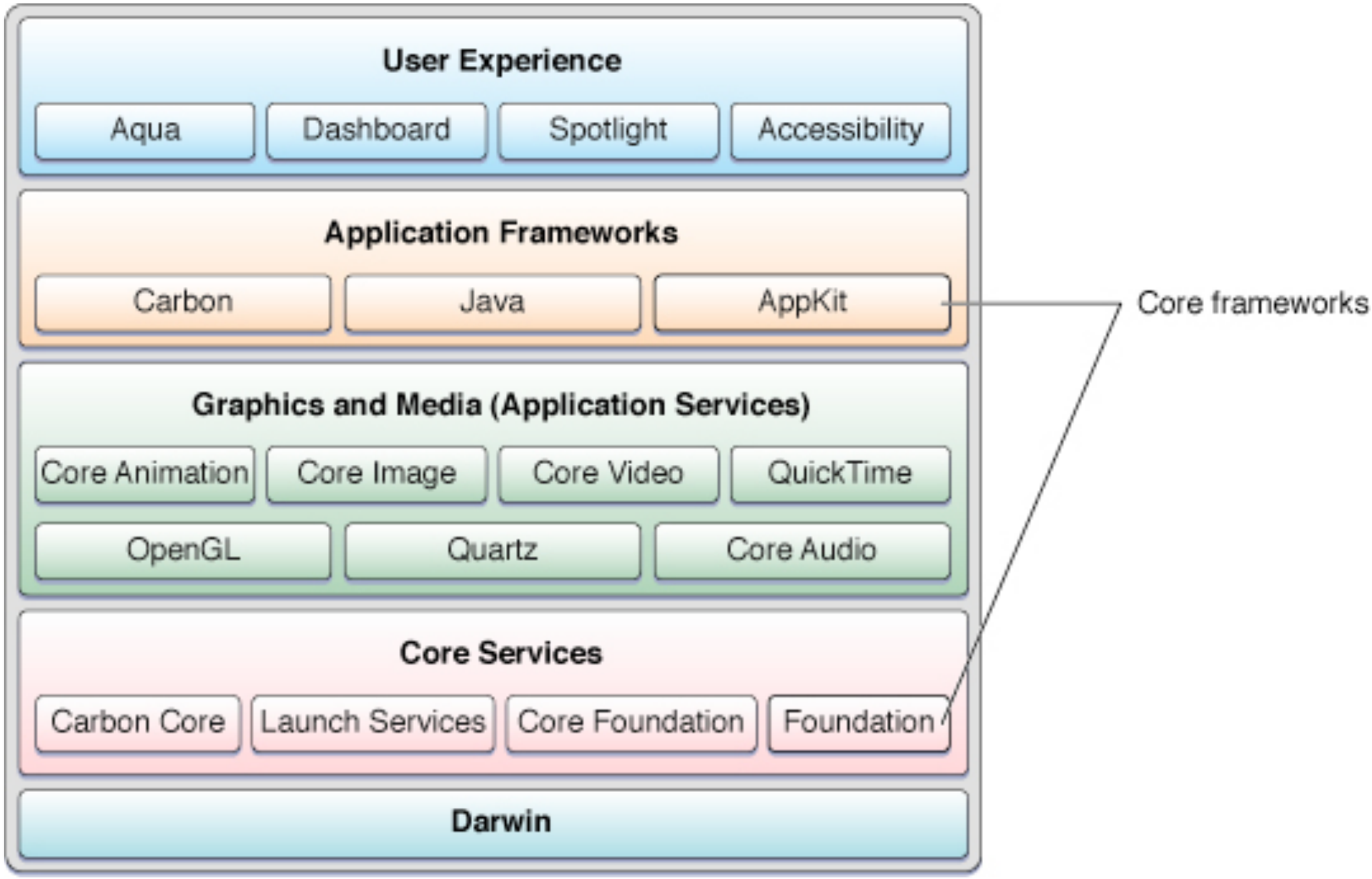
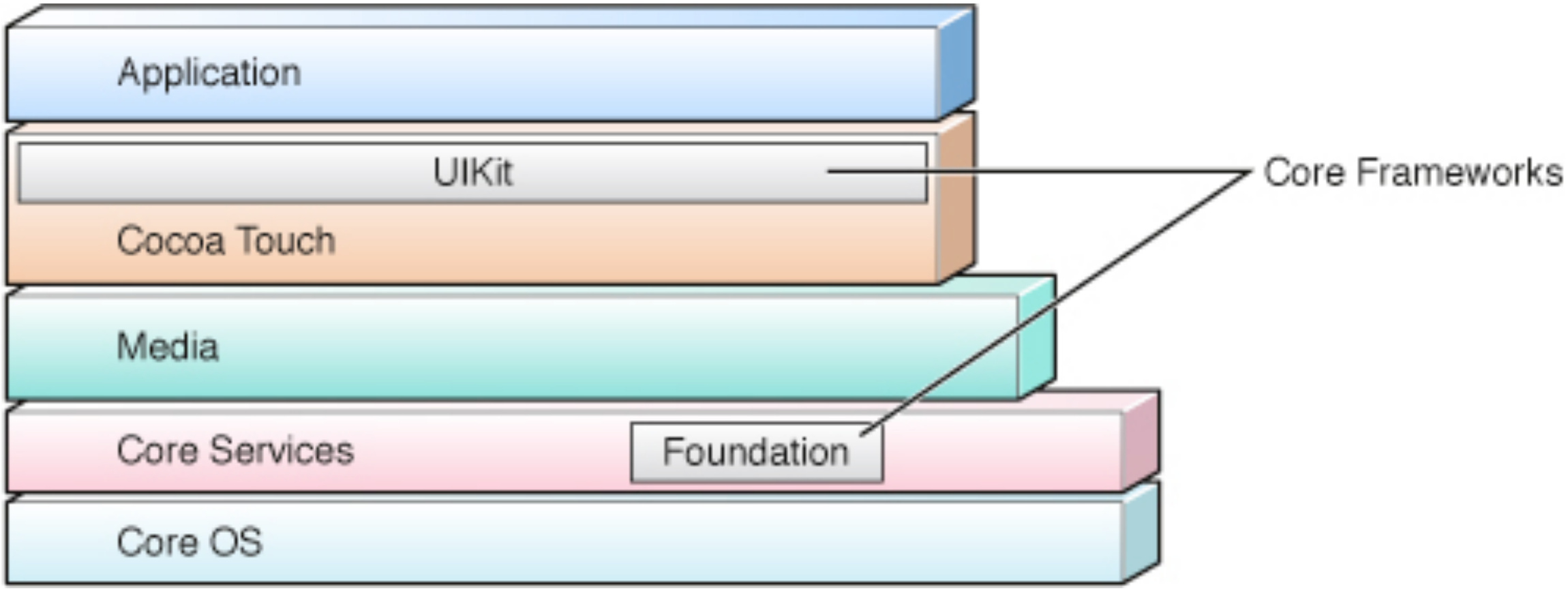
Cocoa

