What's New with Gestures

Quite a bit, actually

Session 231

Raleigh Ledet
Cocoa Software Engineer

These are confidential sessions—please refrain from streaming, blogging, or taking pictures

What's New with Gestures

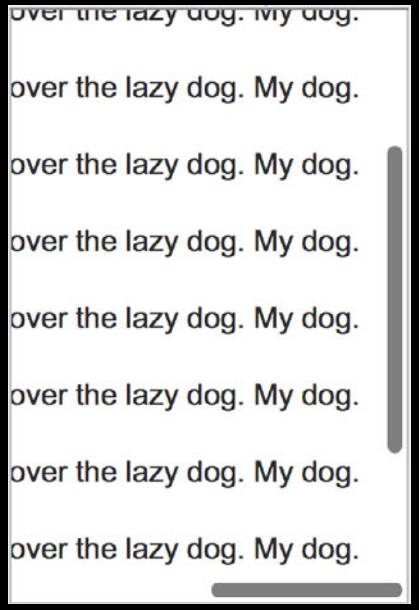
- Scrolling Improvements
- Smart Magnification
- NSScrollView Gesture Adoption
- NSPageController
- Quick Look
- Best Practices

Scrolling Improvements

Scrolling Improvements

Scroller changes

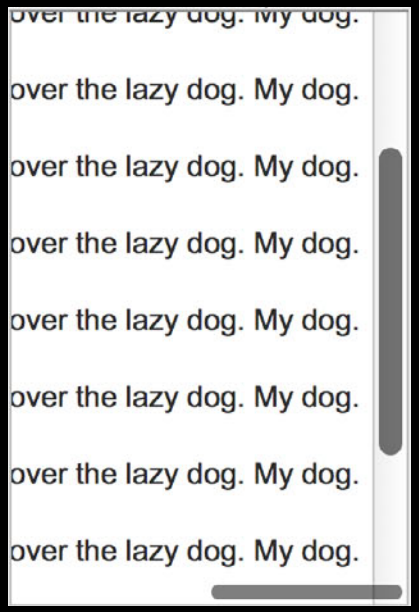
- Expand on rollover



Scrolling Improvements

Scroller changes

- Expand on rollover

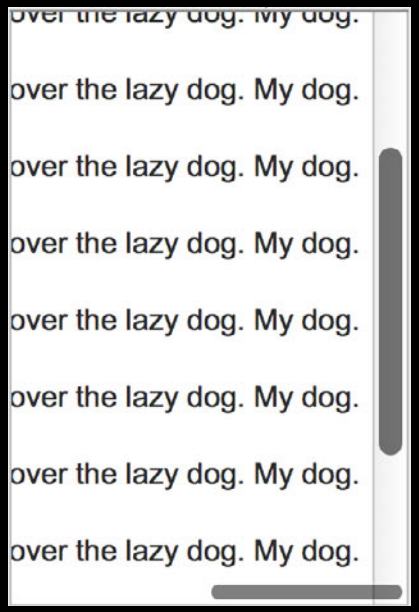


Scrolling Improvements

Scroller changes

- Expand on rollover

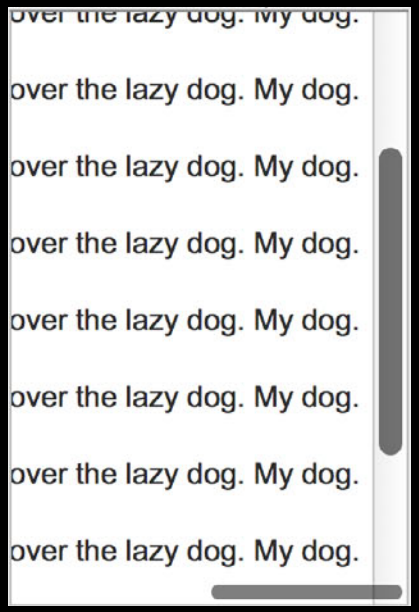
```
+ (CGFloat)scrollerWidthForControlSize:(NSControlSize)controlSize
                          scrollerStyle:(NSScrollerStyle)scrollerStyle
```



Scrolling Improvements

Scroller changes

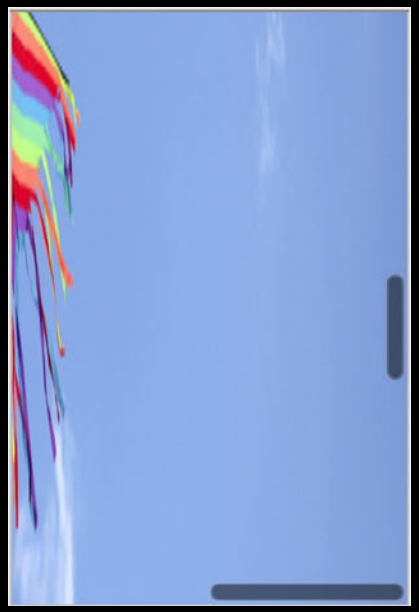
- Expand on rollover
- New look



Scrolling Improvements

Scroller changes

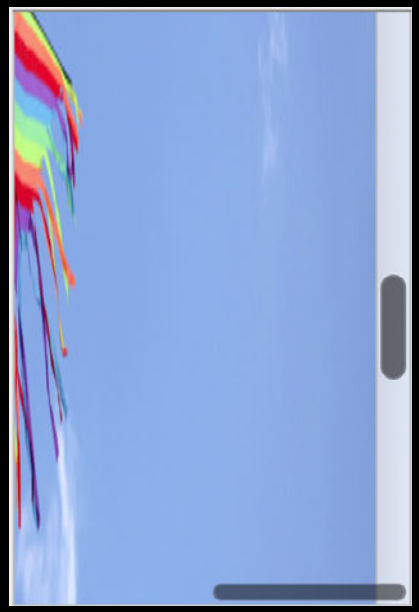
- Expand on rollover
- New look



Scrolling Improvements

Scroller changes

- Expand on rollover
- New look



Scrolling Improvements

Scroller changes

- Expand on rollover
- New look

Scrolling Improvements

Scroller changes

- Expand on rollover
- New look

Scrolling Improvements

Scroller changes

- Expand on rollover
- New look
- Reveal on two-finger touch

Scrolling Improvements

Scroller changes

- Expand on rollover
- New look
- Reveal on two-finger touch

```
NSEventPhaseBegan
NSEventPhaseChanged
NSEventPhaseEnded
NSEventPhaseCancelled
```

Scrolling Improvements

Scroller changes

- Expand on rollover
- New look
- Reveal on two-finger touch

```
NSEventPhaseMayBegin
NSEventPhaseBegan
NSEventPhaseChanged
NSEventPhaseEnded
NSEventPhaseCancelled
```

Scrolling Improvements

Accelerated scrolling

- Nothing for you to do
- `-scrollingDelta` automatically accelerated

Demo

Scrolling improvements

Smart Magnification

Smart Magnification

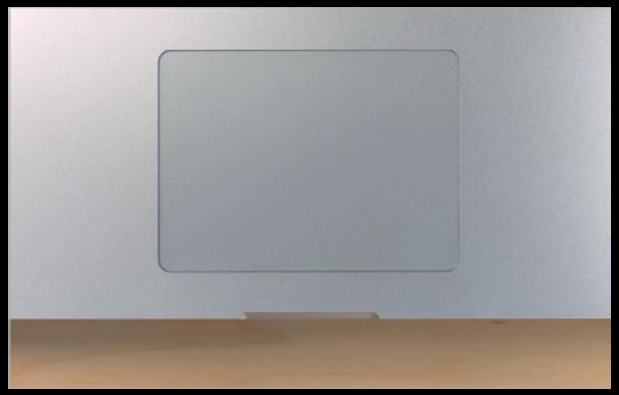
Trigger

- Two-finger double tap

Smart Magnification

Trigger

- Two-finger double tap



Smart Magnification

Responder method

```
- (void)smartMagnifyWithEvent:(NSEvent *)event
```

Smart Magnification

Responder method

```
- (void)smartMagnifyWithEvent:(NSEvent *)event
```

NSEventTypeSmartMagnify

Smart Magnification

Responder method

```
- (void)smartMagnifyWithEvent:(NSEvent *)event
```

NSEventTypeSmartMagnify

-locationInWindow

-modifiers

Smart Magnification

Responder method

```
- (void)smartMagnifyWithEvent:(NSEvent *)event
```

NSEventTypeSmartMagnify

-locationInWindow

-modifiers

NSEventMaskSmartMagnify

Smart Magnification

Responder method

```
- (void)smartMagnifyWithEvent:(NSEvent *)event
```

```
NSEventTypeSmartMagnify*
   -locationInWindow
      -modifiers
NSEventMaskSmartMagnify*
```

*64-bit only

Smart Magnification

Human interface guidelines

- Simple

  Currently at 100%—magnify to 150%

  Currently not 100%—magnify to 100%

Smart Magnification

Human interface guidelines

- Simple

  Currently at 100%—magnify to 150%

  Currently not 100%—magnify to 100%

- Complex

  Magnify such that the width of the target grouping fits inside the scroll view, centering the target point vertically, if possible; magnify to 100% if the mouse hasn't moved and already magnified to the deepest grouping

Smart Magnification

Human interface guidelines

Use NSScrollView

NSScrollView Gesture Adoption

NSScrollView Gesture Adoption

- Magnification
- Smart magnification

Demo

NSScrollView Gesture Adoption

NSScrollView Gesture Adoption

Magnification

```
@property BOOL allowsMagnification
```

NSScrollView Gesture Adoption

Magnification

```
@property BOOL allowsMagnification
@property CGFloat maxMagnification
@property CGFloat minMagnification
```

NSScrollView Gesture Adoption

Magnification

```
@property BOOL allowsMagnification
@property CGFloat maxMagnification
@property CGFloat minMagnification
@property CGFloat magnification
```

NSScrollView Gesture Adoption

Magnification

```
@property BOOL allowsMagnification
@property CGFloat maxMagnification
@property CGFloat minMagnification
@property CGFloat magnification
```

```
[[scrollView animator] setMagnification:value]
```

NSScrollView Gesture Adoption

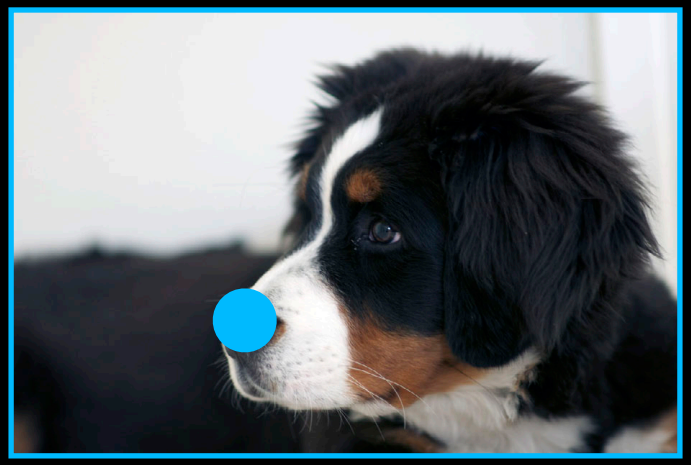
Magnification

```
- (void)setMagnification:(CGFloat)value centeredAtPoint:(NSPoint)point
```

NSScrollView Gesture Adoption

Magnification

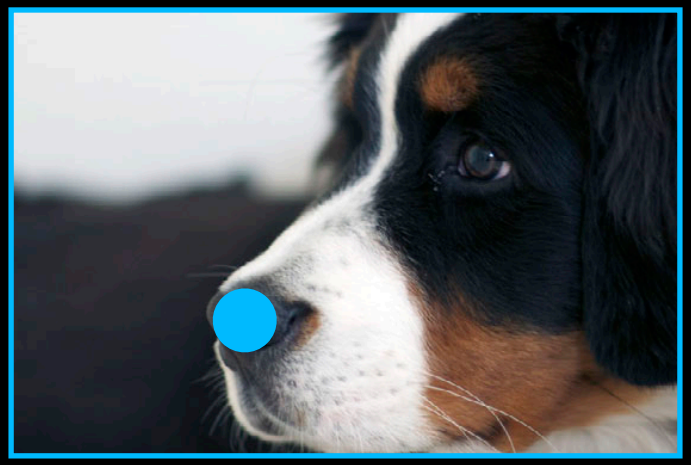
```
- (void)setMagnification:(CGFloat)value centeredAtPoint:(NSPoint)point
```



NSScrollView Gesture Adoption

Magnification

```
- (void)setMagnification:(CGFloat)value centeredAtPoint:(NSPoint)point
```



NSScrollView Gesture Adoption

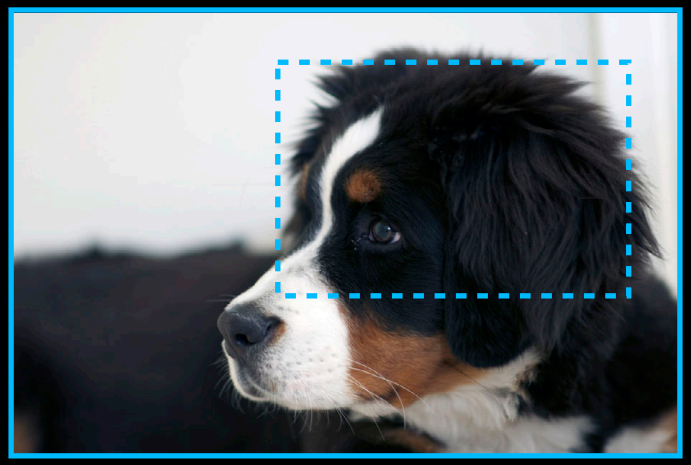
Magnification

```
- (void)setMagnification:(CGFloat)value centeredAtPoint:(NSPoint)point
- (void)magnifyToFitRect:(NSRect)rect
```

NSScrollView Gesture Adoption

Magnification

```
- (void)setMagnification:(CGFloat)value centeredAtPoint:(NSPoint)point
- (void)magnifyToFitRect:(NSRect)rect
```



NSScrollView Gesture Adoption

Magnification

```
- (void)setMagnification:(CGFloat)value centeredAtPoint:(NSPoint)point
- (void)magnifyToFitRect:(NSRect)rect
```

NSScrollView Gesture Adoption

Magnification

```
- (void)setMagnification:(CGFloat)value centeredAtPoint:(NSPoint)point
- (void)magnifyToFitRect:(NSRect)rect
```

NSScrollView Gesture Adoption

Magnification

```
- (void)setMagnification:(CGFloat)value centeredAtPoint:(NSPoint)point
- (void)magnifyToFitRect:(NSRect)rect
```

