

Auto Layout by Example

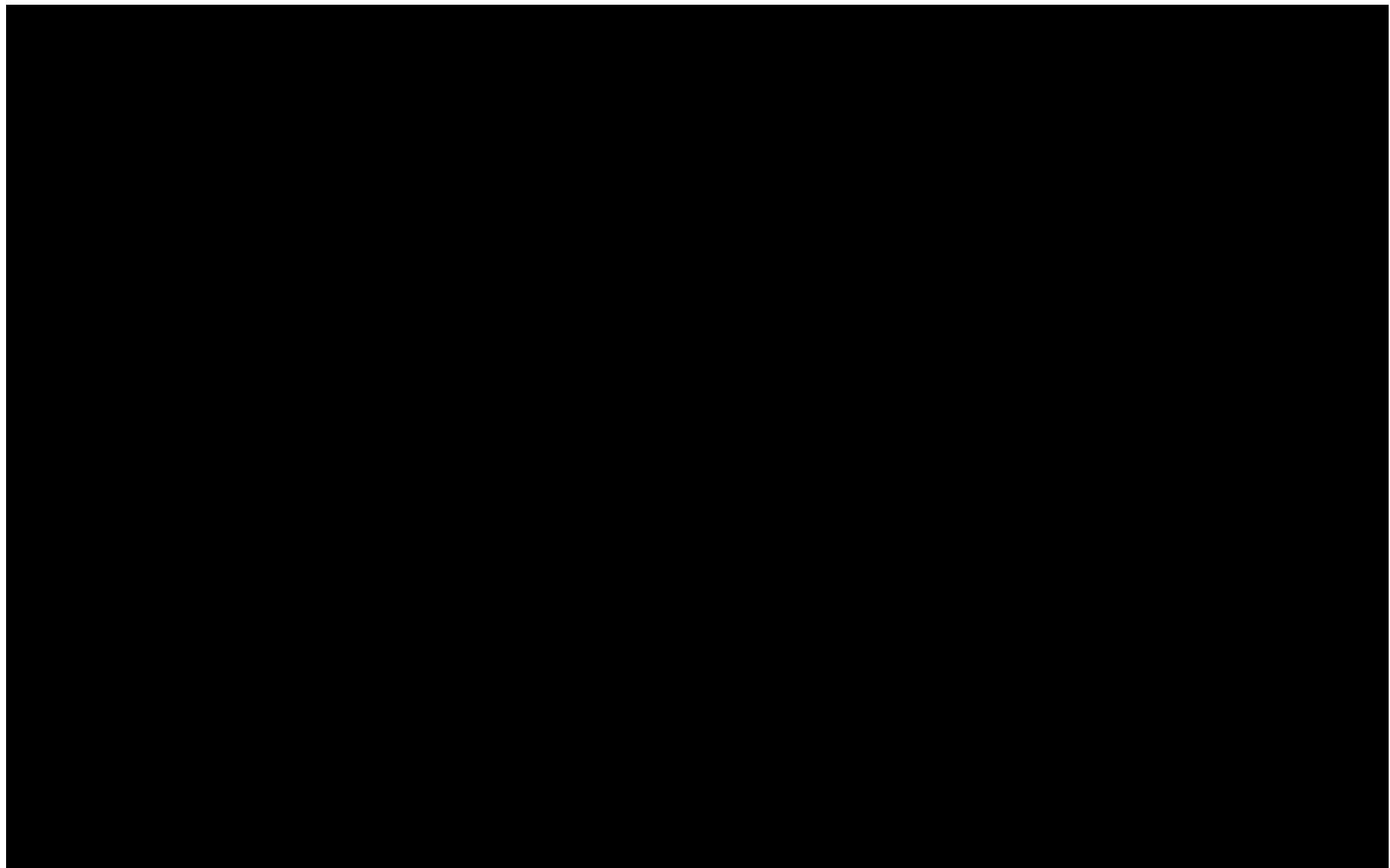
Putting it all together

Session 232

Kevin Cathey

Interface Builder Engineer

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



How do I do that?