Jean-Baptiste.Yunes@univ-paris-diderot.fr

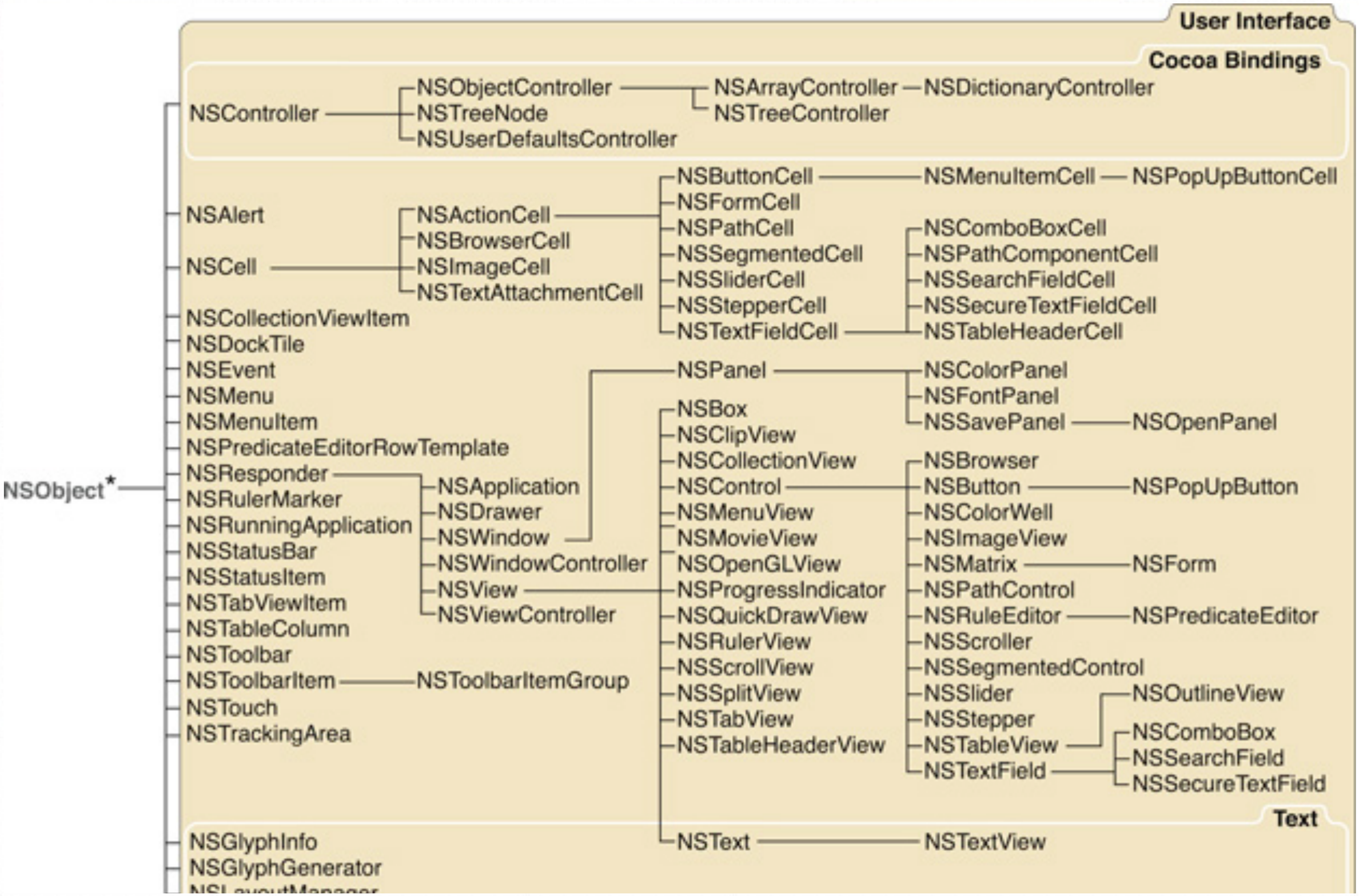
2014—2015

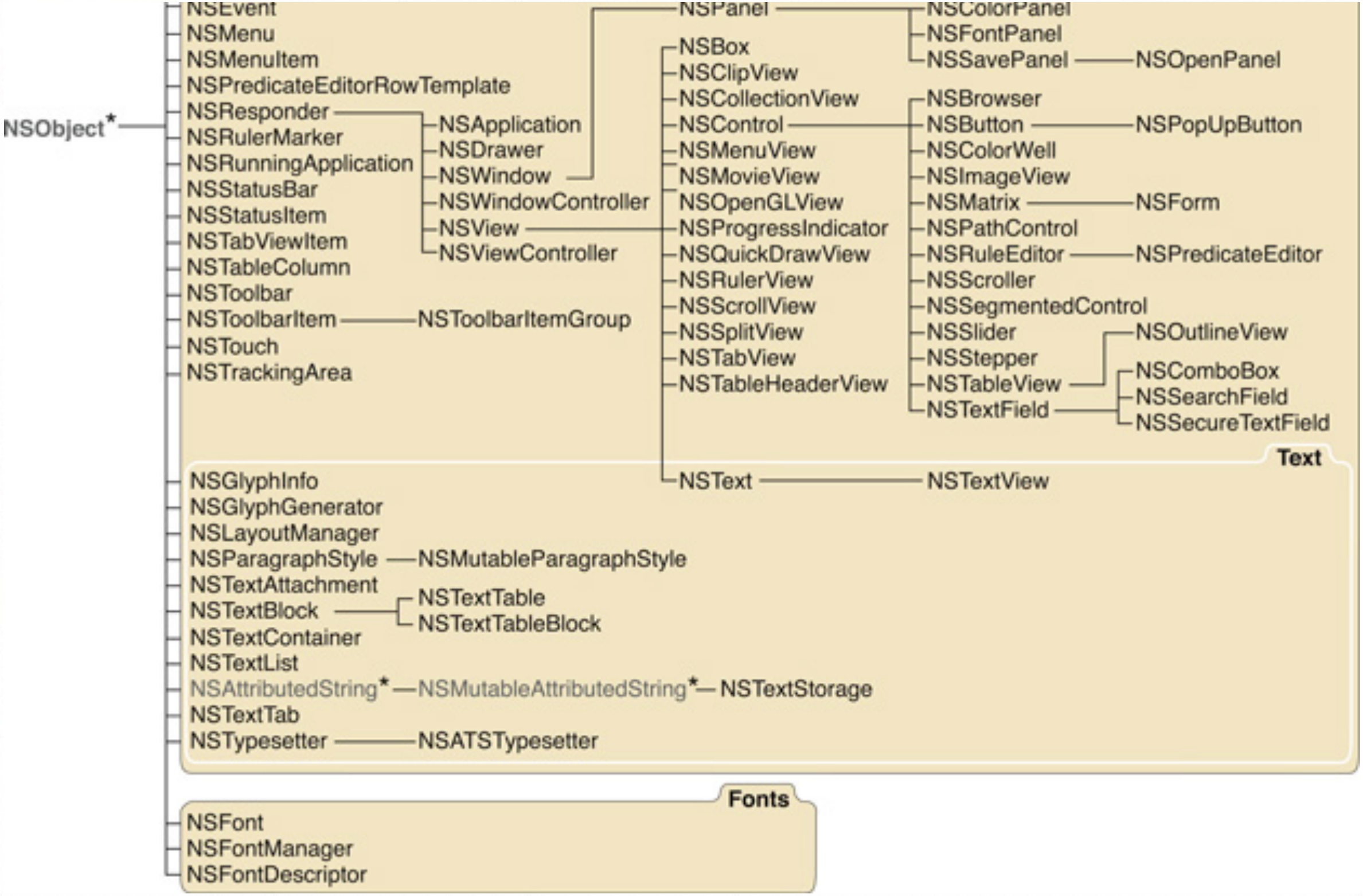
- deux Cocoa (pour le prix d'un) !
  - pour OSX
    - Foundation+AppKit
  - pour iOS
    - Foundation+UIKit
- Appkit : NS... (ex : NSButton)
- UIKit : UI... (ex : UIButton)


User Experience
Aqua
Dashboard
Spotlight
Accessibility
Application Frameworks
Carbon
Java
AppKit
Graphics and Media (Application Services)
Core Animation
Core Image
Core Video
QuickTime
OpenGL
Quartz
Core Audio
Core Services
Carbon Core
Launch Services
Core Foundation
Foundation
Darwin
Core frameworks