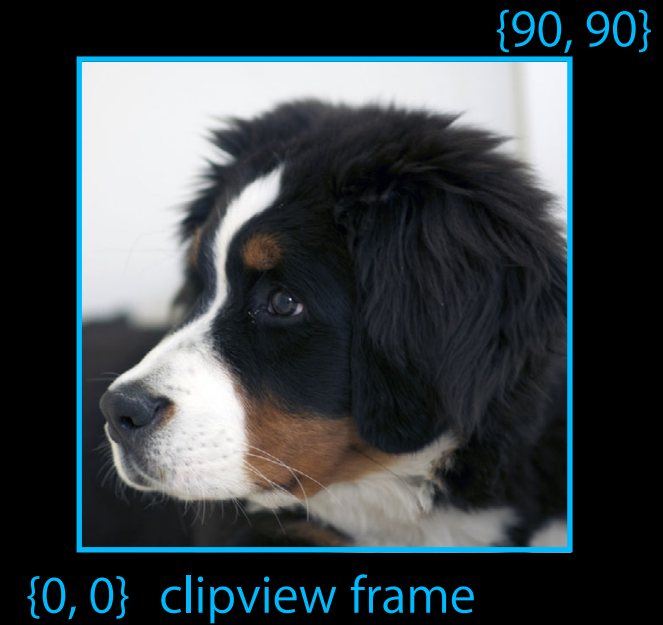
```
[[scrollView animator] ...]
```

NSScrollView Gesture Adoption

Magnification

NSScrollView Gesture Adoption

Magnification



NSScrollView Gesture Adoption

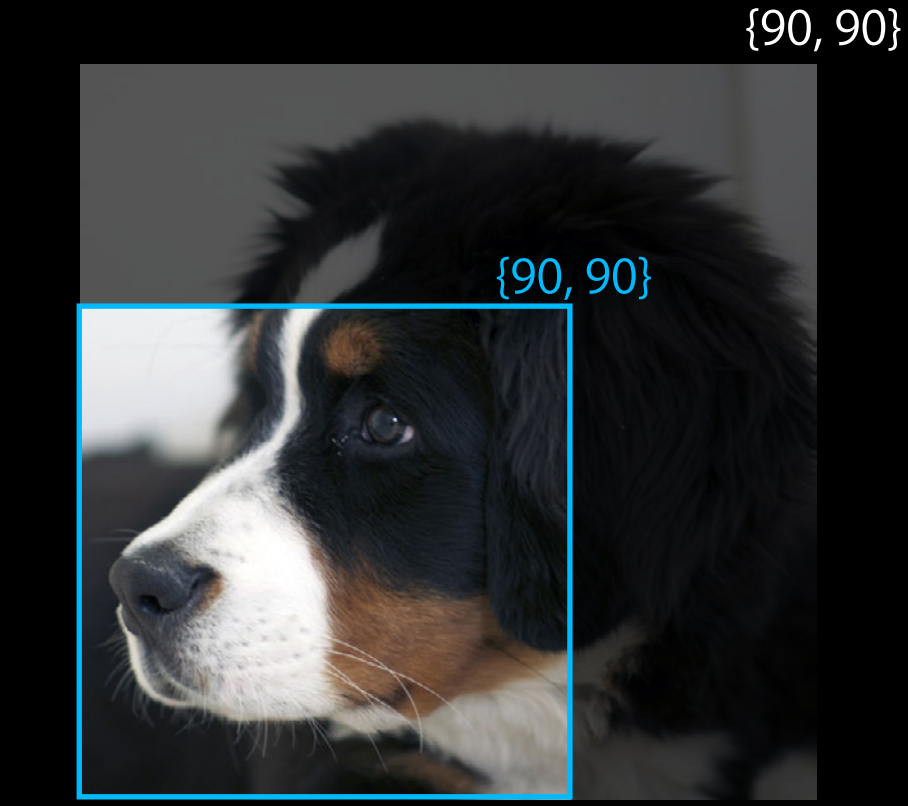
Magnification

{90, 90}
{90, 90}

{0, 0} clipview frame
{0, 0} document frame

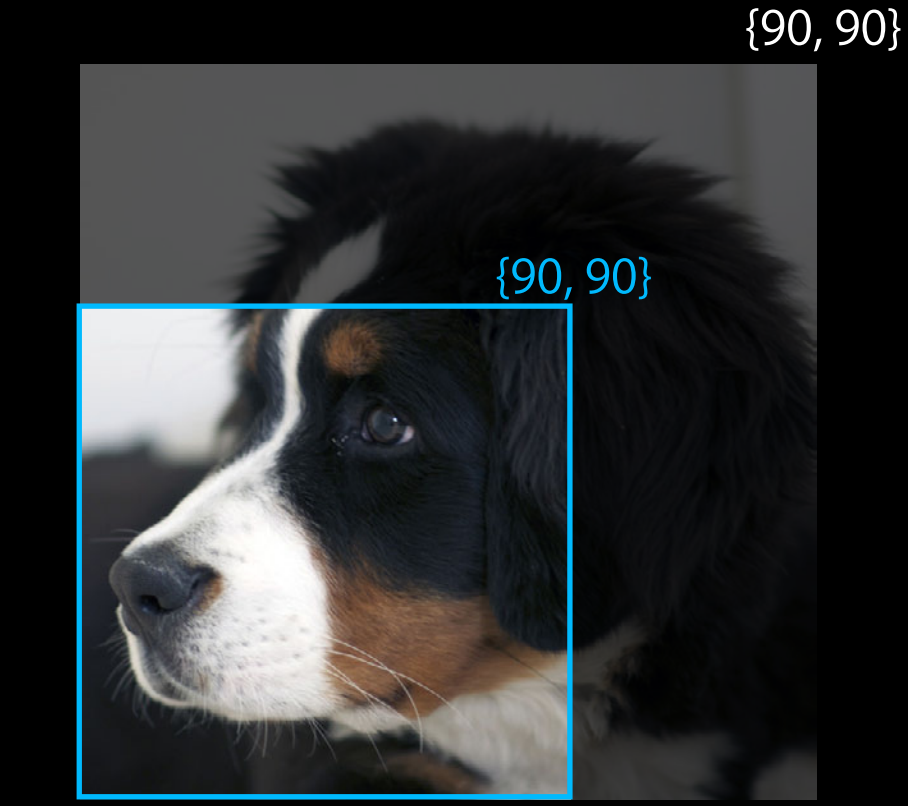
NSScrollView Gesture Adoption

Magnification



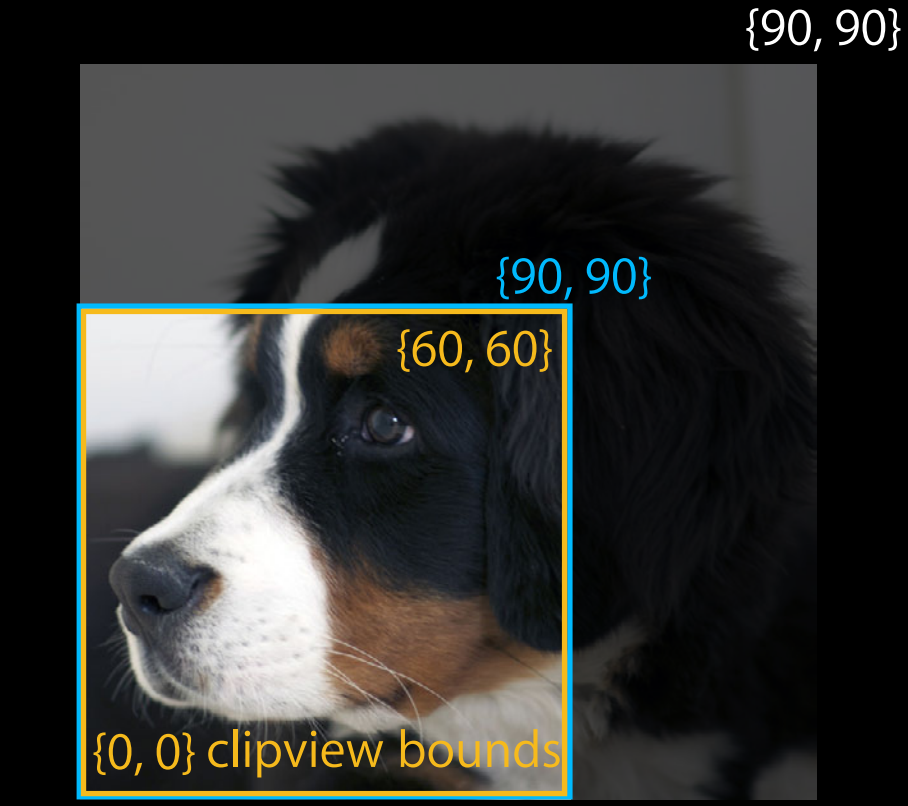
NSScrollView Gesture Adoption

Magnification



NSScrollView Gesture Adoption

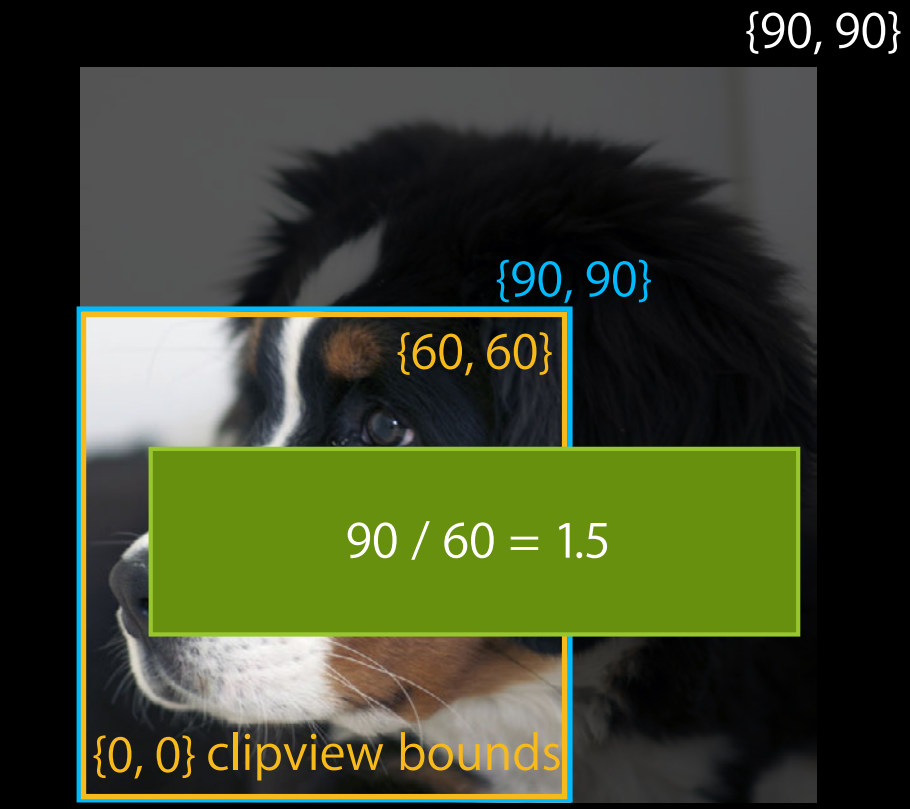
Magnification



{0, 0} clipview frame
{0, 0} document frame

NSScrollView Gesture Adoption

Magnification



{0, 0} clipview frame
{0, 0} document frame

NSScrollView Gesture Adoption

Smart magnification

- Let NSScrollView handle the HI guidelines

NSScrollView Gesture Adoption

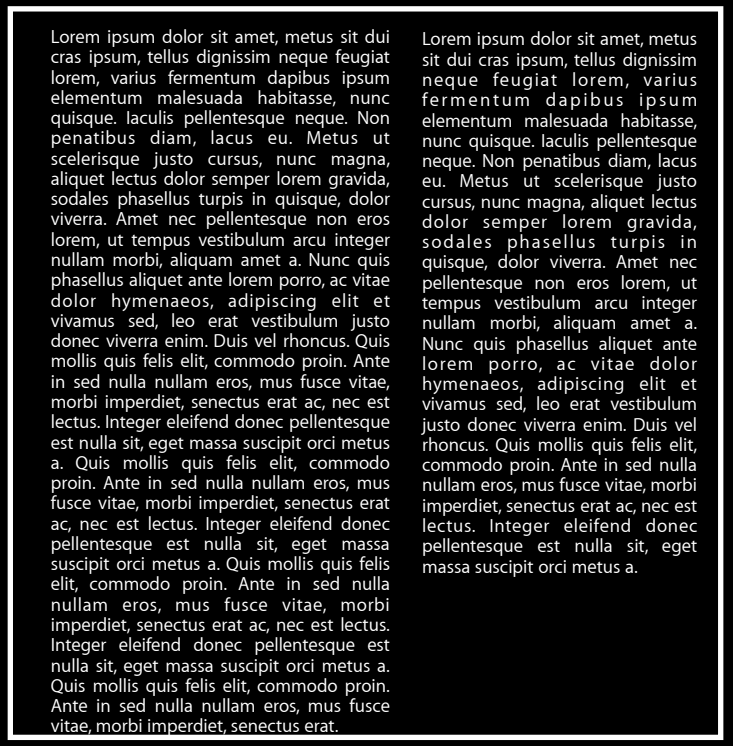
Smart magnification

- Let NSScrollView handle the HI guidelines
- Implement the following method in your document view

```
- (NSRect)rectForSmartMagnificationAtPoint:(NSPoint)location
                                  inRect:(NSRect)visibleRect;
```