Transitioning to Auto Layout

Transitioning to Auto Layout

Explicit Widths

Transitioning to Auto Layout

Explicit Widths

Localization

Transitioning to Auto Layout

Explicit Widths

Localization

New API

Transitioning to Auto Layout

- Explicit Widths

- Localization

- New API

- Animation

Transitioning to Auto Layout

- Explicit Widths

- Localization

- New API

- Animation

Transitioning to Auto Layout

Explicit Widths

Localization

New API

Animation

Transitioning to Auto Layout

Explicit Widths

Localization

New API

Animation

Transitioning to Auto Layout

- Plan your attack—partial or full conversion, or new content
- In Interface Builder
 - Enable Auto Layout
 - Inspect, modify, add, and delete constraints
- In code
 - Convert `-setFrame` calls
 - Sanitize `-addSubview` and `-removeFromSuperview` calls
- Test

Demo

Transitioning part of an application to Auto Layout

Transitioning to Auto Layout

Recap

- Plan your attack—partial or full conversion, or new content
- In Interface Builder
 - Enable Auto Layout
 - Inspect, modify, add, and delete constraints
- In code
 - Convert `- setFrame` calls
 - Sanitize `- addSubview` and `- removeFromSuperview` calls
- Test

Transitioning to Auto Layout

Recap

- Best practices for constraints
 - Avoid explicit widths for controls (more on this soon)
 - Add spacing constraints to avoid overlap
 - Let window minimum size be driven by constraints
 - Other expressive constraints where applicable (e.g. equal sizes)