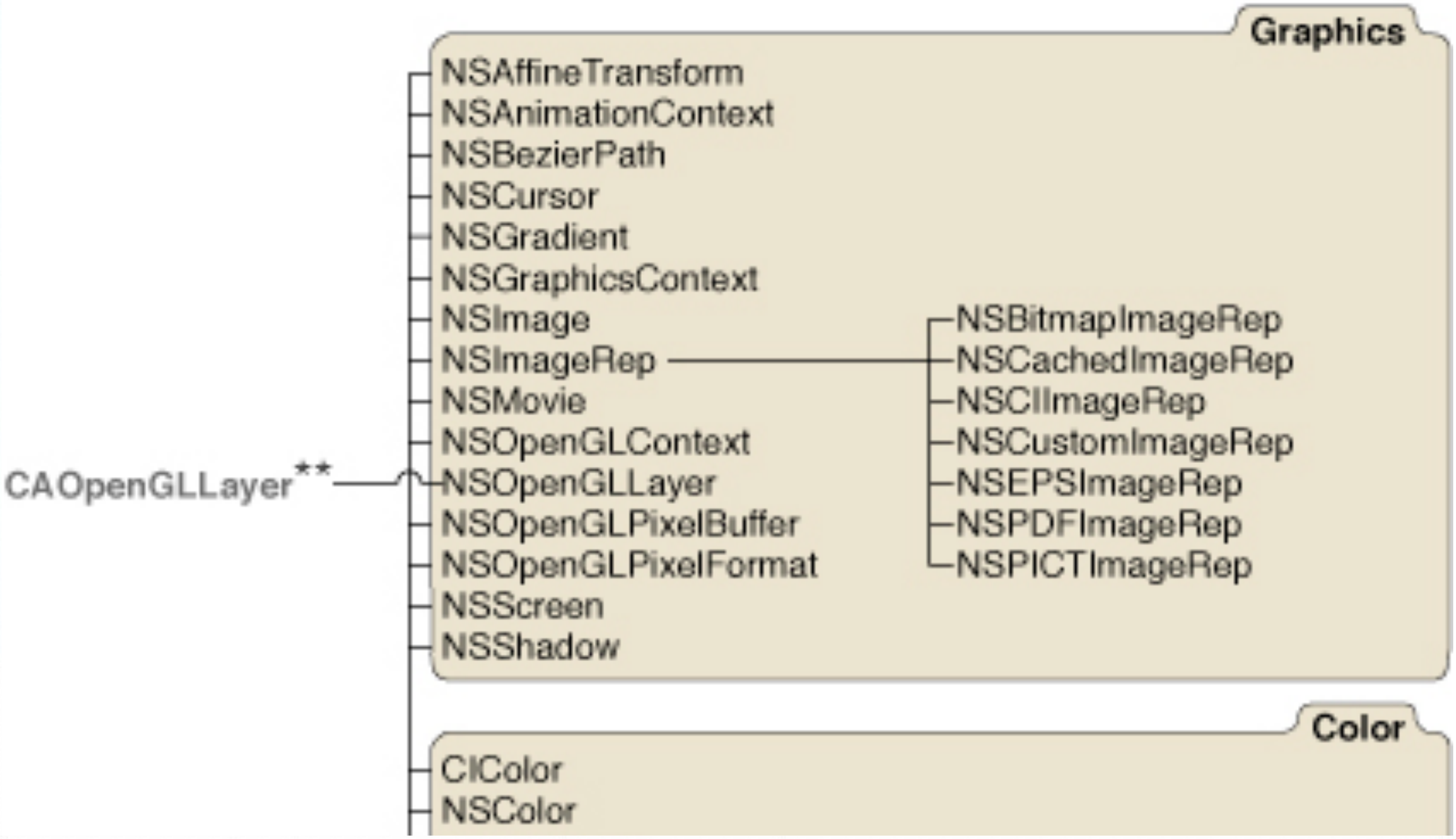

Application
UIKit
Cocoa Touch
Media
Core Services
Foundation
Core OS
Core Frameworks

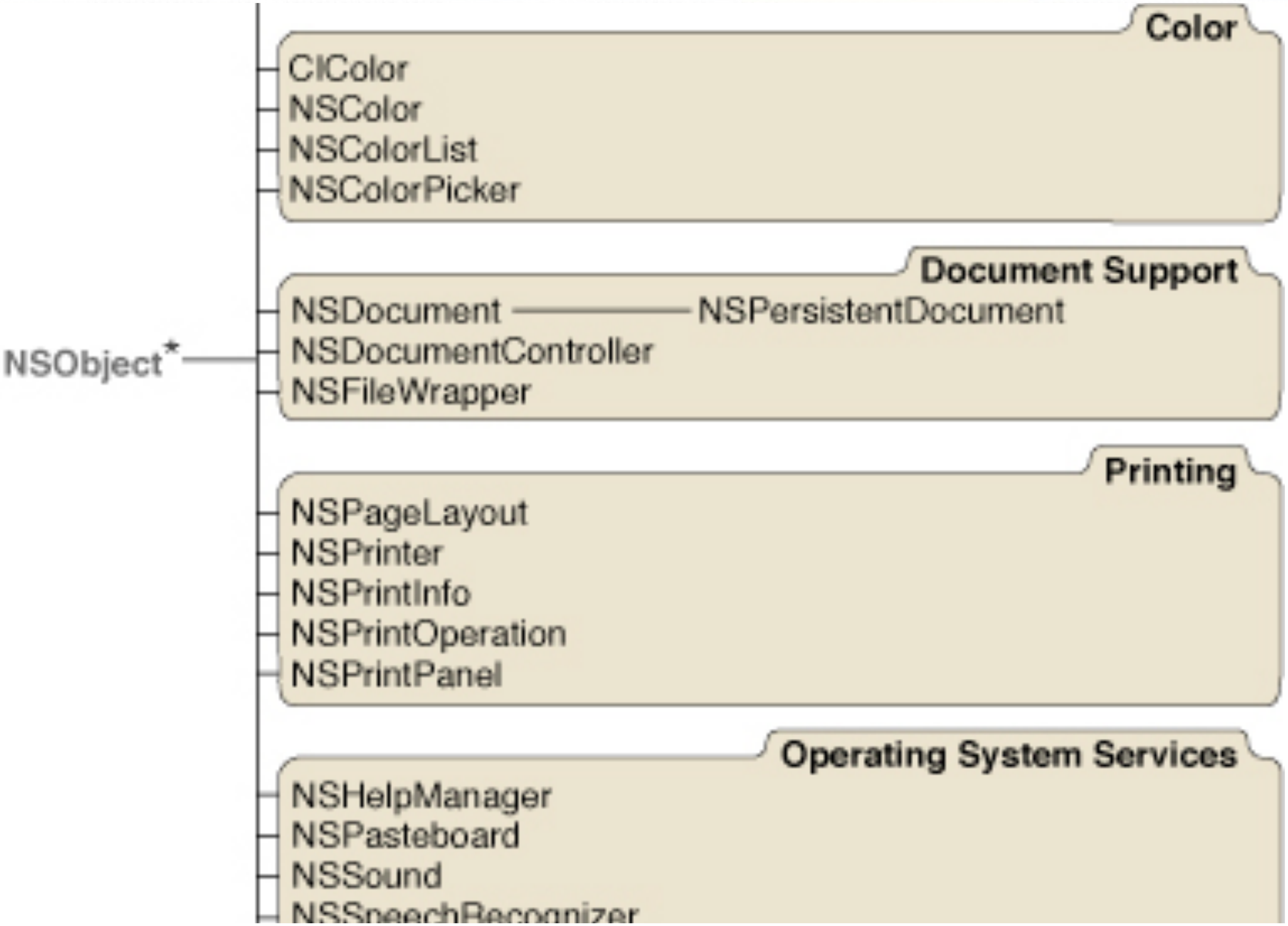
- pour simplifier donc :
  - Cocoa c’est Foundation + l’interface graphique
    - disons que ce sont les premières choses que l’on voit de Cocoa