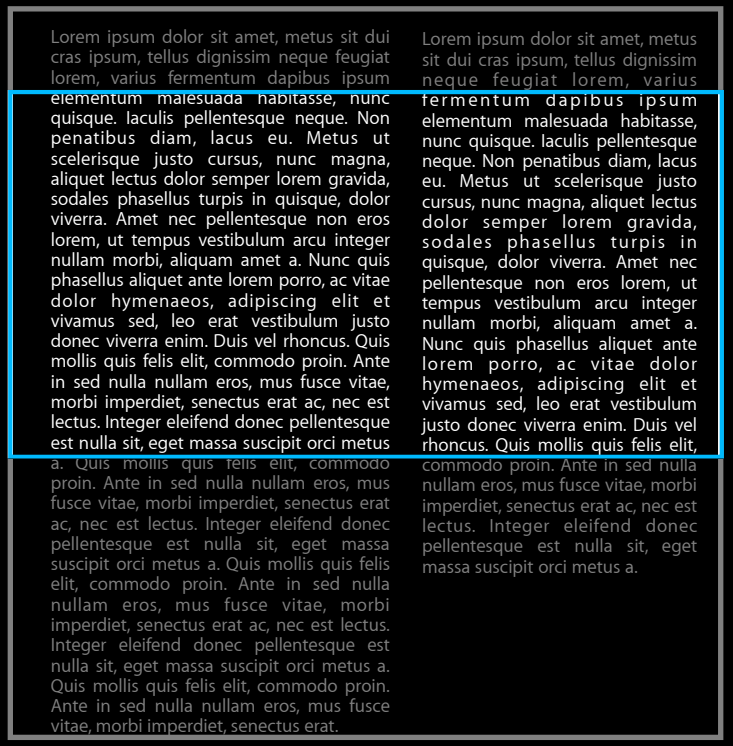
NSScrollView Gesture Adoption

Smart magnification



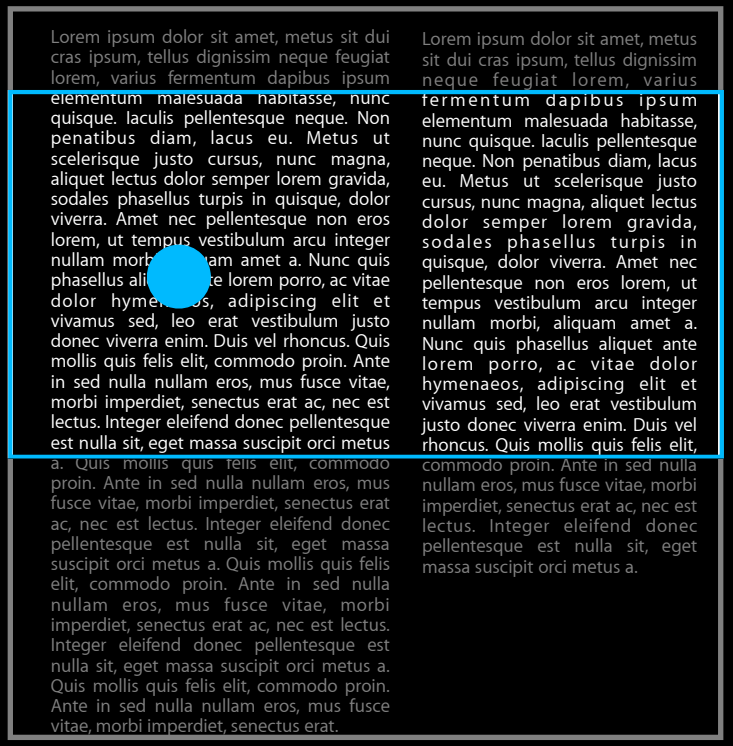
NSScrollView Gesture Adoption

Smart magnification



NSScrollView Gesture Adoption

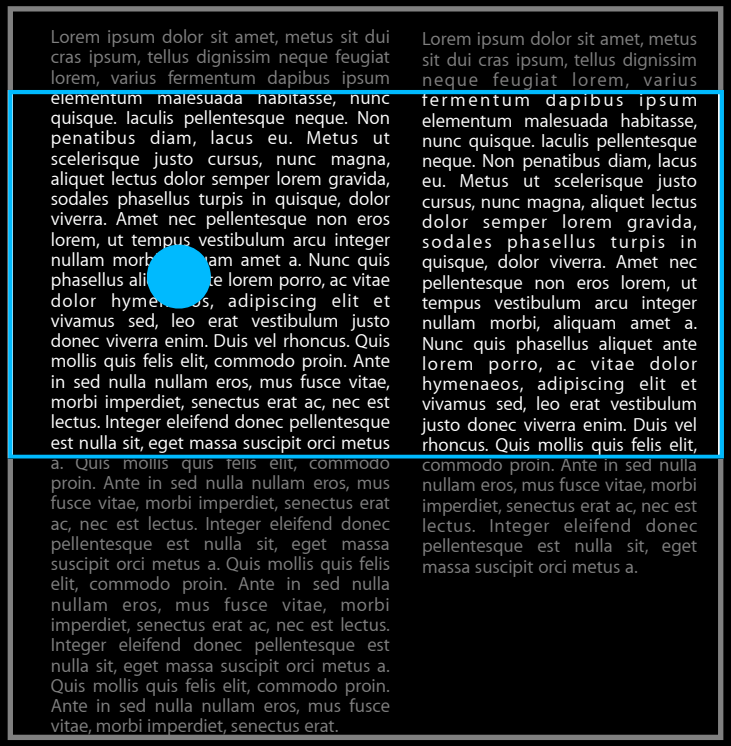
Smart magnification



NSScrollView Gesture Adoption

Smart magnification

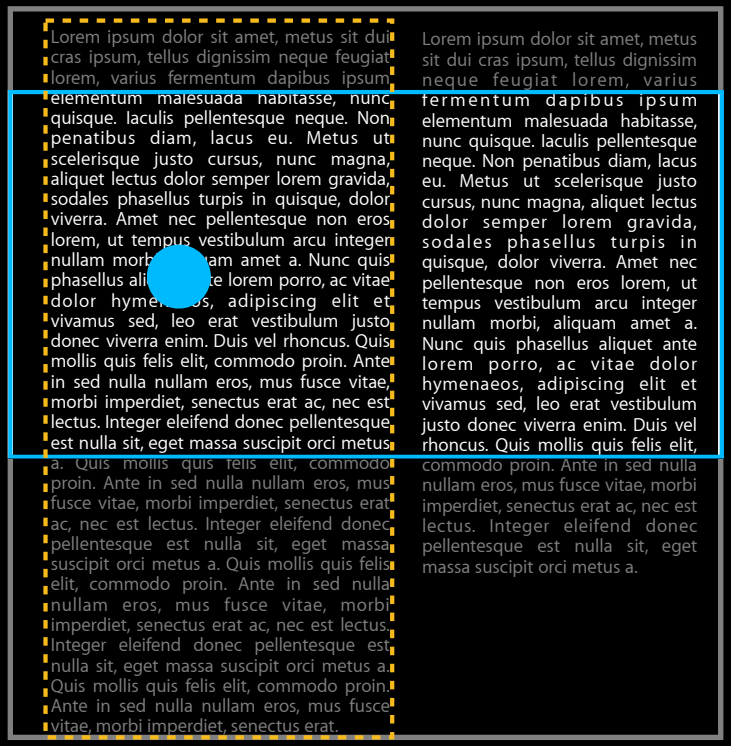
```
- (NSRect)rectForSmartMagnificationAtPoint:(NSPoint)location
                                  inRect:(NSRect)visibleRect;
```



NSScrollView Gesture Adoption

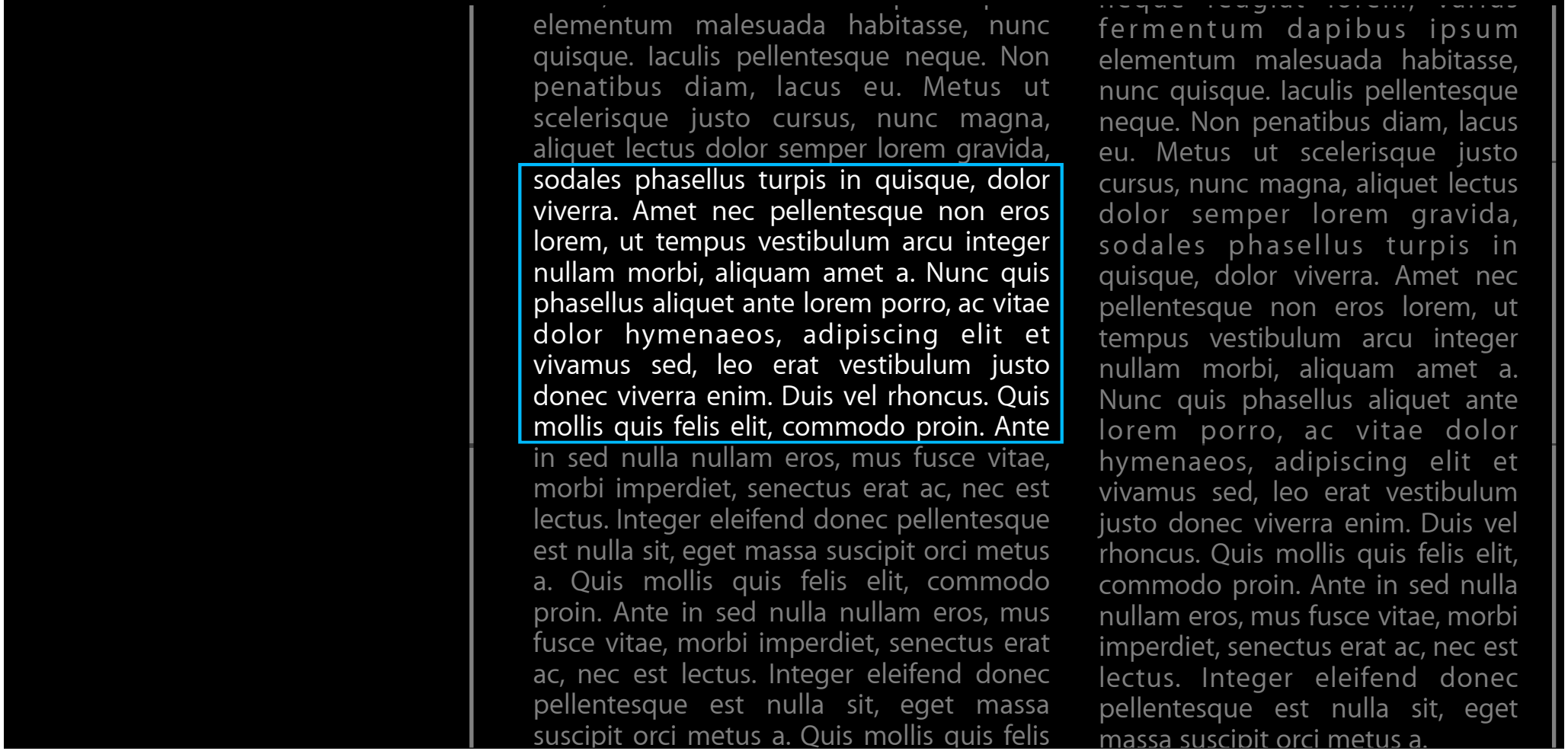
Smart magnification

```
- (NSRect)rectForSmartMagnificationAtPoint:(NSPoint)location
                                  inRect:(NSRect)visibleRect;
```



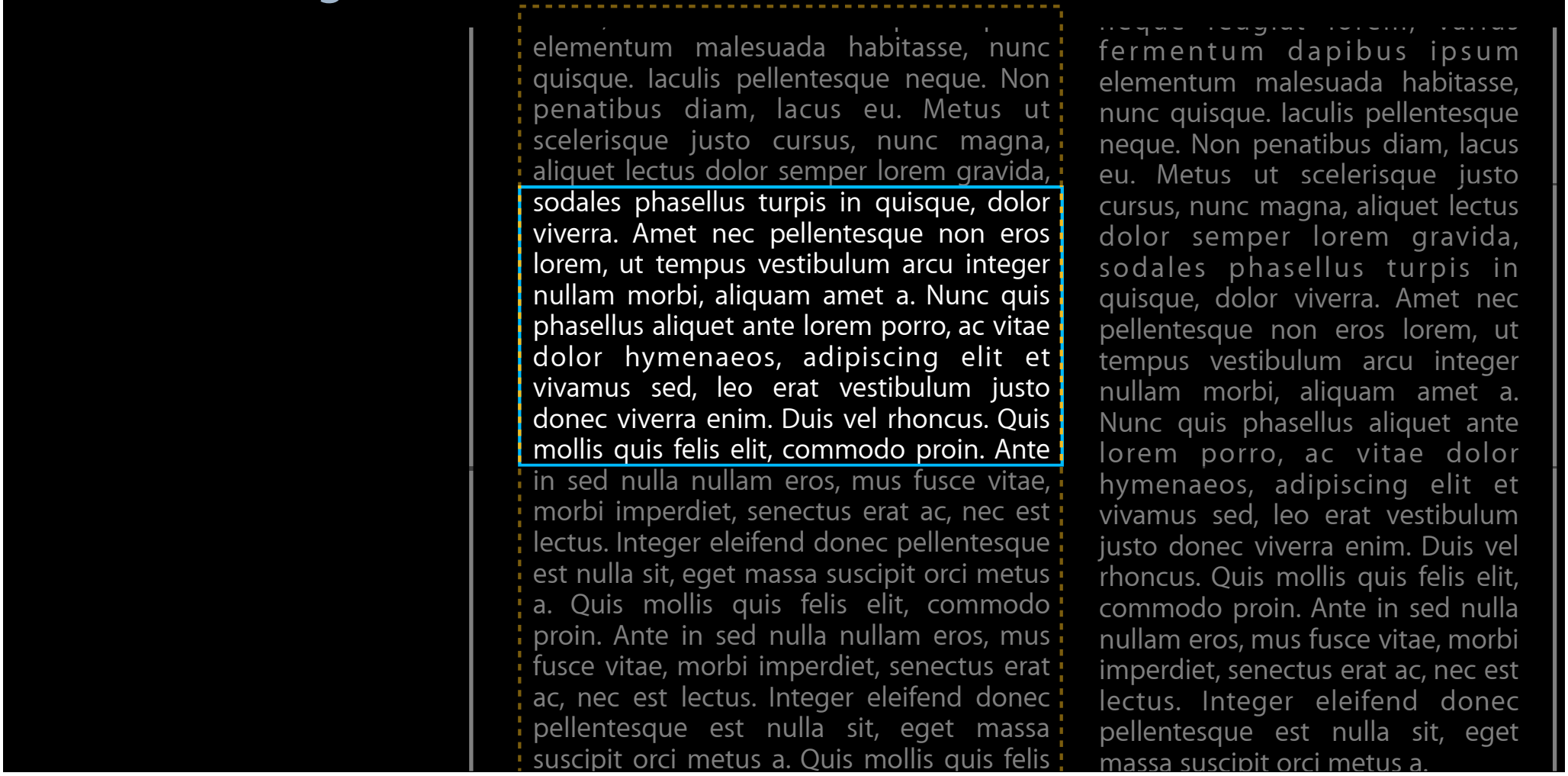
NSScrollView Gesture Adoption

Smart magnification



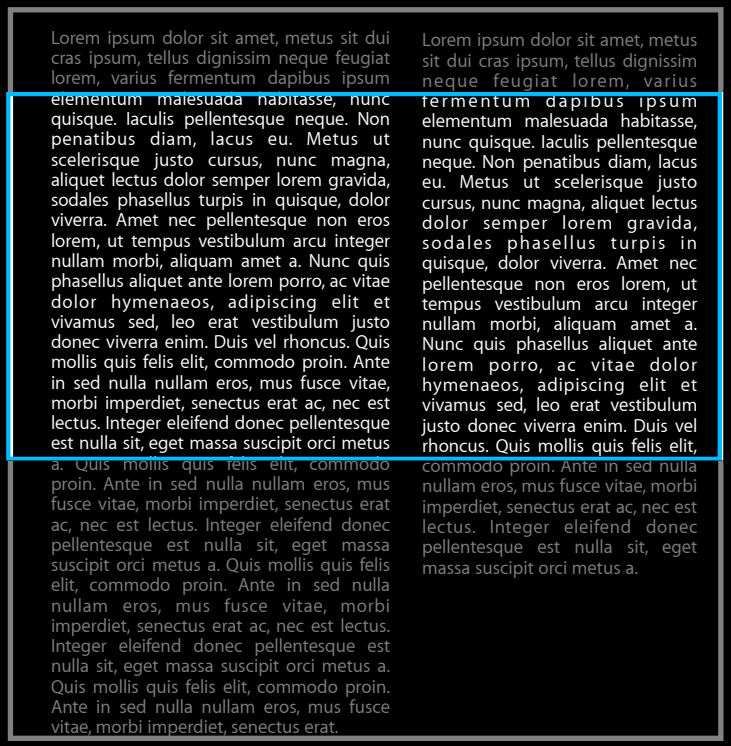
NSScrollView Gesture Adoption

Smart magnification



NSScrollView Gesture Adoption

Smart magnification



NSScrollView Gesture Adoption

Notifications

- NSScrollViewWillStartLiveMagnifyNotification
- NSScrollViewDidEndLiveMagnifyNotification