Transitioning to Auto Layout

Explicit Widths

Localization

New API

Animation

Transitioning to Auto Layout

Explicit Widths

Localization

New API

Animation

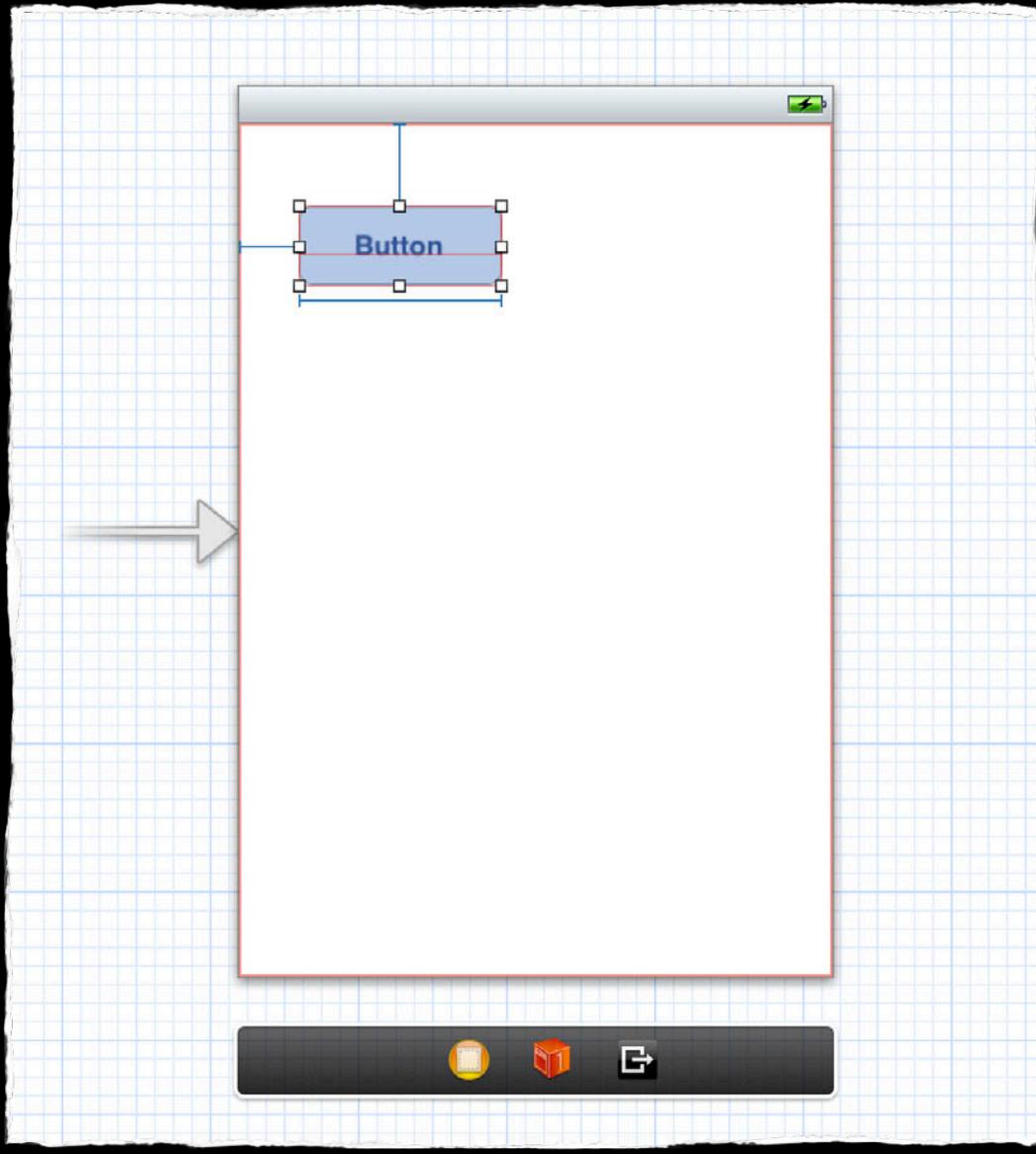
Transitioning to Auto Layout

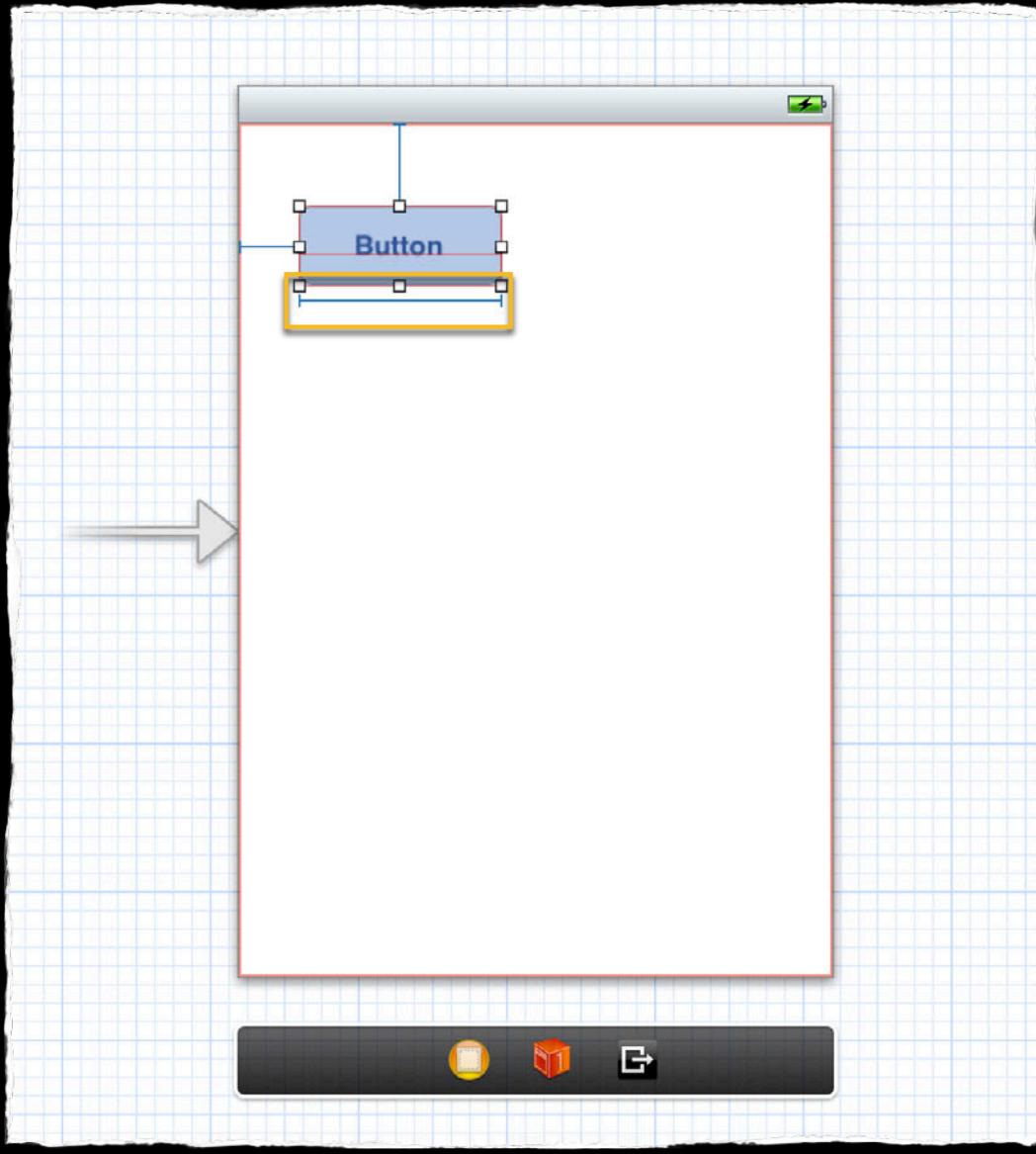
Explicit Widths

Localization

New API

Animation





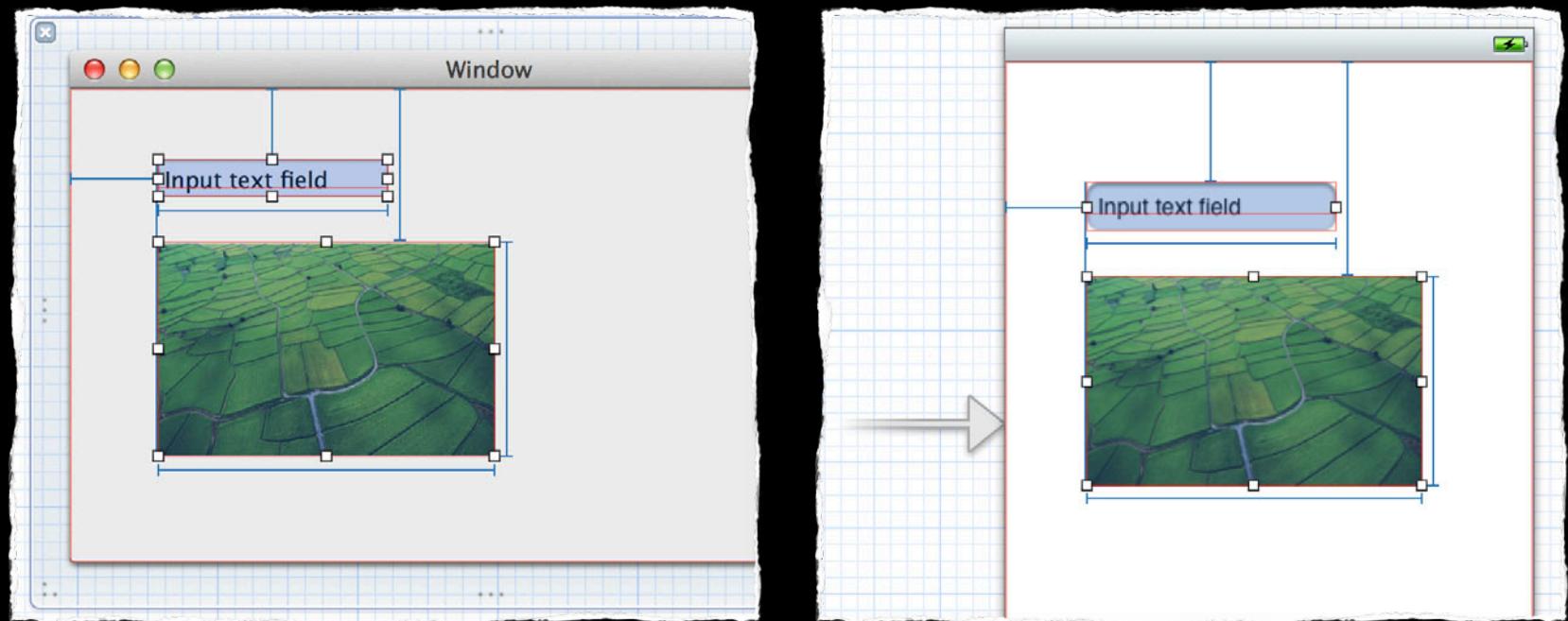
Explicit Widths

Generally speaking

- Generally, avoid explicit widths on controls
- Prevents controls from reacting to changes in content
- Want intrinsic content size to drive size of controls
- Can cause clipping during localization or with dynamic content

Explicit Widths

Few examples of exceptions



Bordered (input) text fields and image views

Explicit Widths in Interface Builder

How do I get rid of them?