User Interface
Cocoa Bindings
NSObject*
NSController
NSObjectController
NSTreeNode
NSUserDefaultsController
NSArrayController
NSTreeController
NSDictionaryController
NSAlert
NSCell
NSActionCell
NSBrowserCell
NSImageCell
NSTextAttachmentCell
NSButtonCell
NSFormCell
NSPathCell
NSSegmentedCell
NSSliderCell
NSStepperCell
NSTextFieldCell
NSMenuItemCell
NSPopUpButtonCell
NSComboBoxCell
NSPathComponentCell
NSSearchFieldCell
NSSecureTextFieldCell
NSTableHeaderCell
NSCollectionViewItem
NSDockTile
NSEvent
NSMenu
NSMenuItem
NSPredicateEditorRowTemplate
NSResponder
NSRulerMarker
NSRunningApplication
NSStatusBar
NSStatusItem
NSTabViewItem
NSTableColumn
NSToolbar
NSToolbarItem
NSTouch
NSTrackingArea
NSApplication
NSDrawer
NSWindow
NSWindowController
NSView
NSViewController
NSToolbarItemGroup
NSPanel
NSColorPanel
NSFontPanel
NSSavePanel
NSOpenPanel
NSBox
NSClipView
NSCollectionView
NSControl
NSMenuView
NSMovieView
NSOpenGLView
NSProgressIndicator
NSQuickDrawView
NSRulerView
NSScrollView
NSSplitView
NSTabView
NSTableHeaderView
NSBrowser
NSButton
NSPopUpButton
NSColorWell
NSImageView
NSMatrix
NSForm
NSPathControl
NSRuleEditor
NSPredicateEditor
NSScroller
NSSegmentedControl
NSSlider
NSStepper
NSTableView
NSTextField
NSOutlineView
NSComboBox
NSSearchField
NSSecureTextField
Text
NSGlyphInfo
NSGlyphGenerator
NSText
NSTextView


NSObject*
NSEvent
NSMenu
NSMenuItem
NSPredicateEditorRowTemplate
NSResponder
NSRulerMarker
NSRunningApplication
NSStatusBar
NSStatusItem
NSTabViewItem
NSTableColumn
NSToolbar
NSToolbarItem
NSTouch
NSTrackingArea
NSApplication
NSDrawer
NSWindow
NSWindowController
NSView
NSViewController
NSToolbarItemGroup
NSPanel
NSBox
NSClipView
NSCollectionView
NSControl
NSMenuView
NSMovieView
NSOpenGLView
NSProgressIndicator
NSQuickDrawView
NSRulerView
NSScrollView
NSSplitView
NSTabView
NSTableHeaderView
NSColorPanel
NSFontPanel
NSSavePanel
NSOpenPanel
NSBrowser
NSButton
NSPopUpButton
NSColorWell
NSImageView
NSMatrix
NSForm
NSPathControl
NSRuleEditor
NSPredicateEditor
NSScroller
NSSegmentedControl
NSSlider
NSStepper
NSTableView
NSTextField
NSOutlineView
NSComboBox
NSSearchField
NSSecureTextField
Text
NSGlyphInfo
NSGlyphGenerator
NSLayoutManager
NSParagraphStyle
NSMutableParagraphStyle
NSTextAttachment
NSTextBlock
NSTextTable
NSTextTableBlock
NSTextContainer
NSTextList
NSAttributedString*
NSMutableAttributedString*
NSTextStorage
NSTextTab
NSTypesetter
NSATSTypesetter
NSText
NSTextView
Fonts
NSFont
NSFontManager
NSFontDescriptor

Objective-C Application Kit Continued




NSObject*
Color
CIColor
NSColor
NSColorList
NSColorPicker
Document Support
NSDocument
NSPersistentDocument
NSDocumentController
NSFileWrapper
Printing
NSPageLayout
NSPrinter
NSPrintInfo
NSPrintOperation
NSPrintPanel
Operating System Services
NSHelpManager
NSPasteboard
NSSound
NSSpeechRecognizer

NSPageLayout
NSPrinter
NSPrintInfo
NSPrintOperation
NSPrintPanel

Operating System Services

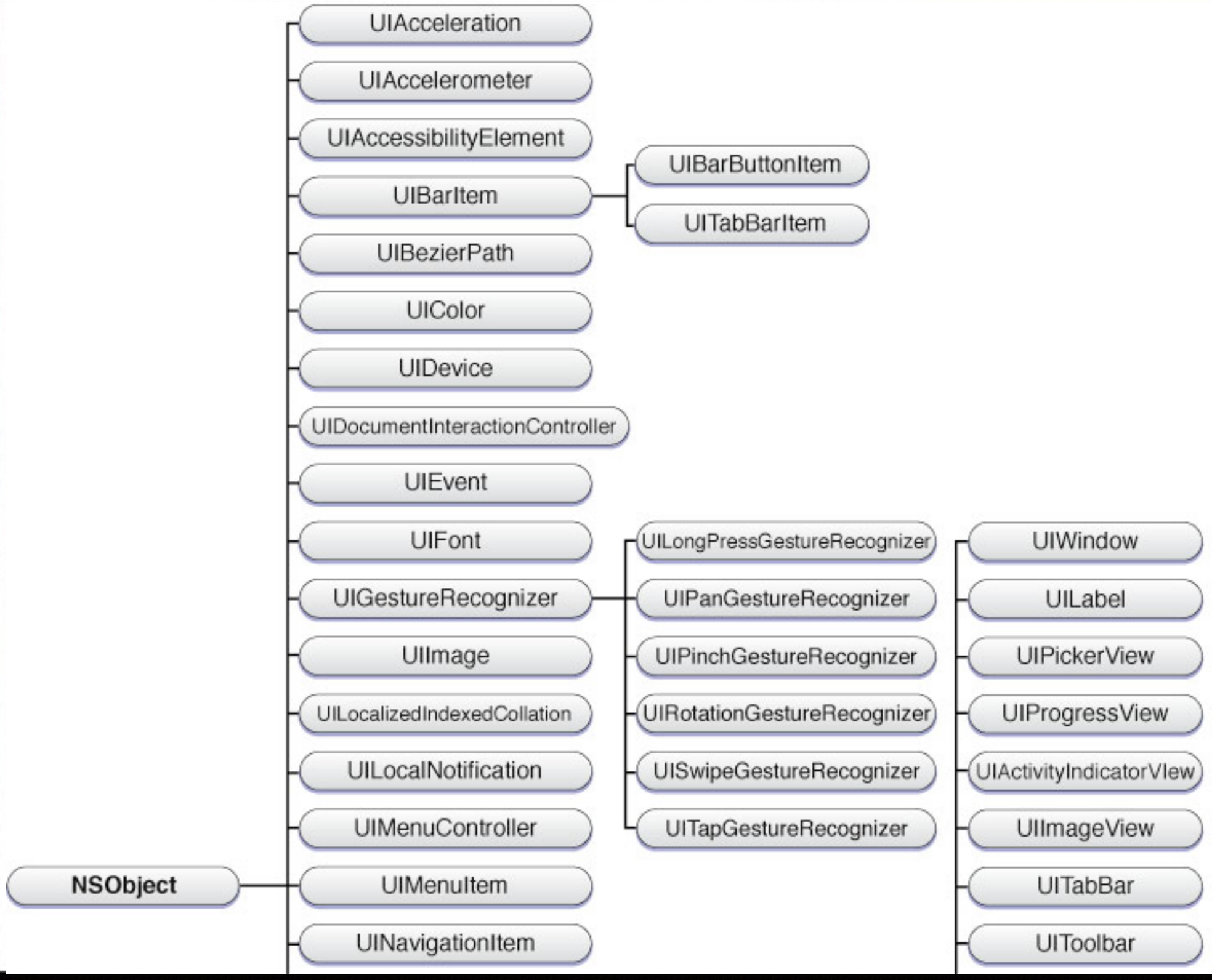
NSHelpManager
NSPasteboard
NSSound
NSSpeechRecognizer
NSSpeechSynthesizer
NSSpellChecker
NSWorkspace