NSPageController

Swiping made easy

Demo

Old vs. new

NSPageController

- History mode
- Custom mode

NSPageController

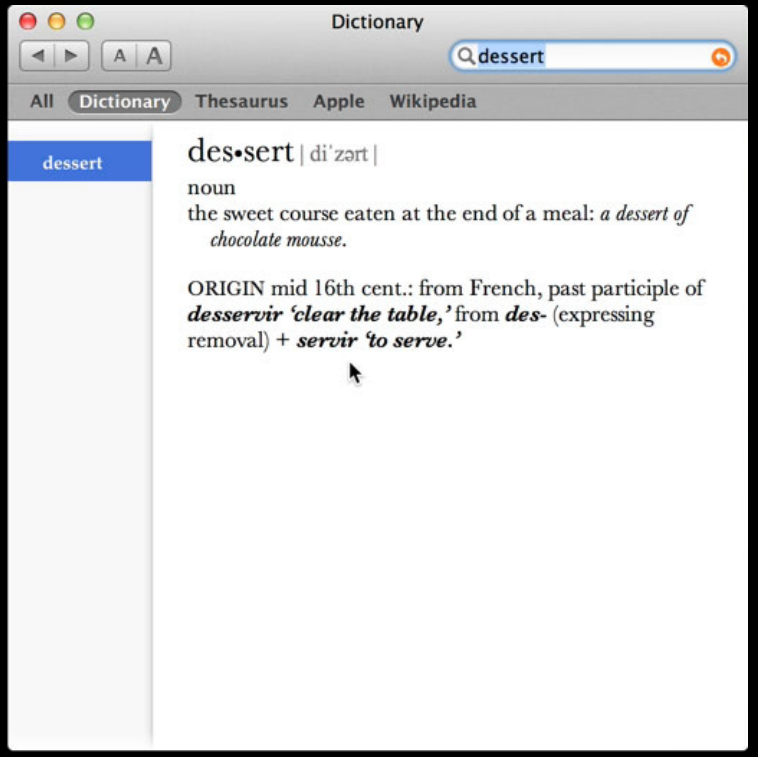
- History mode
- Custom mode

NSPageController

- History mode
- Custom mode

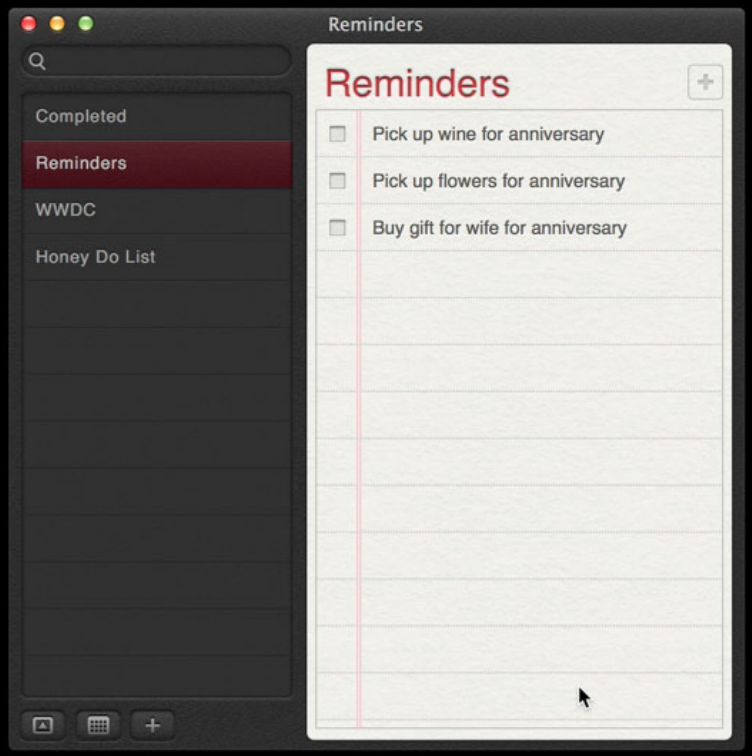
NSPageController

- History mode
- Custom mode



NSPageController

- History mode
- Custom mode



NSPageController

NSViewController

NSPageController

NSPageController

NSPageController

NSPageController

Automatically inserted into responder chain

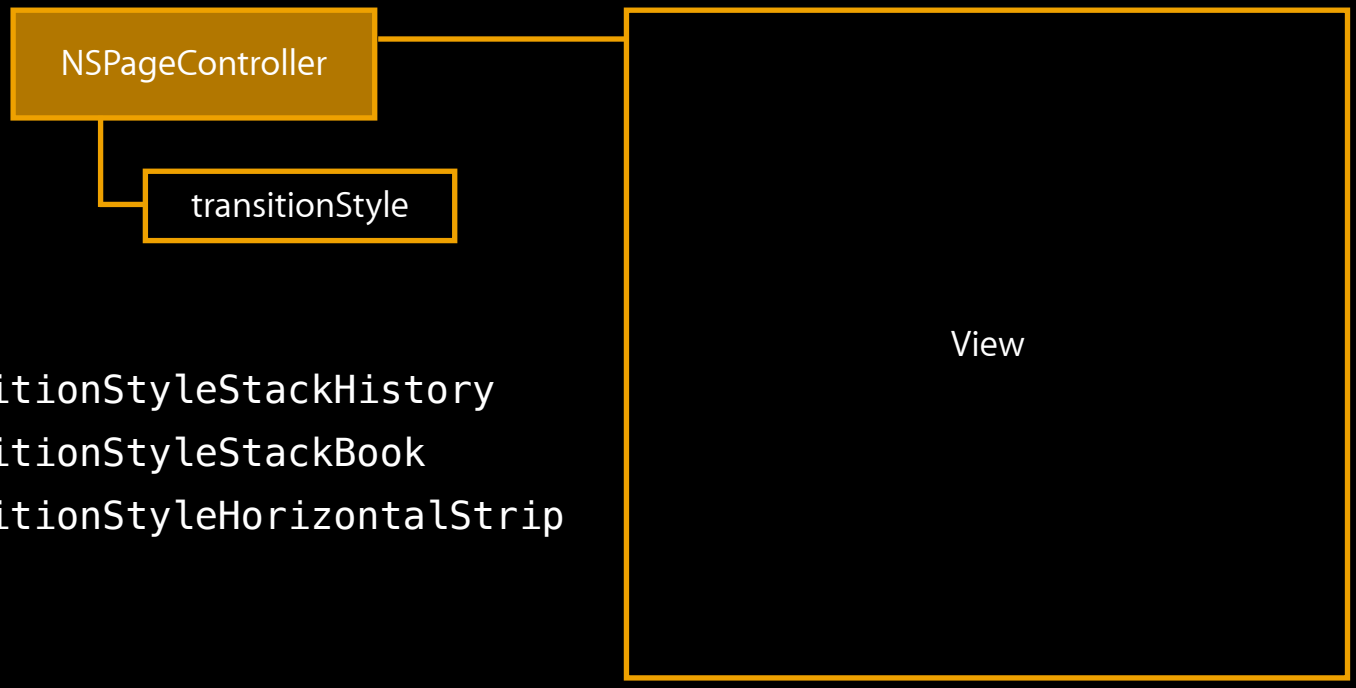
NSPageController

NSPageController

NSPageController

```
NSPageControllerTransitionStyleStackHistory
NSPageControllerTransitionStyleStackBook
NSPageControllerTransitionStyleHorizontalStrip
```

NSPageController




```
NSPageControllerTransitionStyleStackHistory
NSPageControllerTransitionStyleStackBook
NSPageControllerTransitionStyleHorizontalStrip
```

NSPageController

```
NSPageControllerTransitionStyleStackHistory
NSPageControllerTransitionStyleStackBook
NSPageControllerTransitionStyleHorizontalStrip
```

NSPageController

```
NSPageControllerTransitionStyleStackHistory
NSPageControllerTransitionStyleStackBook
NSPageControllerTransitionStyleHorizontalStrip
```

NSPageController

```
NSPageControllerTransitionStyleStackHistory
NSPageControllerTransitionStyleStackBook
NSPageControllerTransitionStyleHorizontalStrip
```

NSPageController

```
NSPageControllerTransitionStyleStackHistory
NSPageControllerTransitionStyleStackBook
NSPageControllerTransitionStyleHorizontalStrip
```

NSPageController

```
NSPageControllerTransitionStyleStackHistory
NSPageControllerTransitionStyleStackBook
NSPageControllerTransitionStyleHorizontalStrip
```

NSPageController

```
NSPageControllerTransitionStyleStackHistory
NSPageControllerTransitionStyleStackBook
NSPageControllerTransitionStyleHorizontalStrip
```

NSPageController

```
NSPageControllerTransitionStyleStackHistory
NSPageControllerTransitionStyleStackBook
NSPageControllerTransitionStyleHorizontalStrip
```

NSPageController

```
NSPageControllerTransitionStyleStackHistory
NSPageControllerTransitionStyleStackBook
NSPageControllerTransitionStyleHorizontalStrip
```

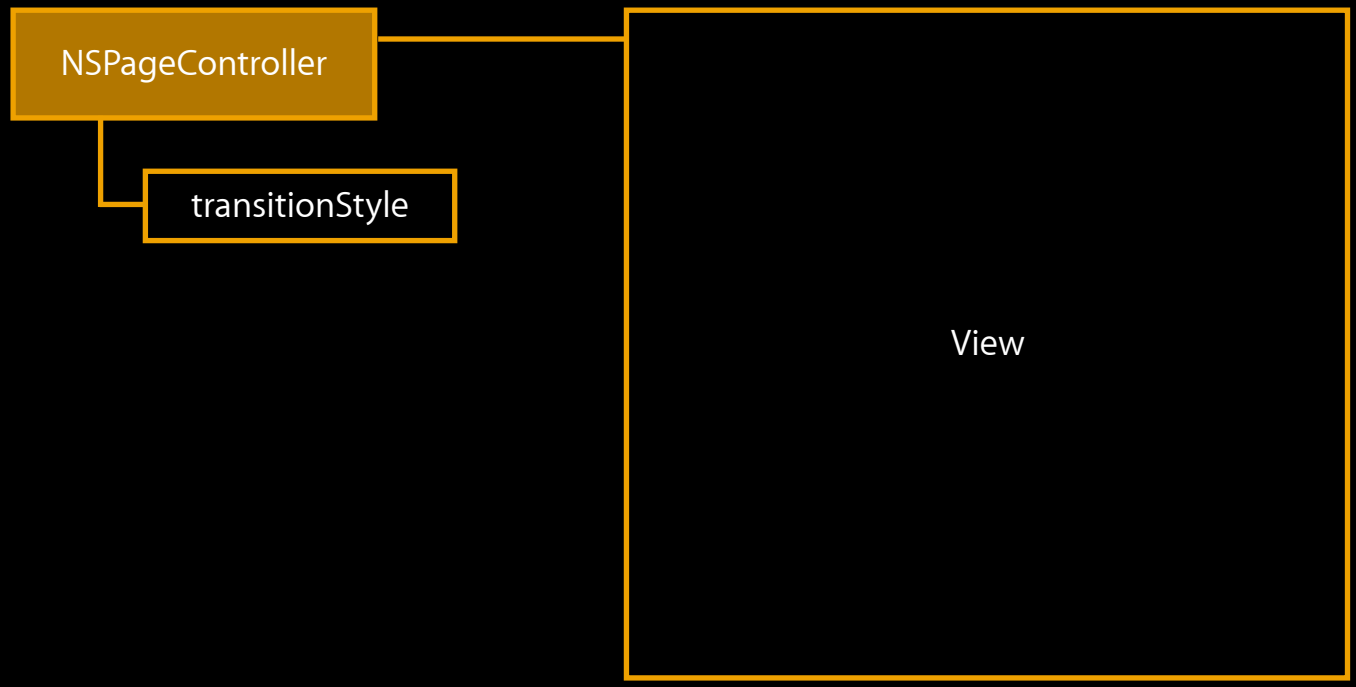
NSPageController

```
NSPageControllerTransitionStyleStackHistory
NSPageControllerTransitionStyleStackBook
NSPageControllerTransitionStyleHorizontalStrip
```

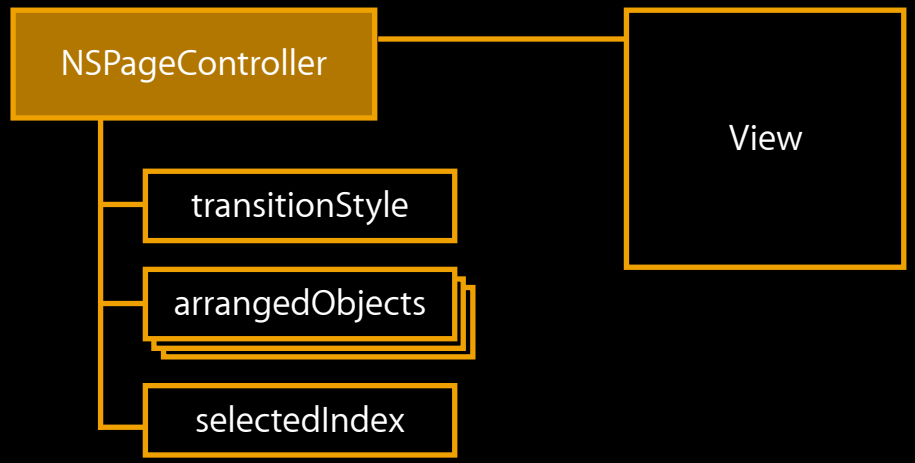
NSPageController

```
NSPageControllerTransitionStyleStackHistory
NSPageControllerTransitionStyleStackBook
NSPageControllerTransitionStyleHorizontalStrip
```

NSPageController



NSPageController



NSPageController

IBAction Methods

```
- (IBAction)navigateBack:(id)sender;
- (IBAction)navigateForward:(id)sender;
- (IBAction)takeSelectedIndexFrom:(id)sender;
```

NSPageController

Programmatically changing index

```
[pageController setSelectedIndex:newIndex];
```

Programmatically changing index with animation

```
[[pageController animator] setSelectedIndex:newIndex];
```

NSPageController

Programmatically changing index

```
[pageController setSelectedIndex:newIndex];
```

Programmatically changing index with Animation

```
[NSAnimationContext runAnimationGroup:^(NSAnimationContext *context) {
    [[pageController animator] setSelectedIndex:newIndex];
} completionHandler:^{
    [pageController completeTransition];
}];
```

NSPageController

History mode

- Manage history
- Responsive
- Simple to implement

NSPageController

History mode

NSPageController

History mode

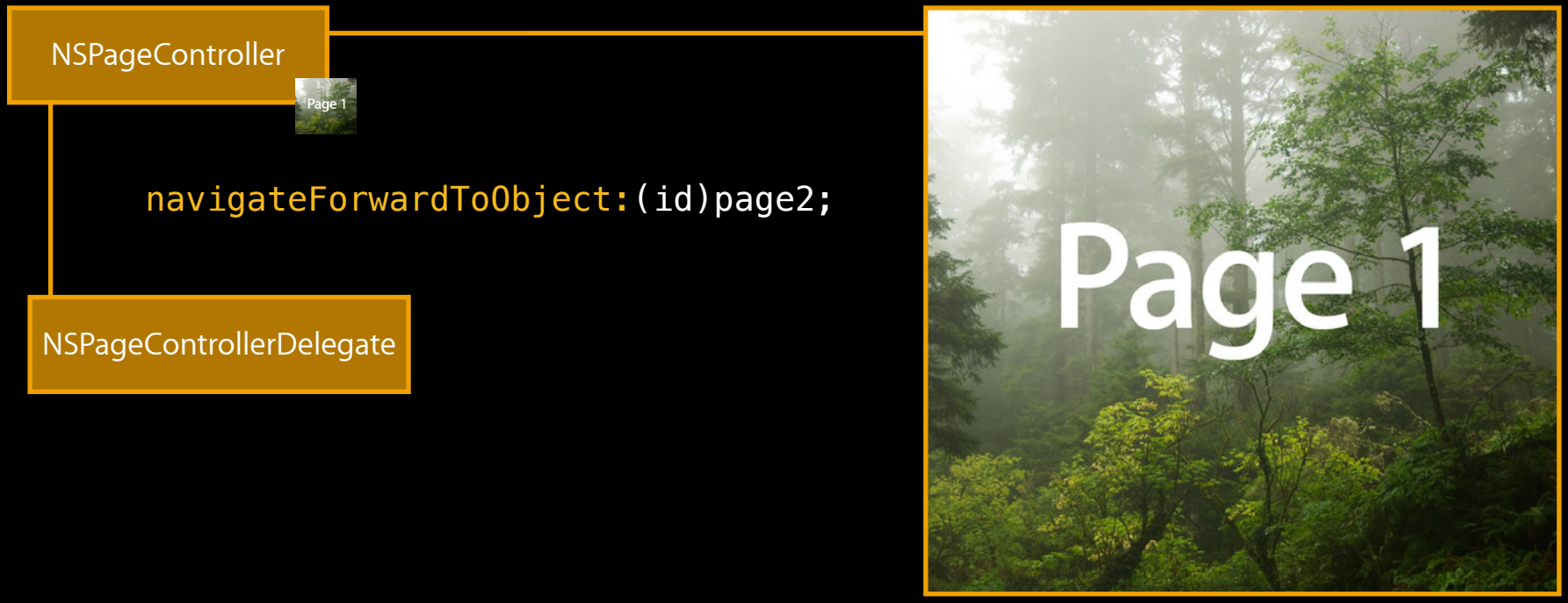
NSPageController

```
navigateForwardToObject:(id)page2;
```

NSPageControllerDelegate

NSPageController

History mode



NSPageController

History mode

NSPageController

History mode

```
pageControllerWillStartLiveTransition:
```

NSPageController

History mode

NSPageController

History mode

NSPageController

History mode

NSPageController

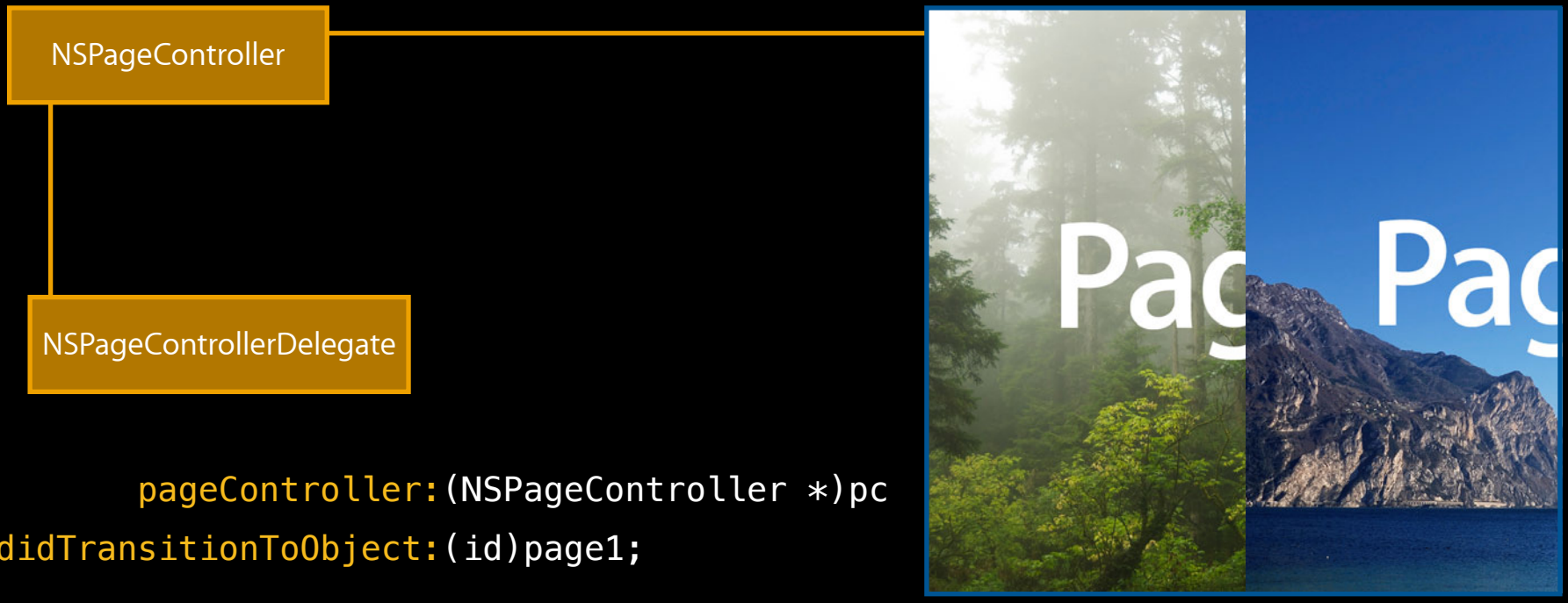
History mode

NSPageController

History mode

NSPageController

History mode




```
pageController:(NSPageController *)pc
didTransitionToObject:(id)page1;
```

NSPageController

History mode

NSPageController

History mode

NSPageController

History mode

```
pageControllerDidEndLiveTransition:
```

NSPageController

History mode

NSPageController

History mode

NSPageController

Custom mode

- Responsive
- More control

NSPageController

Custom mode

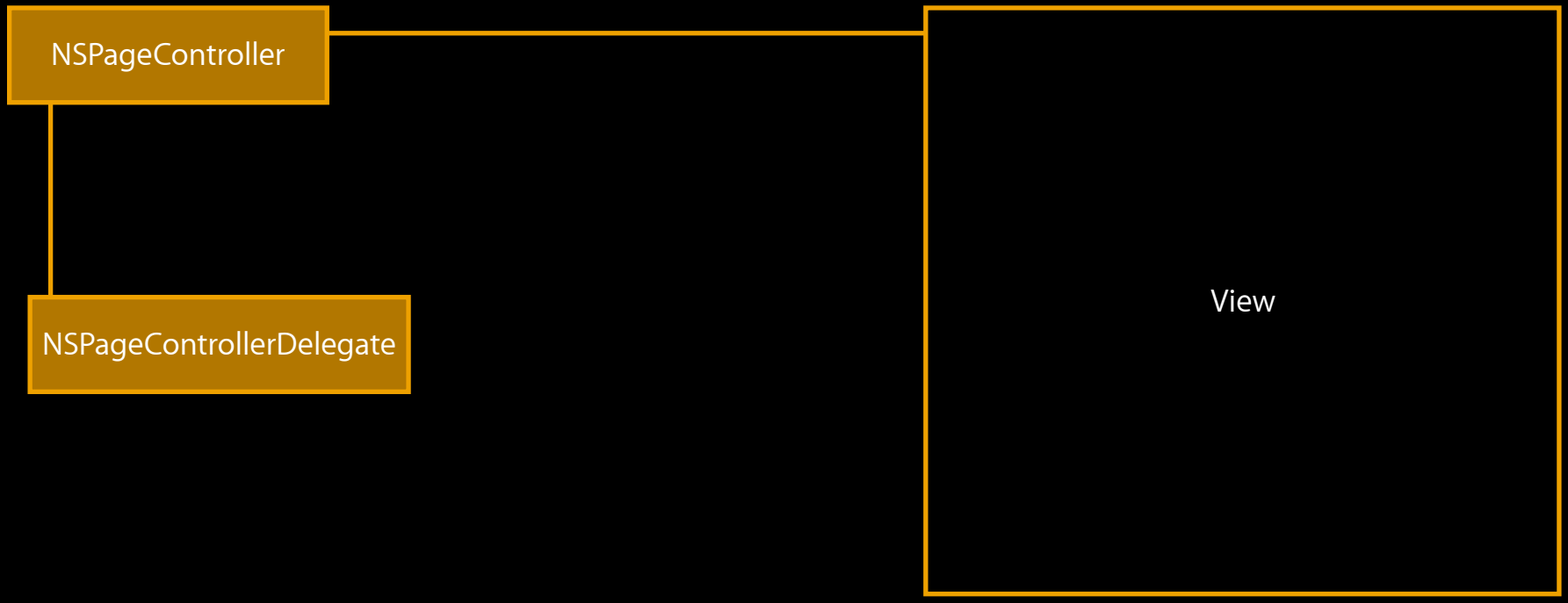
- Responsive
- More control

```
-(NSString *)pageController:(NSPageController *)pc
        identifierForObject:(id)object;

-(NSViewController *)pageController:(NSPageController *)pc
        viewControllerForIdentifier:(NSString *)identifier
```

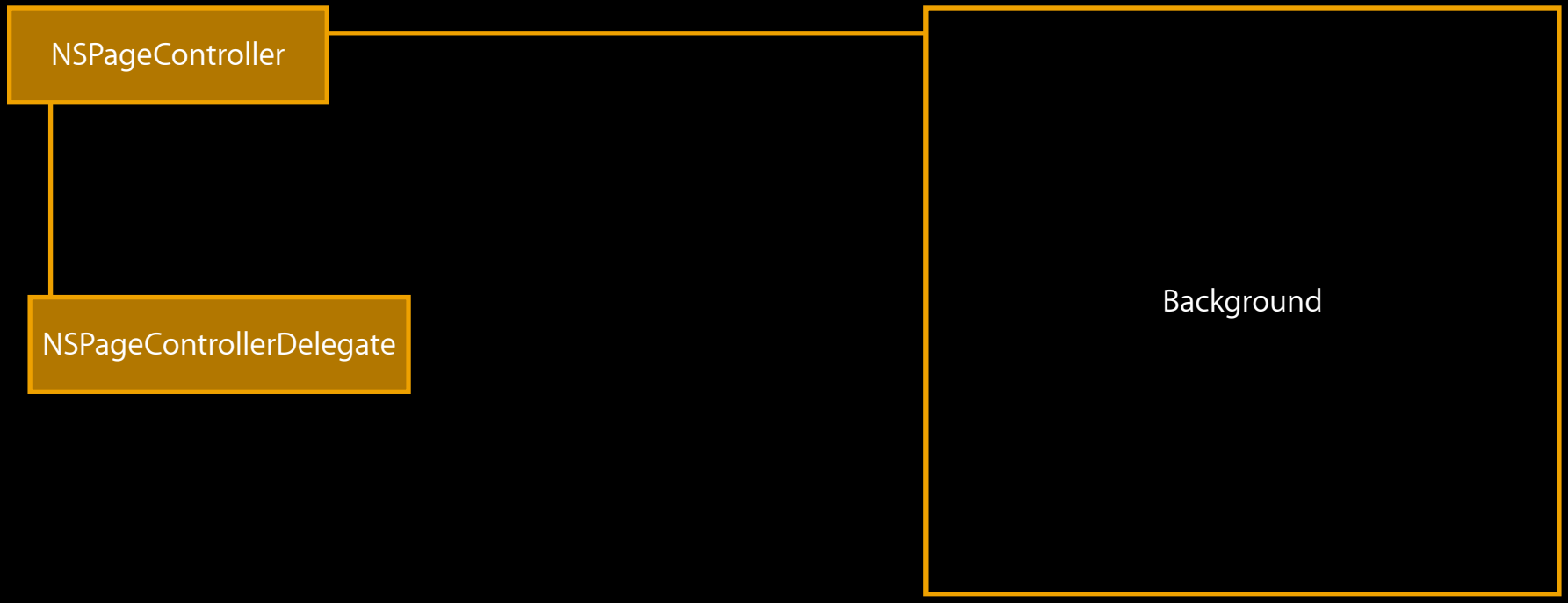
NSPageController

Custom mode



NSPageController

Custom mode



NSPageController

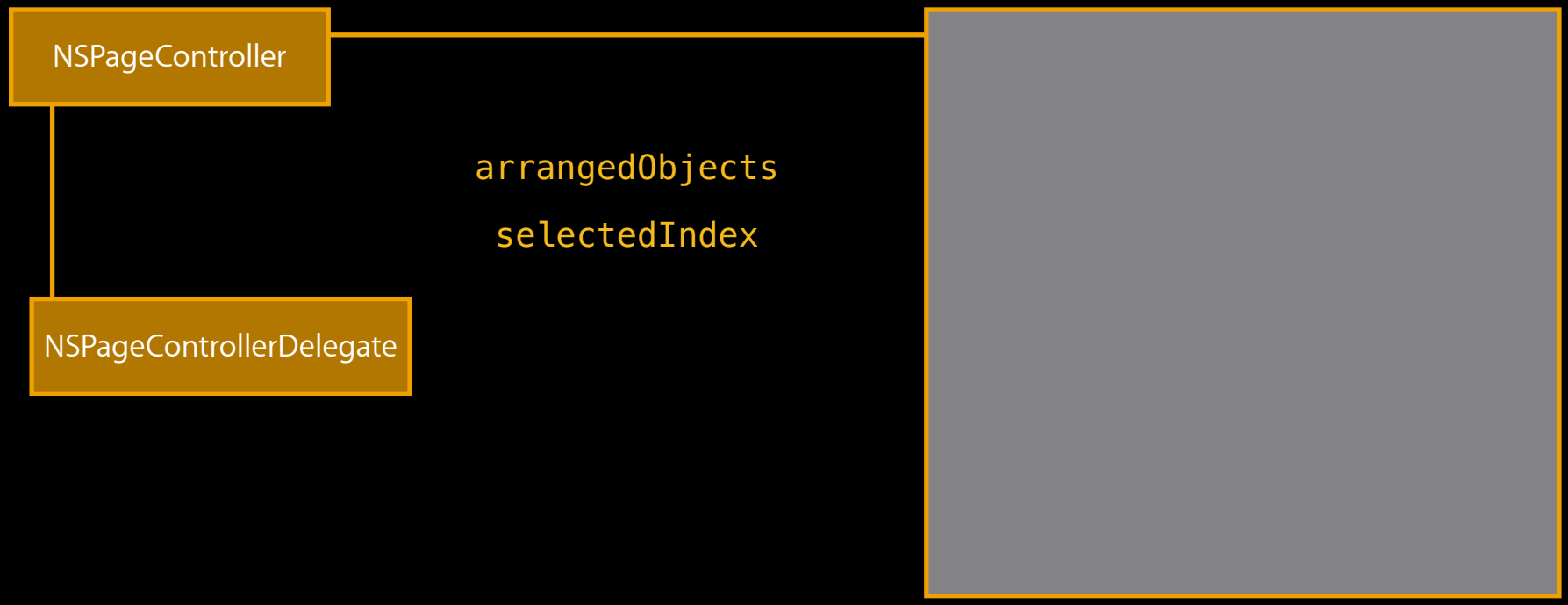
Custom mode

NSPageController

Custom mode

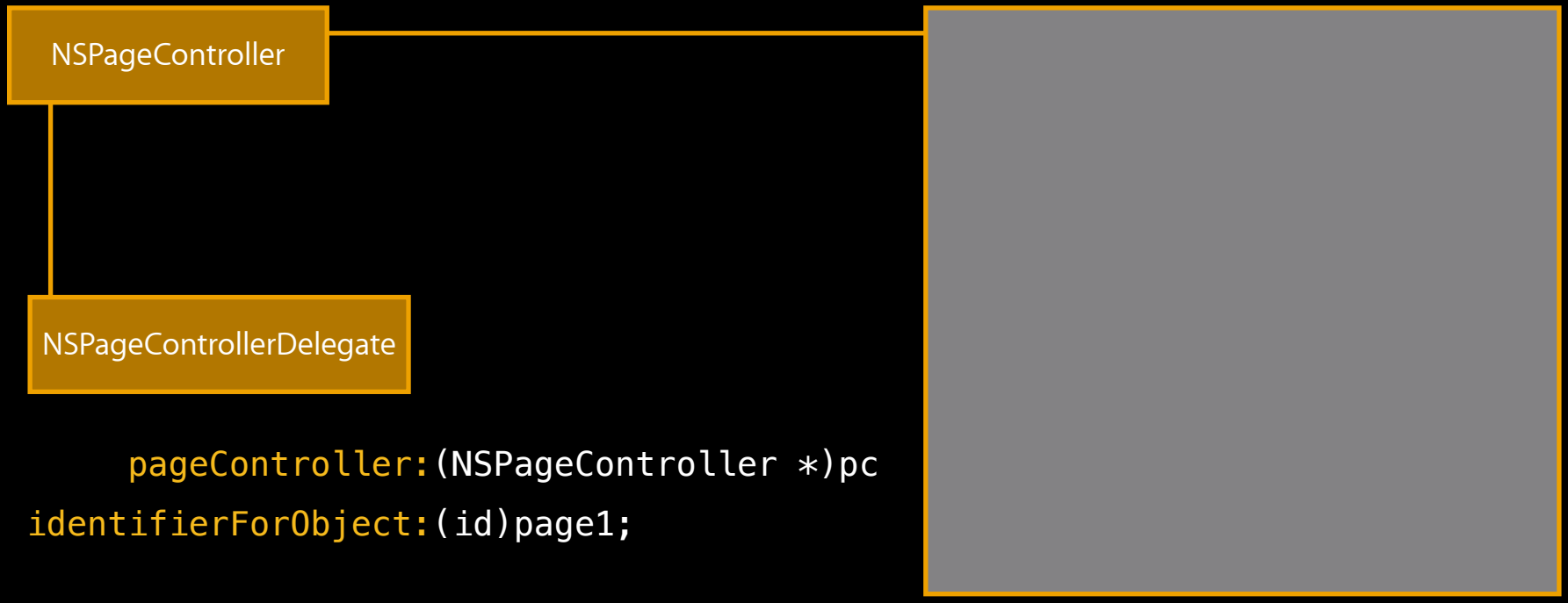
NSPageController

Custom mode



NSPageController

Custom mode




```
     pageController:(NSPageController *)pc
identifierForObject:(id)page1;
```