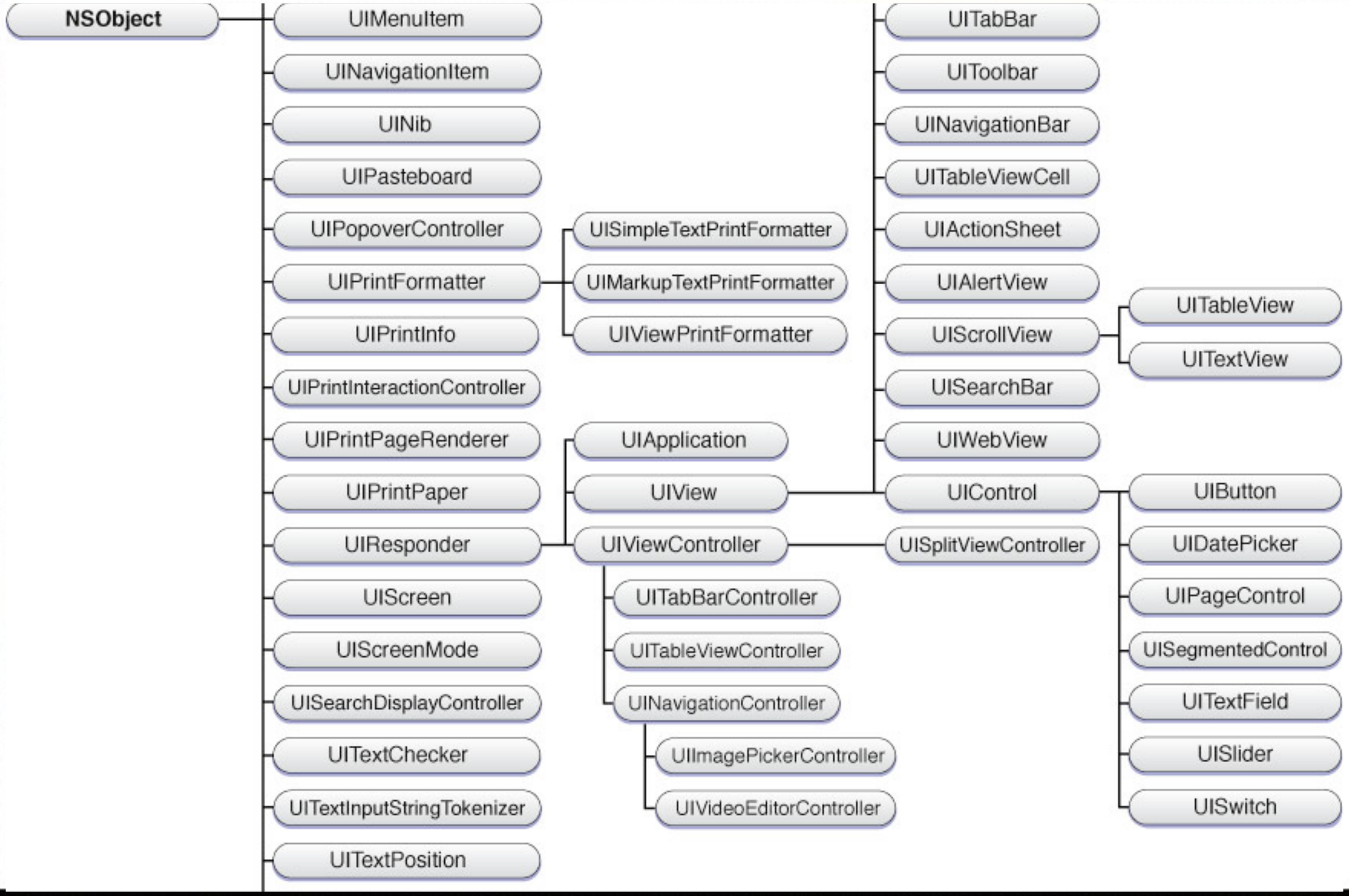
International Character Input Support

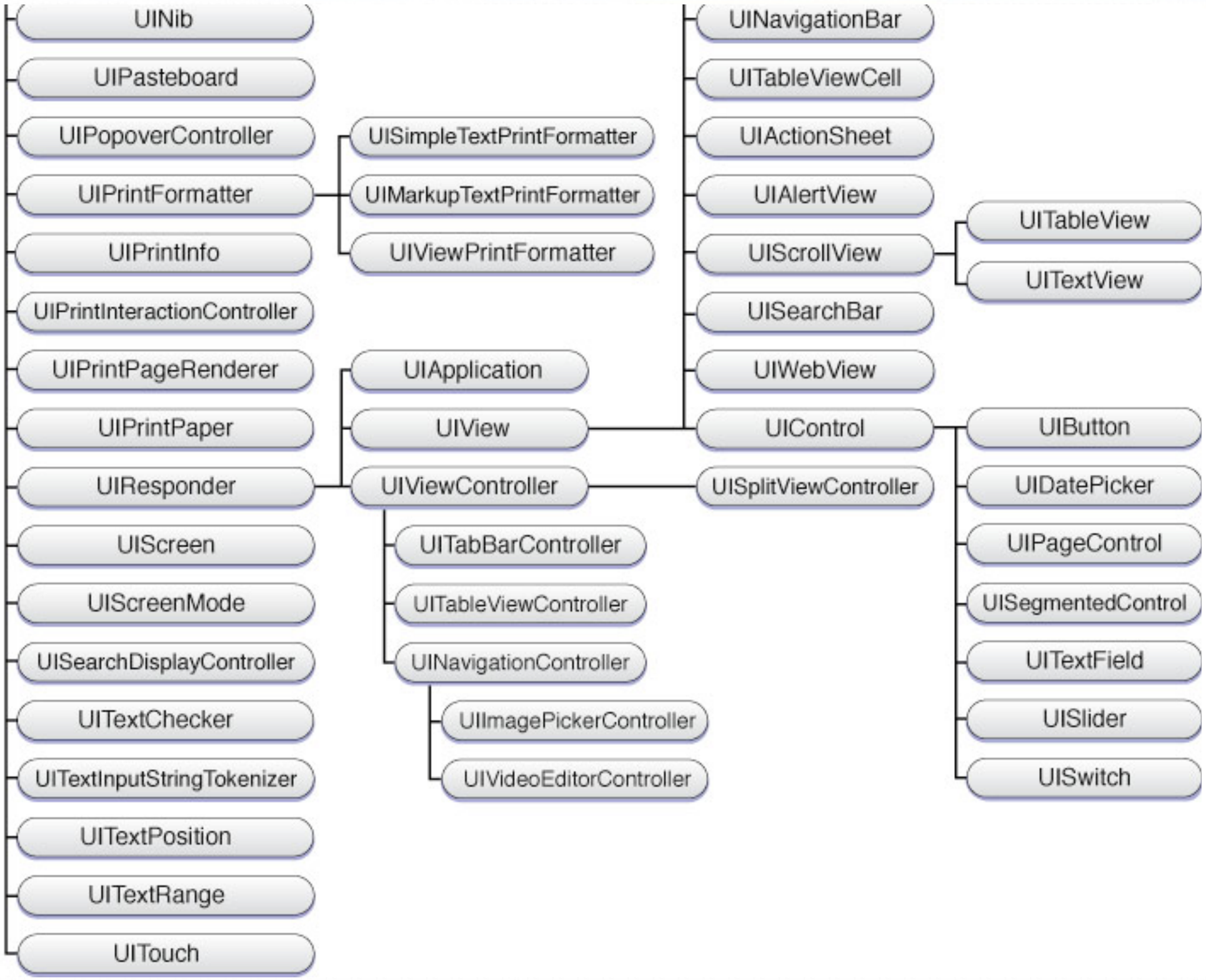
NSTextInputContext

Interface Builder Support

NSNib
NSNibConnector
- NSNibControlConnector
- NSNibOutletConnector


NSObject
UIAcceleration
UIAccelerometer
UIAccessibilityElement
UIBarItem
UIBarButtonItem
UITabBarItem
UIBezierPath
UIColor
UIDevice
UIDocumentInteractionController
UIEvent
UIFont
UIGestureRecognizer
UILongPressGestureRecognizer
UIPanGestureRecognizer
UIPinchGestureRecognizer
UIRotationGestureRecognizer
UISwipeGestureRecognizer
UITapGestureRecognizer
UIImage
UILocalizedIndexedCollation
UILocalNotification
UIMenuController
UIMenuItem
UINavigationItem
UIWindow
UILabel
UIPickerView
UIProgressView
UIActivityIndicatorView
UIImageView
UITabBar
UIToolbar

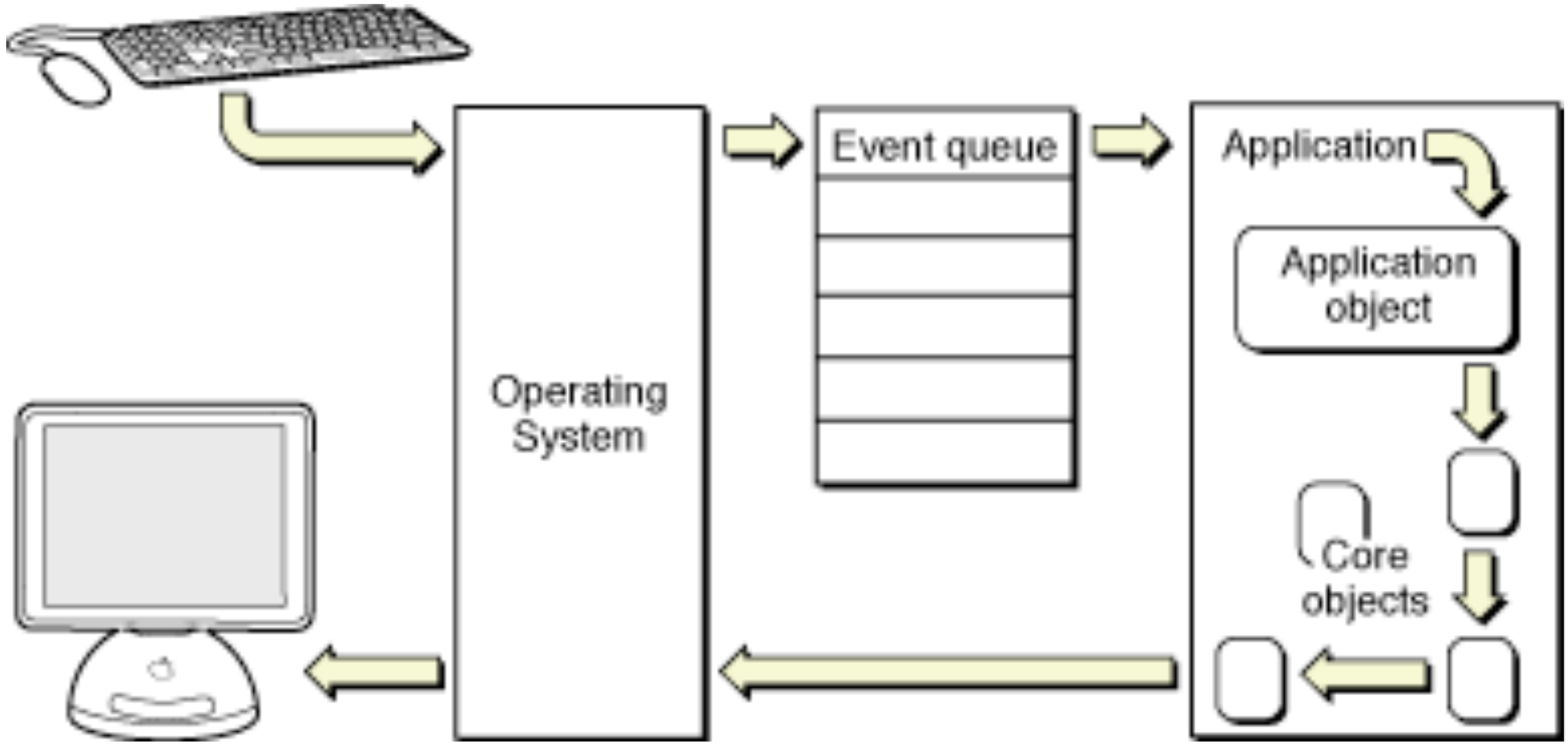

NSObject
UIMenuItem
UINavigationItem
UINib
UIPasteboard
UIPopoverController
UIPrintFormatter
UIPrintInfo
UIPrintInteractionController
UIPrintPageRenderer
UIPrintPaper
UIResponder
UIScreen
UIScreenMode
UISearchDisplayController
UITextChecker
UITextInputStringTokenizer
UITextPosition
UISimpleTextPrintFormatter
UIMarkupTextPrintFormatter