NSPageController

Custom mode

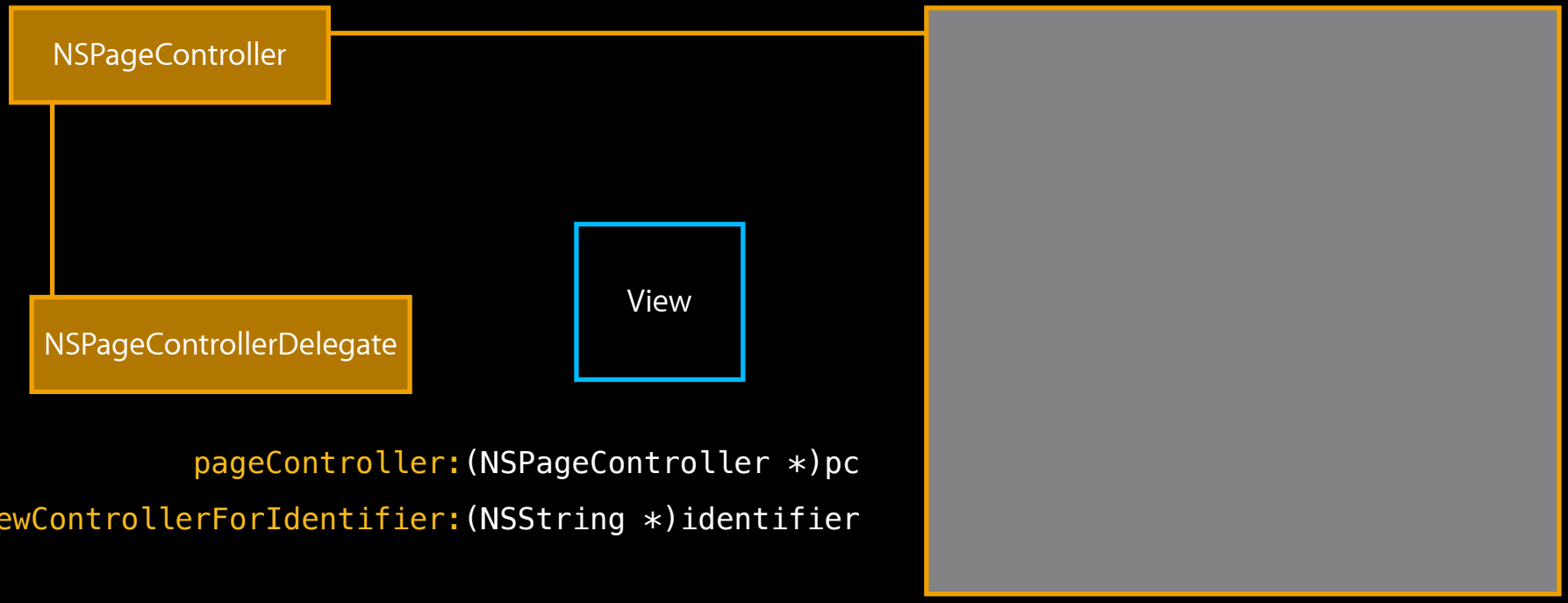
NSPageController

Custom mode

```
pageController:(NSPageController *)pc
viewControllerForIdentifier:(NSString *)identifier
```

NSPageController

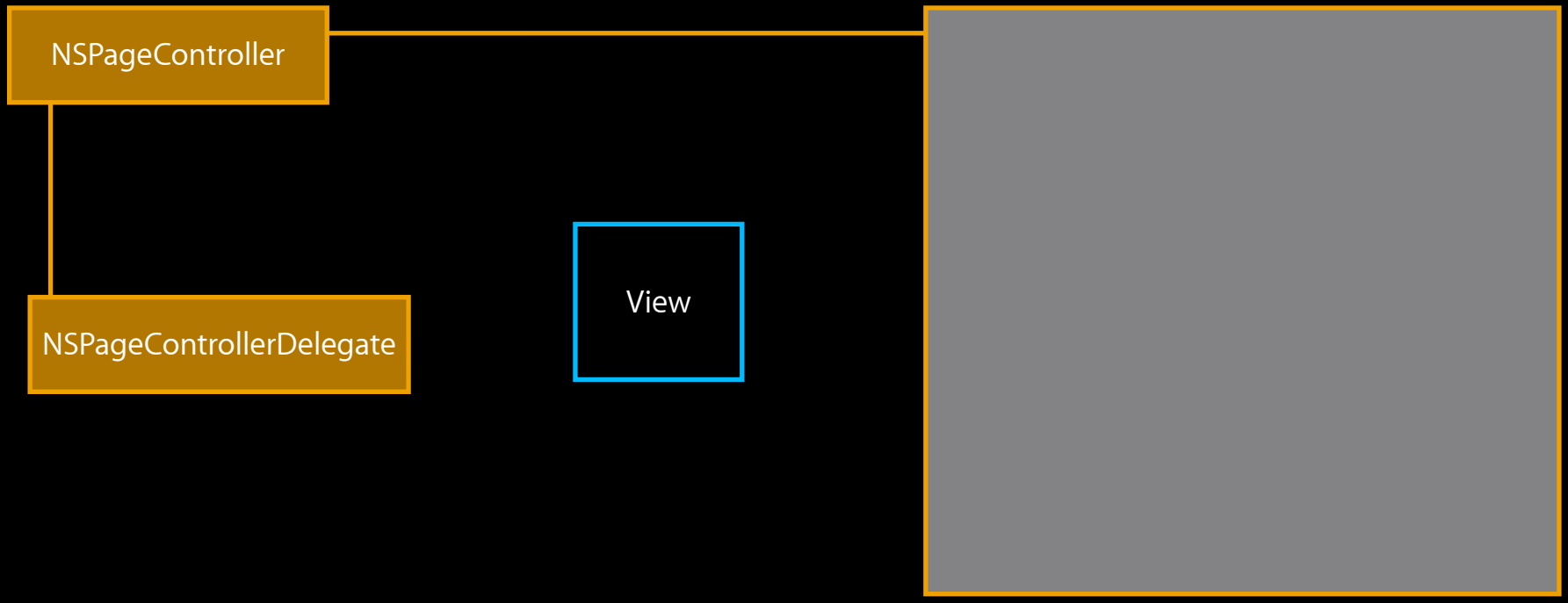
Custom mode




```
pageController:(NSPageController *)pc
viewControllerForIdentifier:(NSString *)identifier
```

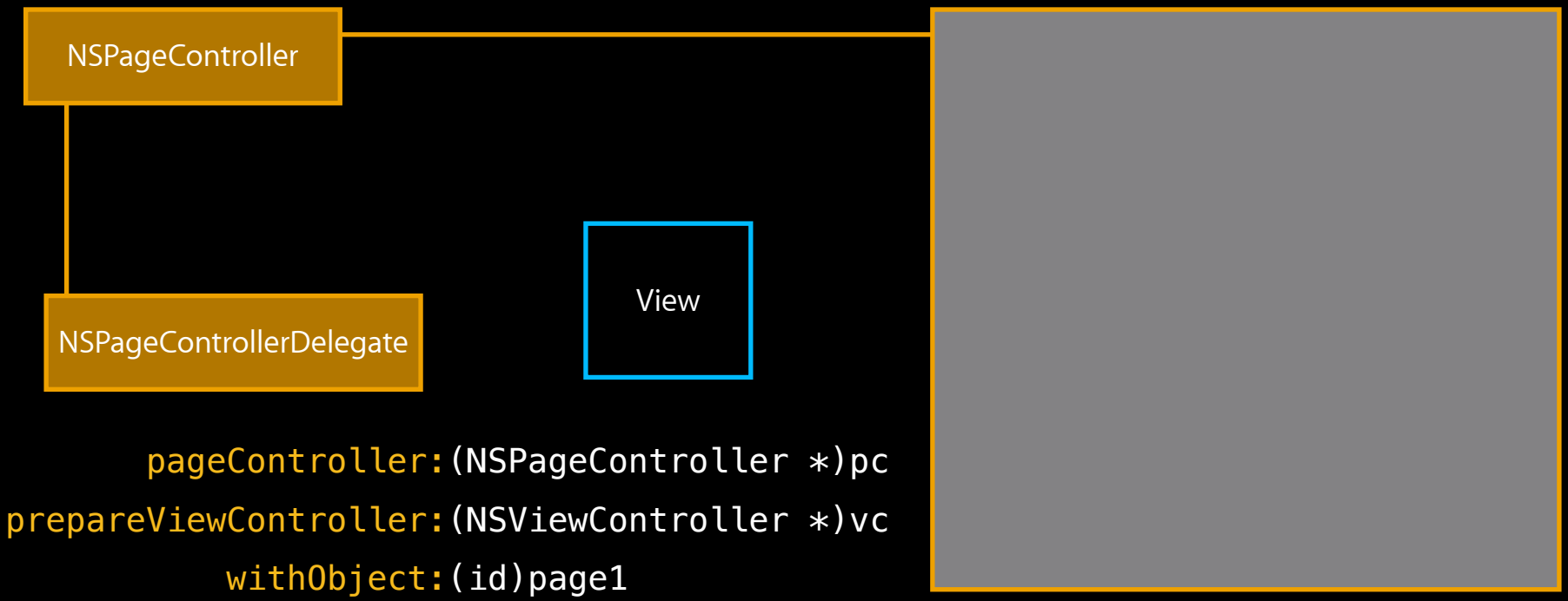
NSPageController

Custom mode



NSPageController

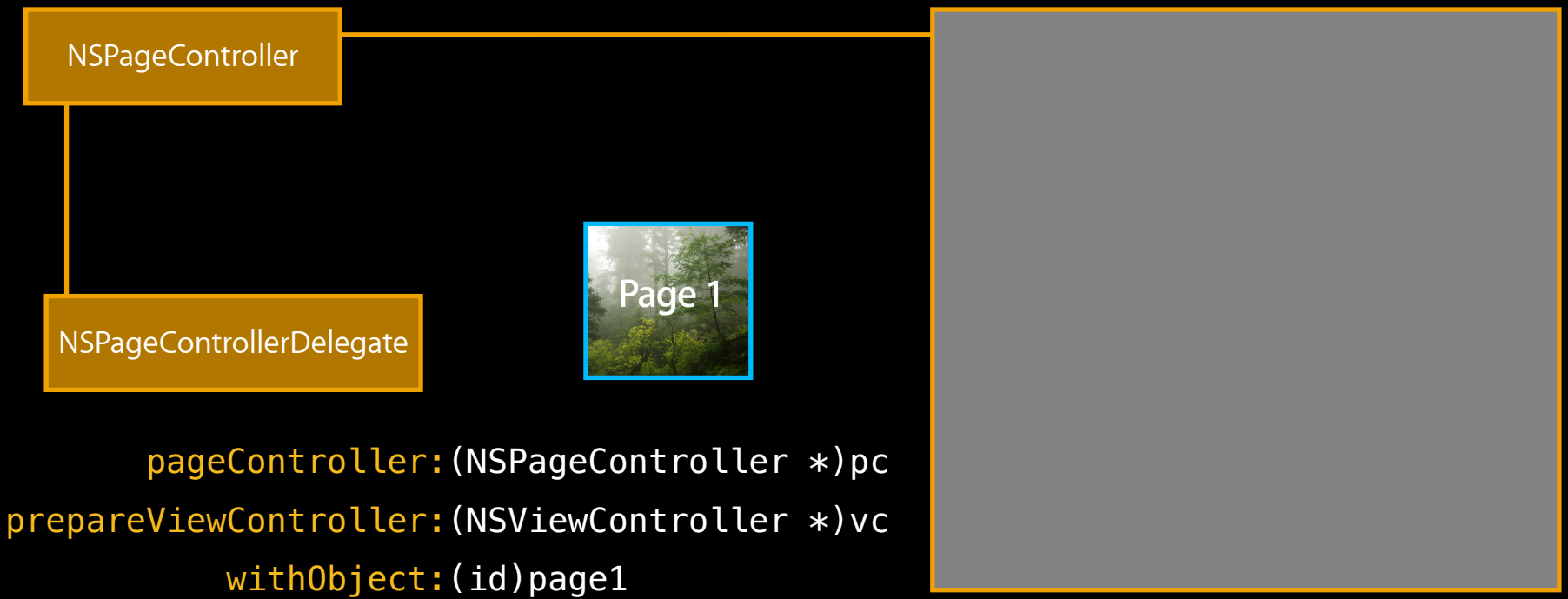
Custom mode




```
pageController:(NSPageController *)pc
prepareViewController:(NSViewController *)vc
withObject:(id)page1
```

NSPageController

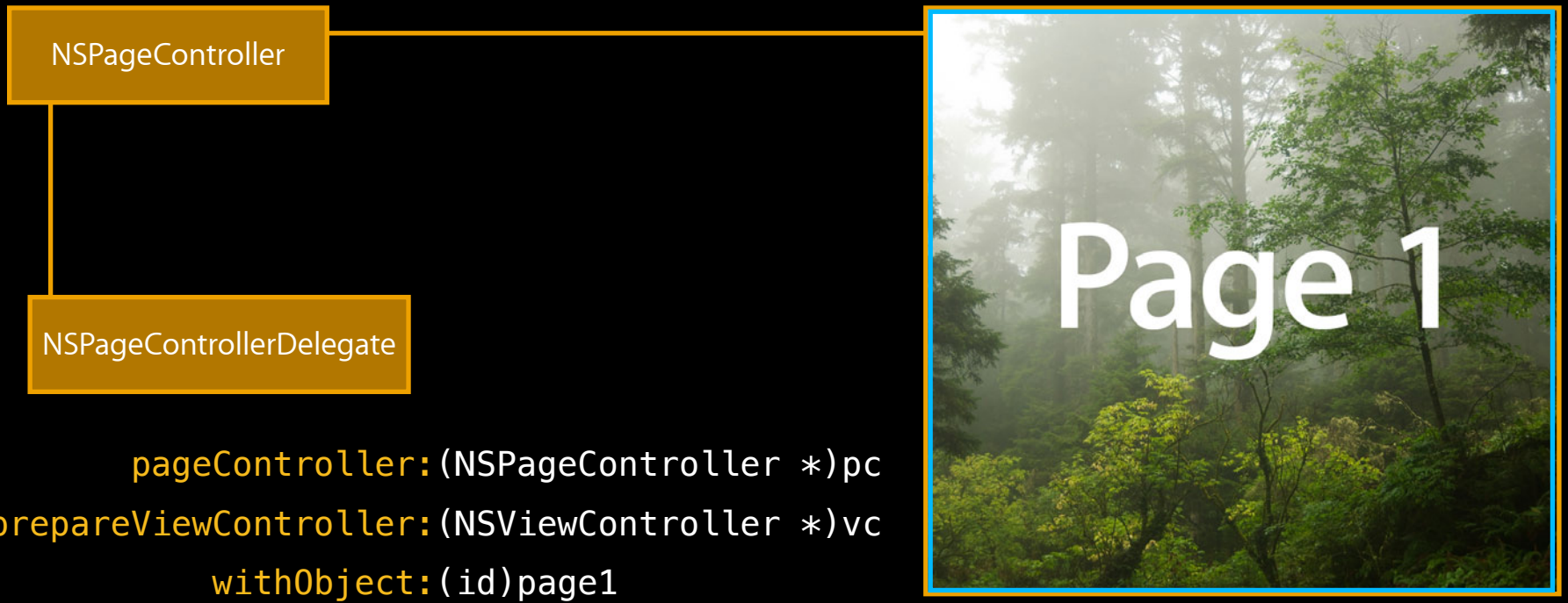
Custom mode




```
pageController:(NSPageController *)pc
prepareViewController:(NSViewController *)vc
withObject:(id)page1
```