UIViewPrintFormatter
UIApplication
UIView
UIViewController
UITabBarController
UITableViewController
UINavigationController
UIImagePickerController
UIVideoEditorController
UITabBar
UIToolbar
UINavigationBar
UITableViewCell
UIActionSheet
UIAlertView
UIScrollView
UISearchBar
UIWebView
UIControl
UISplitViewController
UITableView
UITextView
UIButton
UIDatePicker
UIPageControl
UISegmentedControl
UITextField
UISlider
UISwitch


UINib
UIPasteboard
UIPopoverController
UIPrintFormatter
UIPrintInfo
UIPrintInteractionController
UIPrintPageRenderer
UIPrintPaper
UIResponder
UIScreen
UIScreenMode
UISearchDisplayController
UITextChecker
UITextInputStringTokenizer
UITextPosition
UITextRange
UITouch
UISimpleTextPrintFormatter
UIMarkupTextPrintFormatter
UIViewPrintFormatter
UIApplication
UIView
UIViewController
UITabBarController
UITableViewController
UINavigationController
UIImagePickerController
UIVideoEditorController
UINavigationBar
UITableViewCell
UIActionSheet
UIAlertView
UIScrollView
UISearchBar
UIWebView
UIControl
UISplitViewController
UITableView
UITextView
UIButton
UIDatePicker
UIPageControl
UISegmentedControl
UITextField
UISlider
UISwitch

La boucle d'évènements

Jean-Baptiste.Yunes@univ-paris-diderot.fr / 2014–2015 Master II

La boucle d'événements



Usage des classes Cocoa

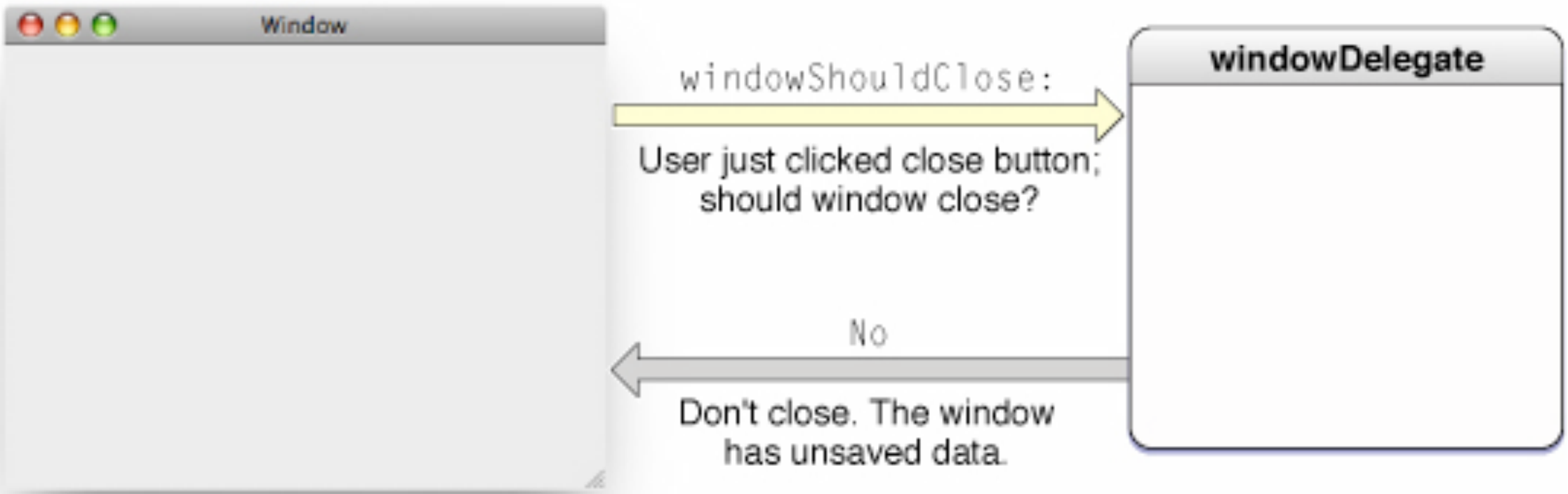
Jean-Baptiste.Yunes@univ-paris-diderot.fr / 2014–2015 Master II

- 4 façons d'utiliser des classes Cocoa
  - nature : on utilise des objets prédéfinis que l'on paramètre (ex : NSButton)
  - sans le savoir : des objets sont créés en arrière-plan
  - générique (ex : UIView)
  - par délégation ou notification

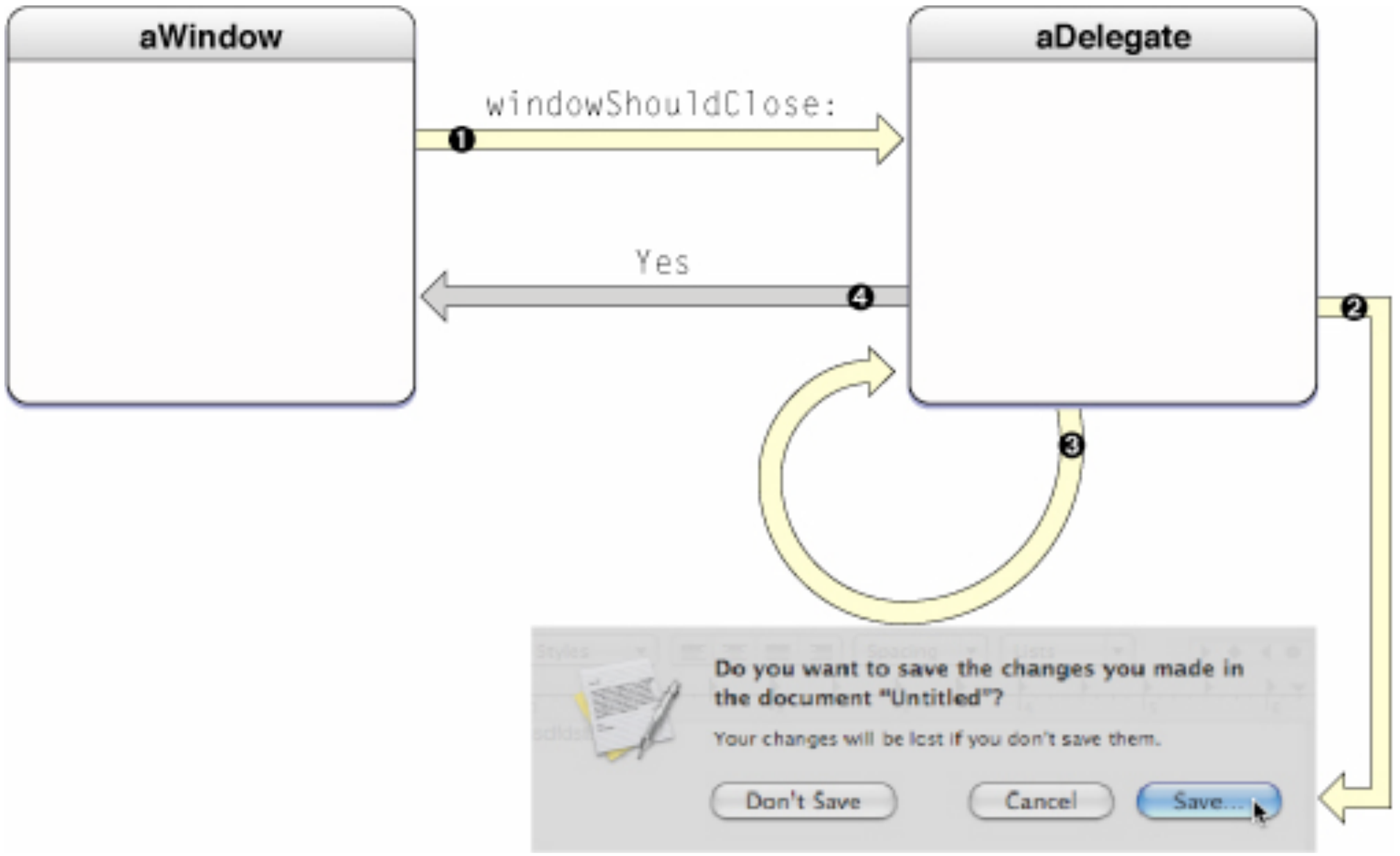
- Les outlets
  - variable d'instance qui référence un autre objet
  - manipulable par l'éditeur d'interface (configurable et archivable)

```
@interface MaClasse : NSObject
{
   IBOutlet NSObject *out;
}
```