NSPageController

Custom mode




```
pageController:(NSPageController *)pc
prepareViewController:(NSViewController *)vc
withObject:(id)page1
```

NSPageController

Custom mode

NSPageController

Custom mode

```
pageControllerWillStartLiveTransition:
```

NSPageController

Custom mode

NSPageController

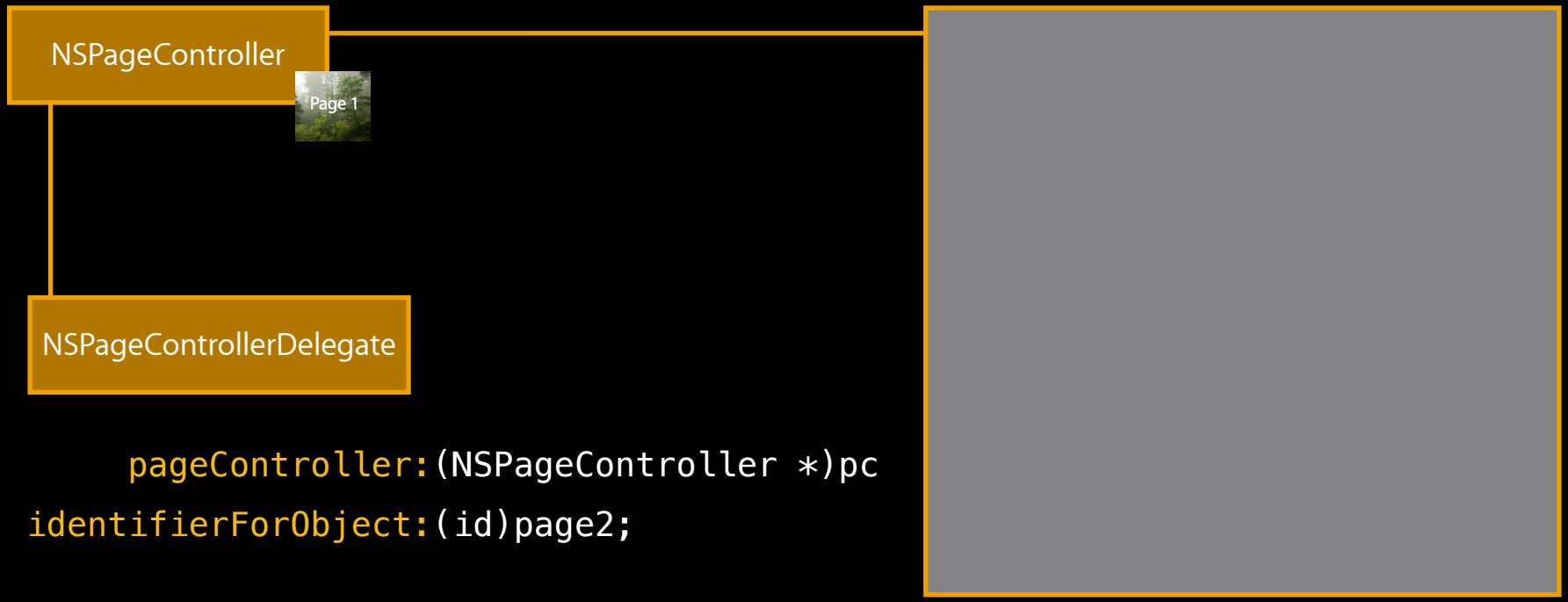
Custom mode

NSPageController

Custom mode

NSPageController

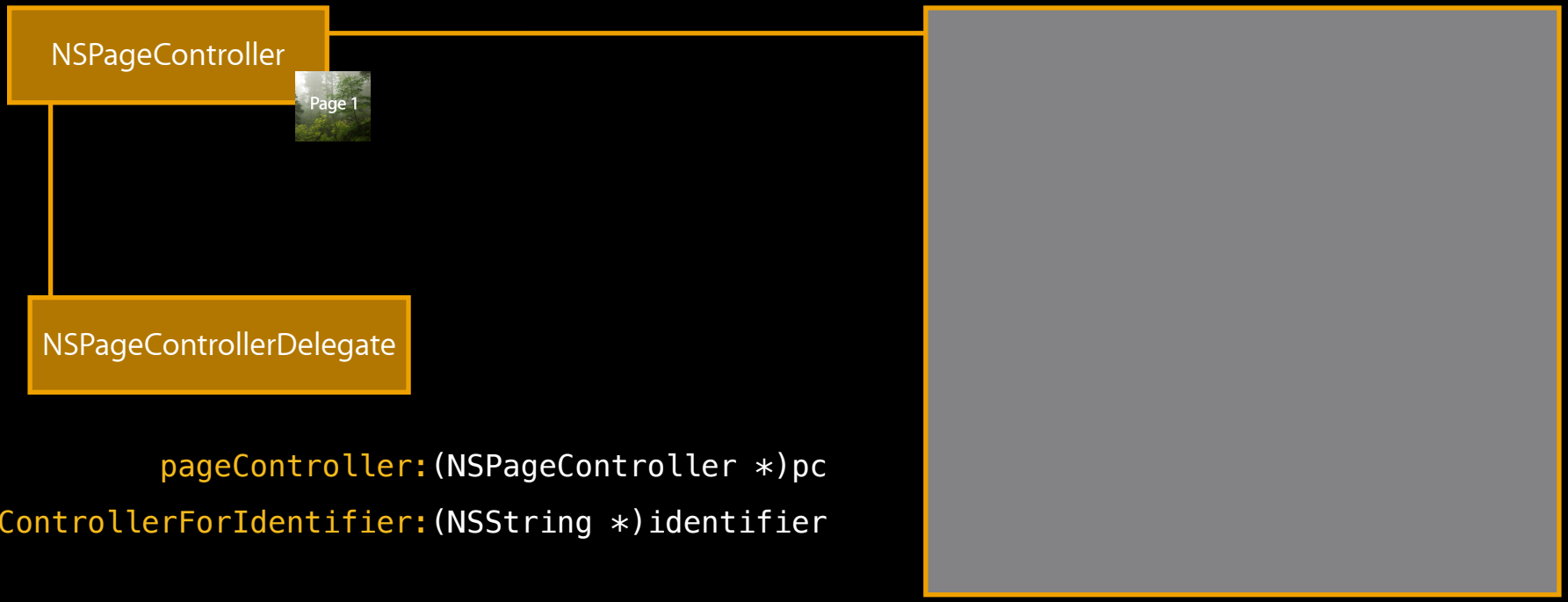
Custom mode




```
     pageController:(NSPageController *)pc
identifierForObject:(id)page2;
```

NSPageController

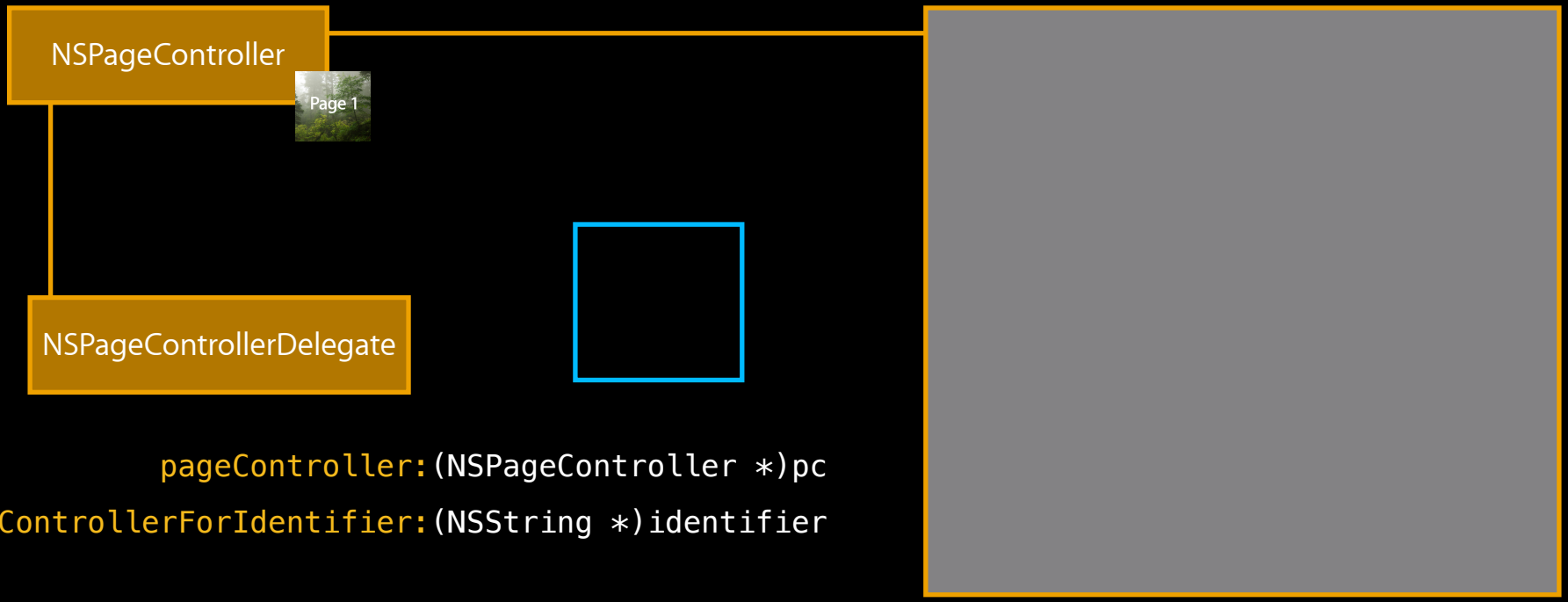
Custom mode




```
pageController:(NSPageController *)pc
viewControllerForIdentifier:(NSString *)identifier
```

NSPageController

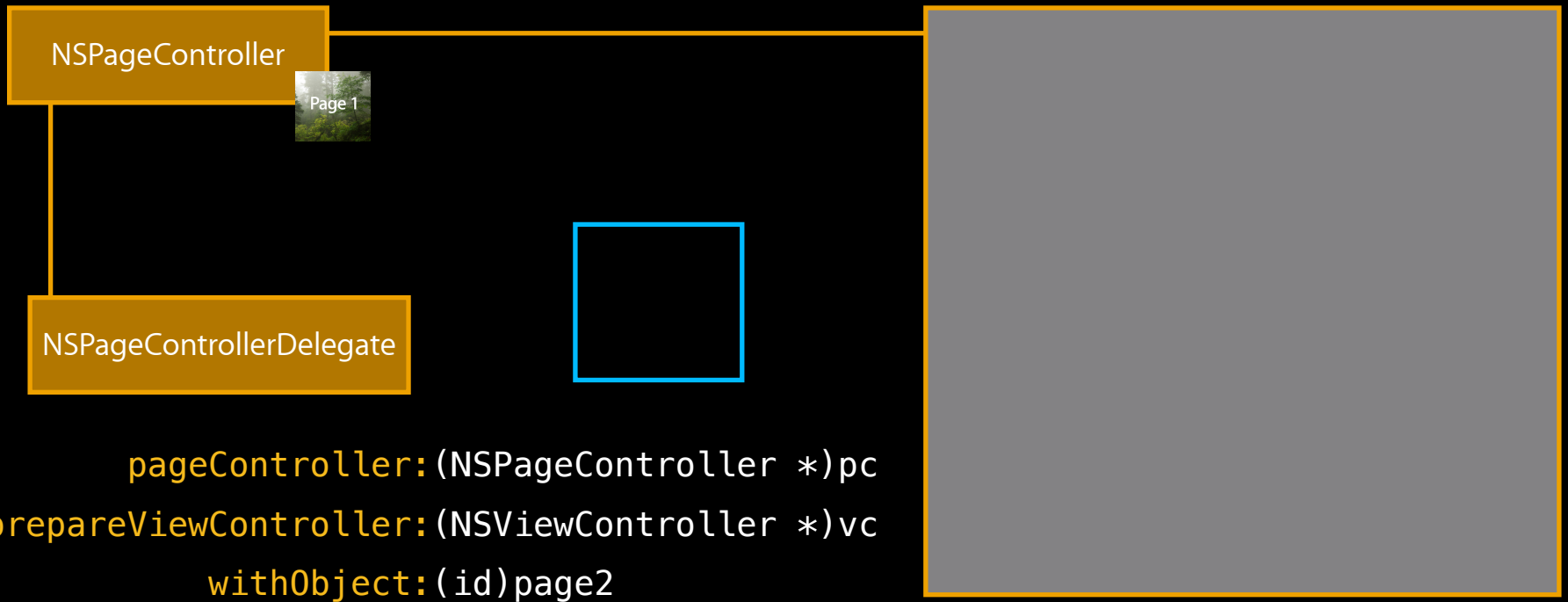
Custom mode




```
pageController:(NSPageController *)pc
viewControllerForIdentifier:(NSString *)identifier
```

NSPageController

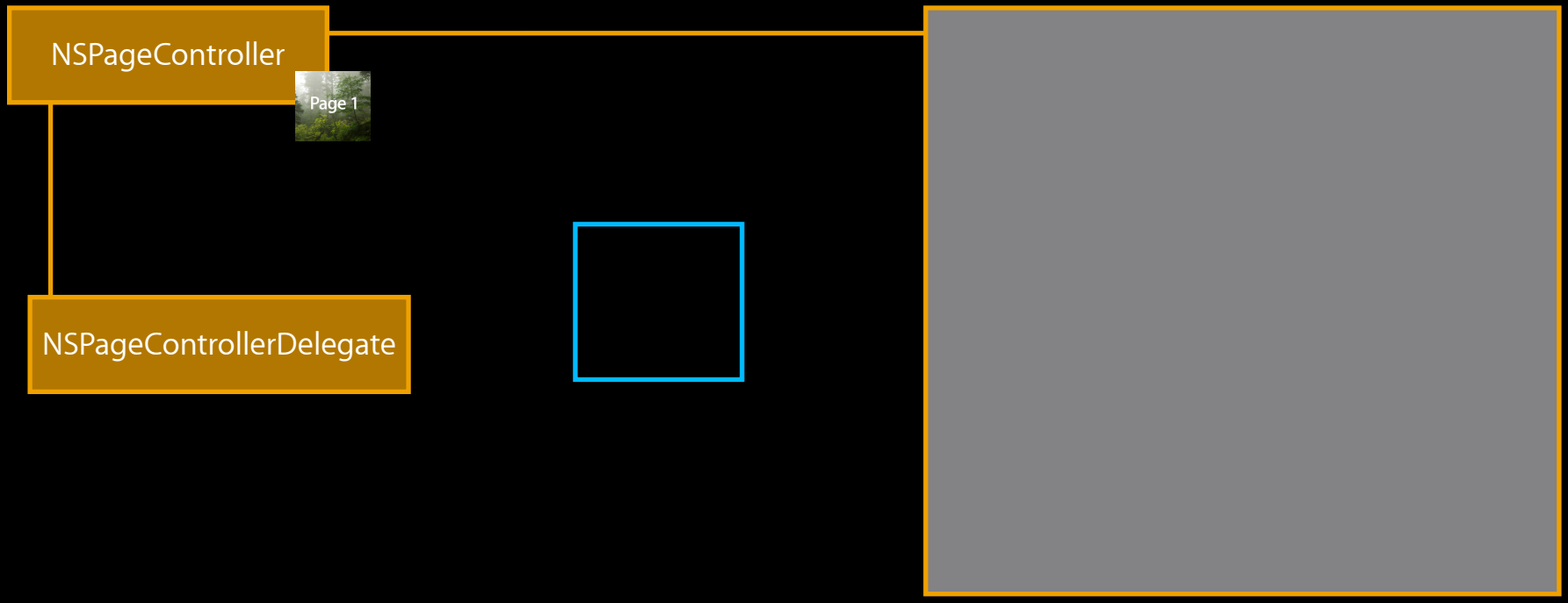
Custom mode




```
pageController:(NSPageController *)pc
prepareViewController:(NSViewController *)vc
withObject:(id)page2
```

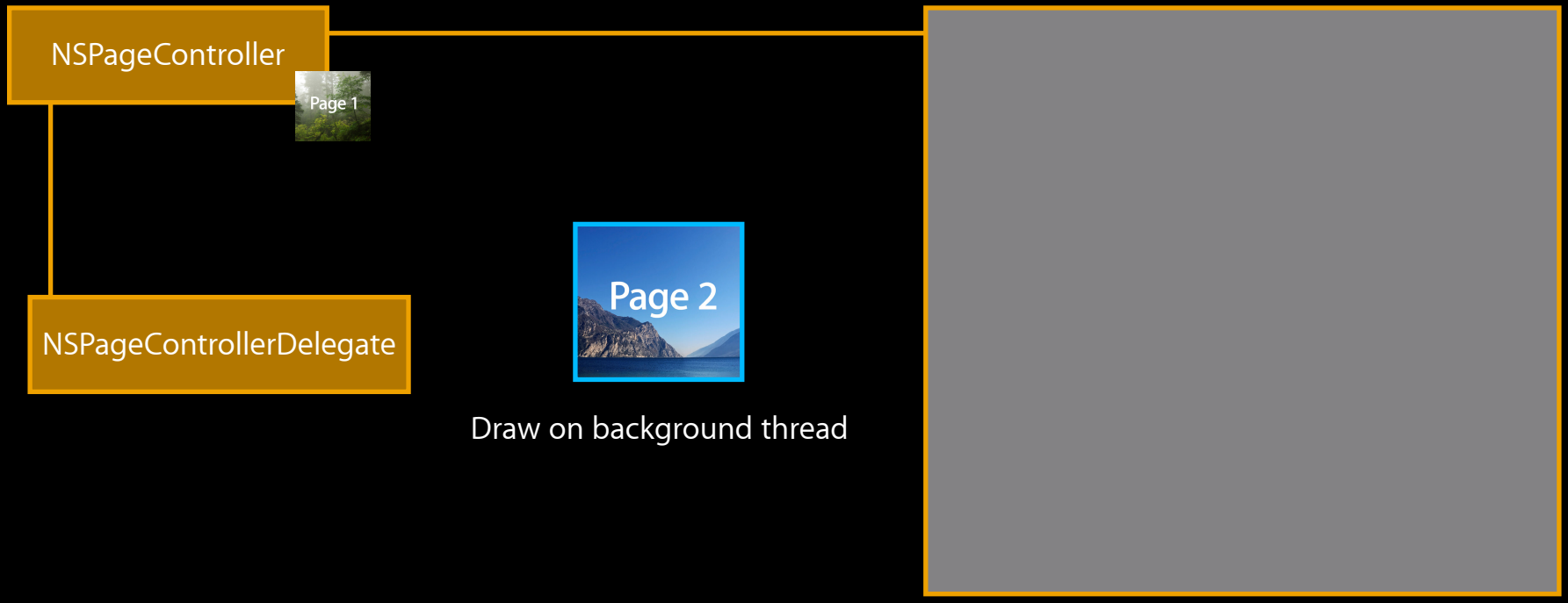
NSPageController

Custom mode



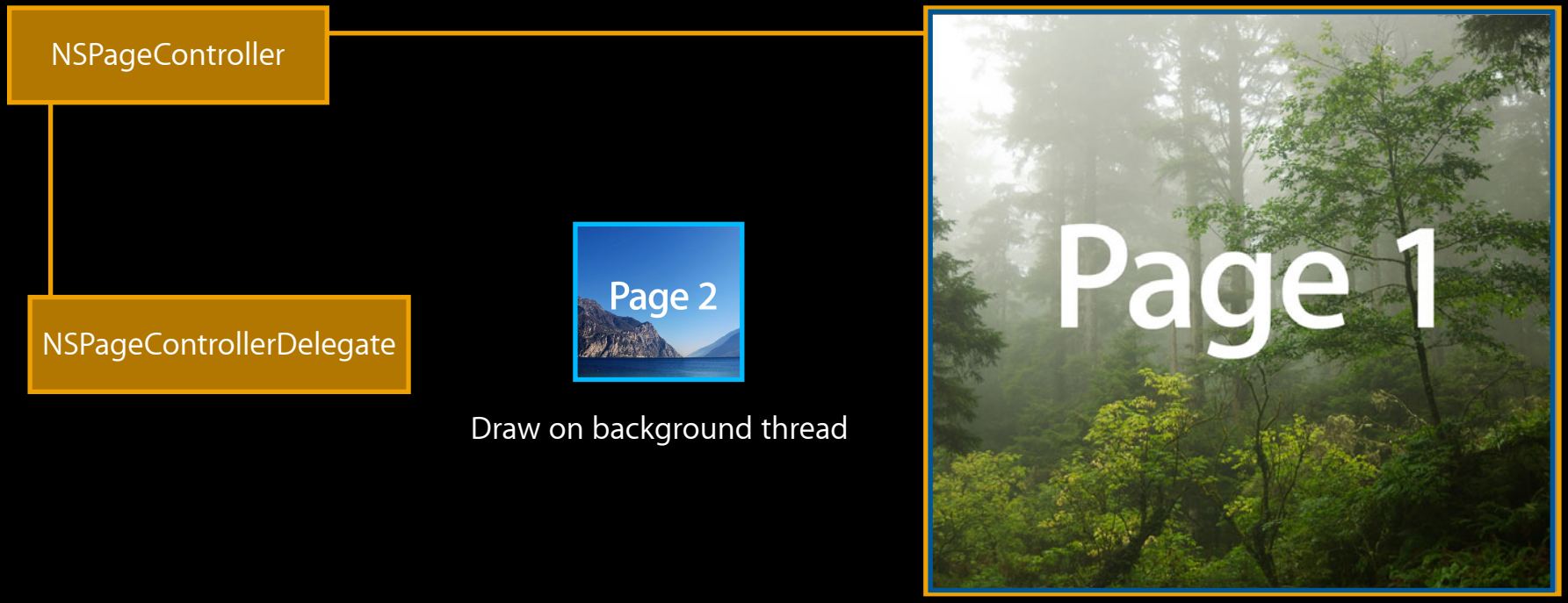
NSPageController

Custom mode



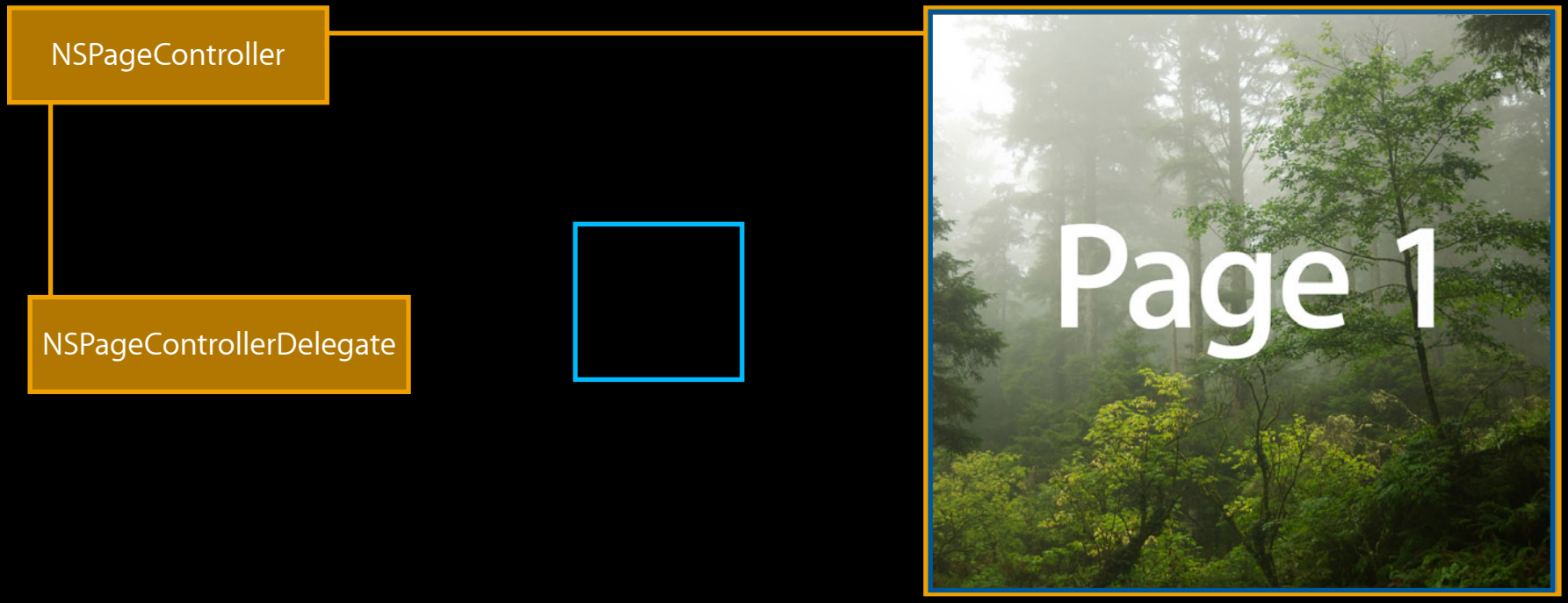
NSPageController

Custom mode



NSPageController

Custom mode



NSPageController

Custom mode

NSPageController

Custom mode

NSPageController

Custom mode

```
pageControllerDidEndLiveTransition:
```

NSPageController

Custom mode

NSPageController

Custom mode

Demo

File cards

NSPageController

Notes

- Do not connect to one of NSSplitView's views (history mode)

NSPageController

Notes

- Do not connect to one of NSSplitView's views (history mode)
- Snapshots are drawn on background thread

NSPageController

Notes

- Do not connect to one of NSSplitView's views (history mode)
- Snapshots are drawn on background thread
- Do not update NSTableView data while swiping in progress

NSPageController

Notes

- Do not connect to one of NSSplitView's views (history mode)
- Snapshots are drawn on background thread
- Do not update NSTableView data while swiping in progress
- When layer-backed, live layers are used

Quick Look

Quick Look

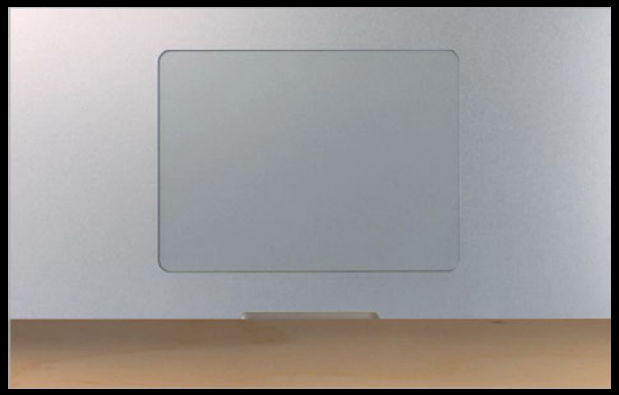
Triggers

- Three-finger single tap
  (trackpad only)

Quick Look

Triggers

- Three-finger single tap (trackpad only)



Quick Look

Triggers

- Three-finger single tap (trackpad only)
- Hot key

Quick Look

Triggers

- Three-finger single tap (trackpad only)
- Hot key

Quick Look

Responder methods

```
- (void)quickLookWithEvent:(NSEvent *)event
```

Quick Look

Responder methods

```
- (void)quickLookWithEvent:(NSEvent *)event
```

NSEventTypeQuickLook

Quick Look

Responder methods

```
- (void)quickLookWithEvent:(NSEvent *)event
```

NSEventTypeQuickLook

-locationInWindow

Quick Look

Responder methods

```
- (void)quickLookWithEvent:(NSEvent *)event
```

`NSEventTypeQuickLook`

`-locationInWindow`

No corresponding event mask

Quick Look

Responder methods

```
- (void)quickLookWithEvent:(NSEvent *)event
- (void)quickLookPreviewItems:(id)sender
```

Quick Look

Responder methods

```
- (void)quickLookWithEvent:(NSEvent *)event
- (void)quickLookPreviewItems:(id)sender
```

Quick Look

Responder methods

```
- (void)quickLookWithEvent:(NSEvent *)event
- (void)quickLookPreviewItems:(id)sender
```

NSApp routes to Dictionary lookup

Demo

Quick Look

Best Practices

Attempts at future proofing

Best Practices

- Use built-in implementations

Best Practices

- Use built-in implementations
- Observe NSClipView bounds

Best Practices

- Use built-in implementations
- Observe NSClipView bounds

```
NSViewBoundsDidChangeNotification
```

Best Practices

- Use built-in implementations
- Observe NSClipView bounds

```
[[NSNotificationCenter defaultCenter] addObserver:self
                                      selector:@selector(boundsChanged:)
                                      name:NSViewBoundsDidChangeNotification
                                      object:[scrollView contentView]]
```

Best Practices

- Use built-in implementations
- Observe NSClipView bounds
- Forward end and cancelled gesture event

Best Practices

- Use built-in implementations
- Observe NSClipView bounds
- Forward end and cancelled gesture event
- Forward scrollWheel events with end and cancelled phases

Summary

Summary

- API hooks for Smart Magnify and Quick Look gestures

Summary

- API hooks for Smart Magnify and Quick Look gestures
- NSScrollView has built-in support for magnification

Summary

- API hooks for Smart Magnify and Quick Look gestures
- NSScrollView has built-in support for magnification
- NSPageController

More Information

Jake Behrens
UI Frameworks Evangelist
behrens@apple.com

Documentation
Mac OS X Human Interface Guidelines
http://developer.apple.com/ue

Apple Developer Forums
http://devforums.apple.com

Related Sessions

Layer-Backed Views: AppKit + Core Animation	Nob Hill Wednesday 10:15AM
Building Advanced Gesture Recognizers	Marina Thursday 11:30AM

Labs

OS X Gestures and Cocoa Lab	Essentials Lab B Thursday 2:00PM
Cocoa and XPC Lab	Essentials Lab A Friday 10:15AM

WWDC2012