- Une classe qui implémente ce pattern possède une propriété nommée delegate



- Un scénario de délégation plus riche

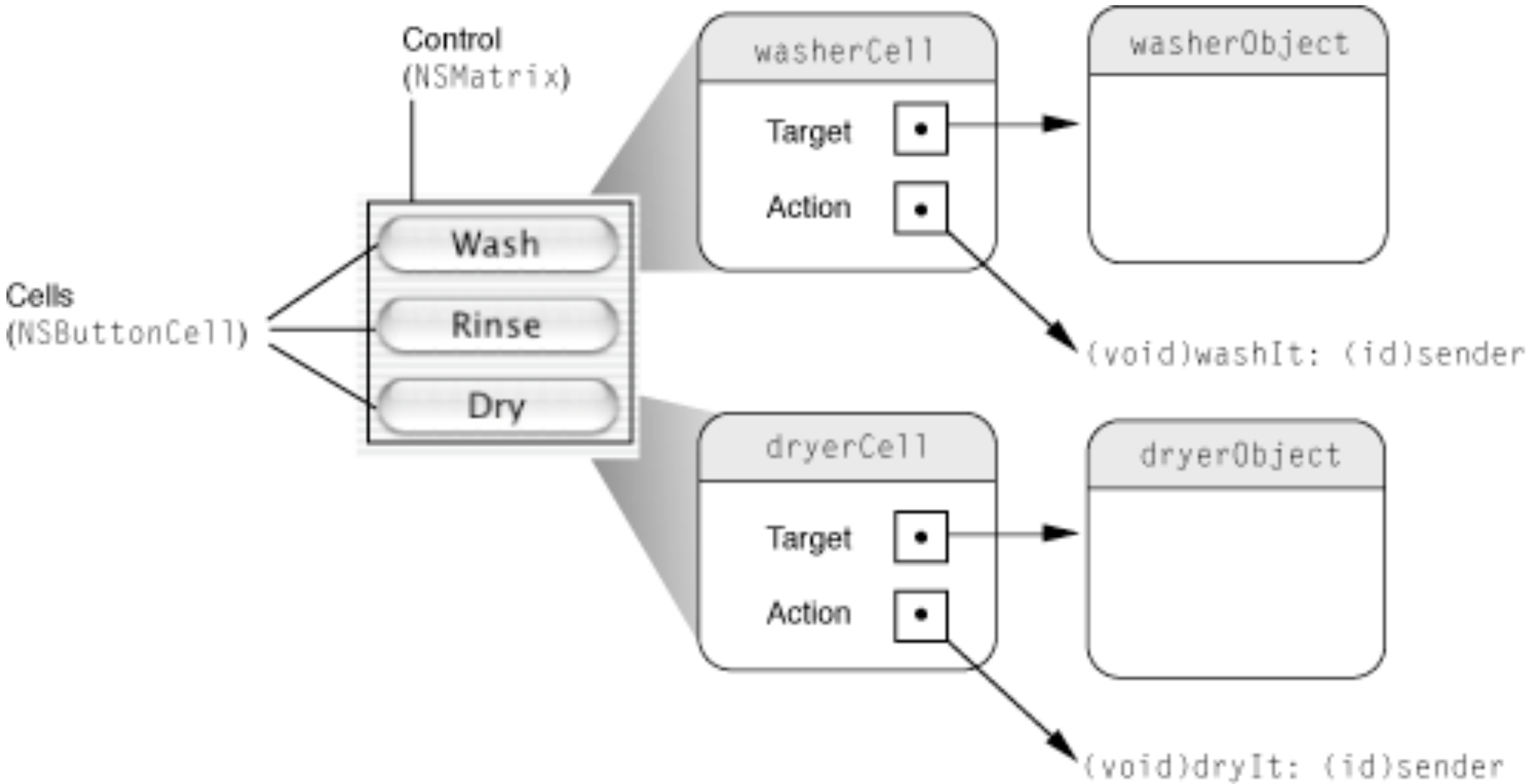


- Il existe une délégation pour les sources de données (data source) et non les interfaces
  - Le grand classique est le UITableView

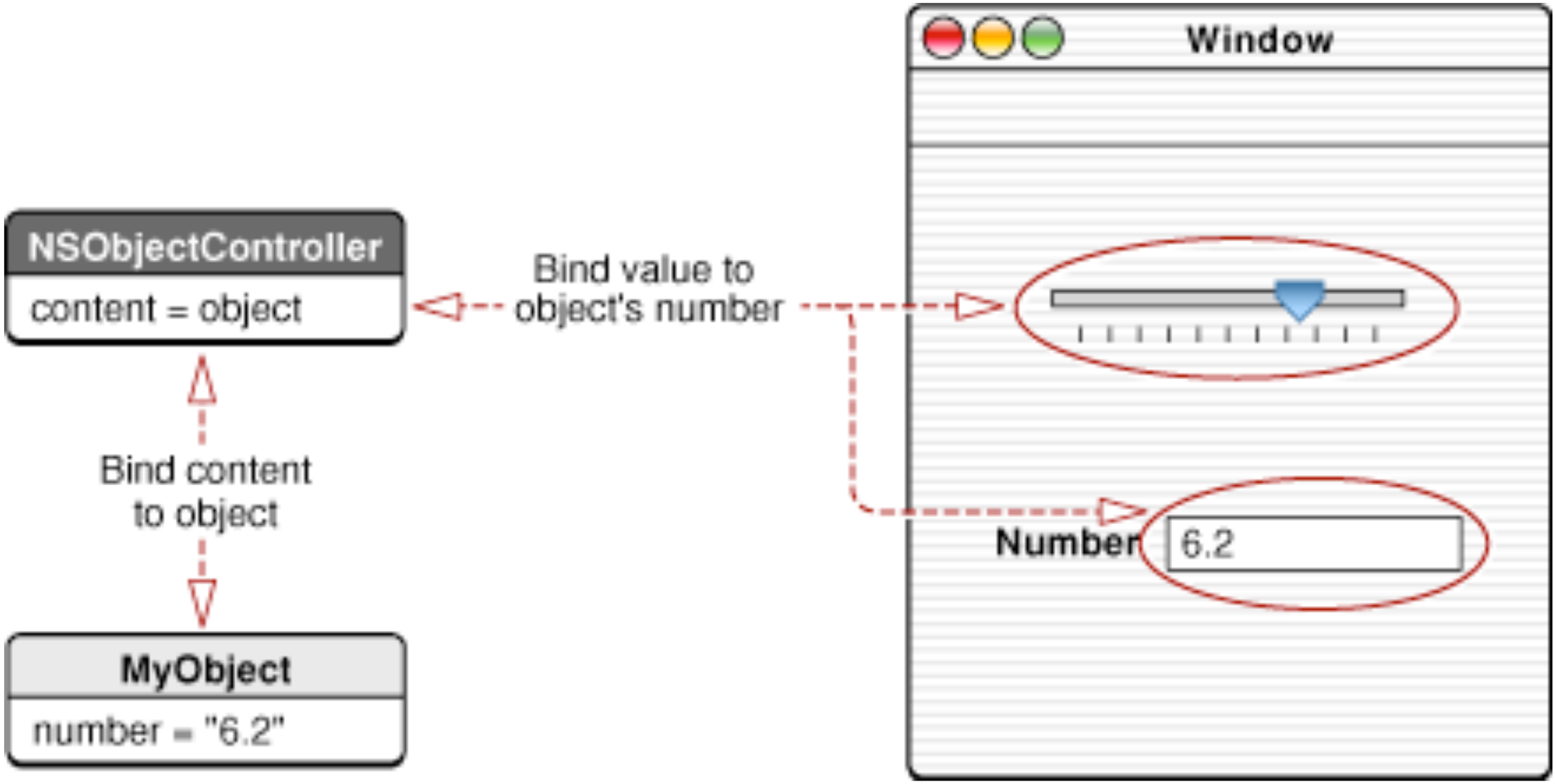
- Le pattern target-action
  - La cible (target) est le receveur d'un message correspondant à une action; il s'agit généralement d'une instance d'une classe personnalisée
  - L'action est le message que le contrôle envoie à la cible
- Une classe éligible pour le pattern doit posséder une méthode signalée comme IBAction

  - (IBAction)faisMoiMal:(id)source

- IBAction n'est pas un type, c'est un tag pour l'éditeur d'interface


Control
(NSMatrix)
Cells
(NSButtonCell)
Wash
Rinse
Dry
washerCell
Target
Action
washerObject
(void)washIt: (id)sender
dryerCell
Target
Action
dryerObject
(void)dryIt: (id)sender

- Les bindings
  - dans le cadre du MVC
    - view : affiche les données
    - model : représente le concept
    - controller : médiateur entre le vue et le modèle
  - reposent sur des objets conforme aux patterns KVO et KVC (suffisant pour les modèles)


NSObjectController
content = object
Bind value to
object's number
Bind content
to object
MyObject
number = "6.2"
Window
Number
6.2

- les points d'entrée et sortie dans les objets sont :
  - `+initialize` initialisation de la classe
  - `-init` initialisation d'un objet
  - `-initWithCoder:` initialisation par désérialisation
  - `-awakeFromNib:` réveil d'un objet lors de sa désérialisation depuis un fichier d'interface
  - `-encodeWithCoder:` sérialisation si nécessaire
  - `-dealloc` contrôle de la destruction de l'objet