- How Interface Builder generates constraints for you
 - Picks minimum and best set of constraints
 - Avoids unsatisfiable constraints
 - Avoids ambiguity

Demo

Explicit widths in Interface Builder

Explicit Widths in Interface Builder

Recap

- Add the constraint you want
- If visual layout can be changed
 - Keep controls at their intrinsic content size (size to fit size)
- If visual layout cannot be changed
 - Change relation of explicit widths to \geq
- Adjust content hugging and compression resistance priorities

Transitioning to Auto Layout

Explicit Widths

Localization

New API

Animation

Transitioning to Auto Layout

Explicit Widths

Localization

New API

Animation

Transitioning to Auto Layout

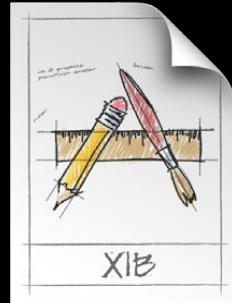
Explicit Widths

Localization

New API

Animation

Localization Before Auto Layout



Development Region
(English)

Localization Before Auto Layout



Development Region
(English)

Localization Before Auto Layout



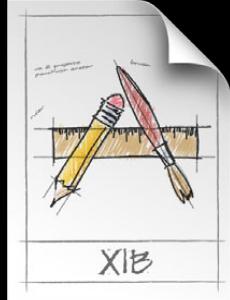
Development Region
(English)

Engineering

Localized Nibs



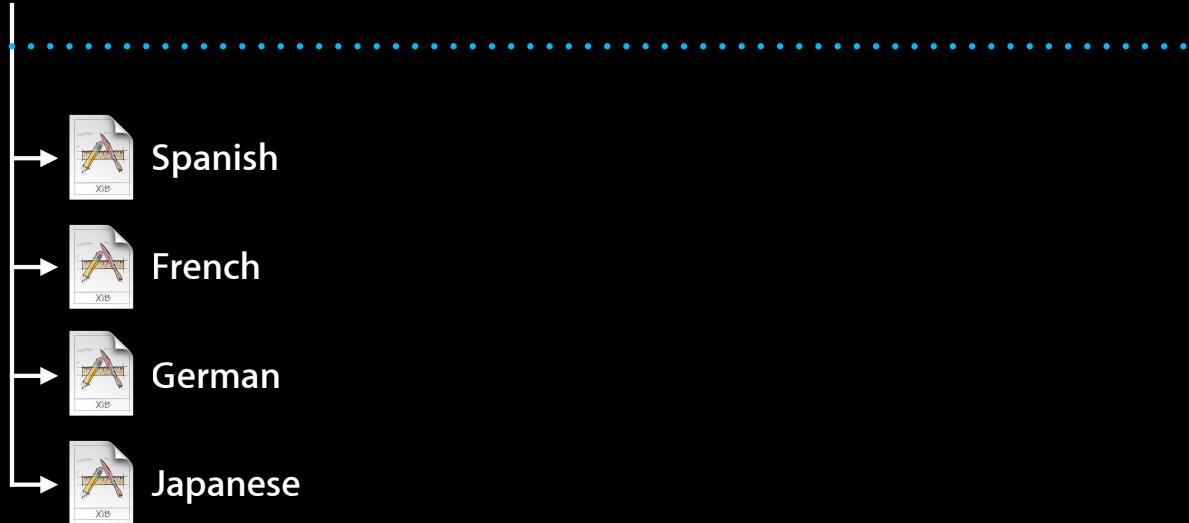
Localization Before Auto Layout



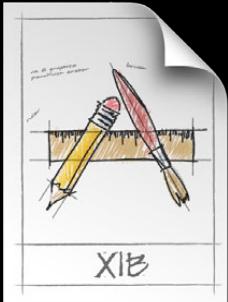
Development Region
(English)

Engineering

Localized Nibs



Localization Before Auto Layout



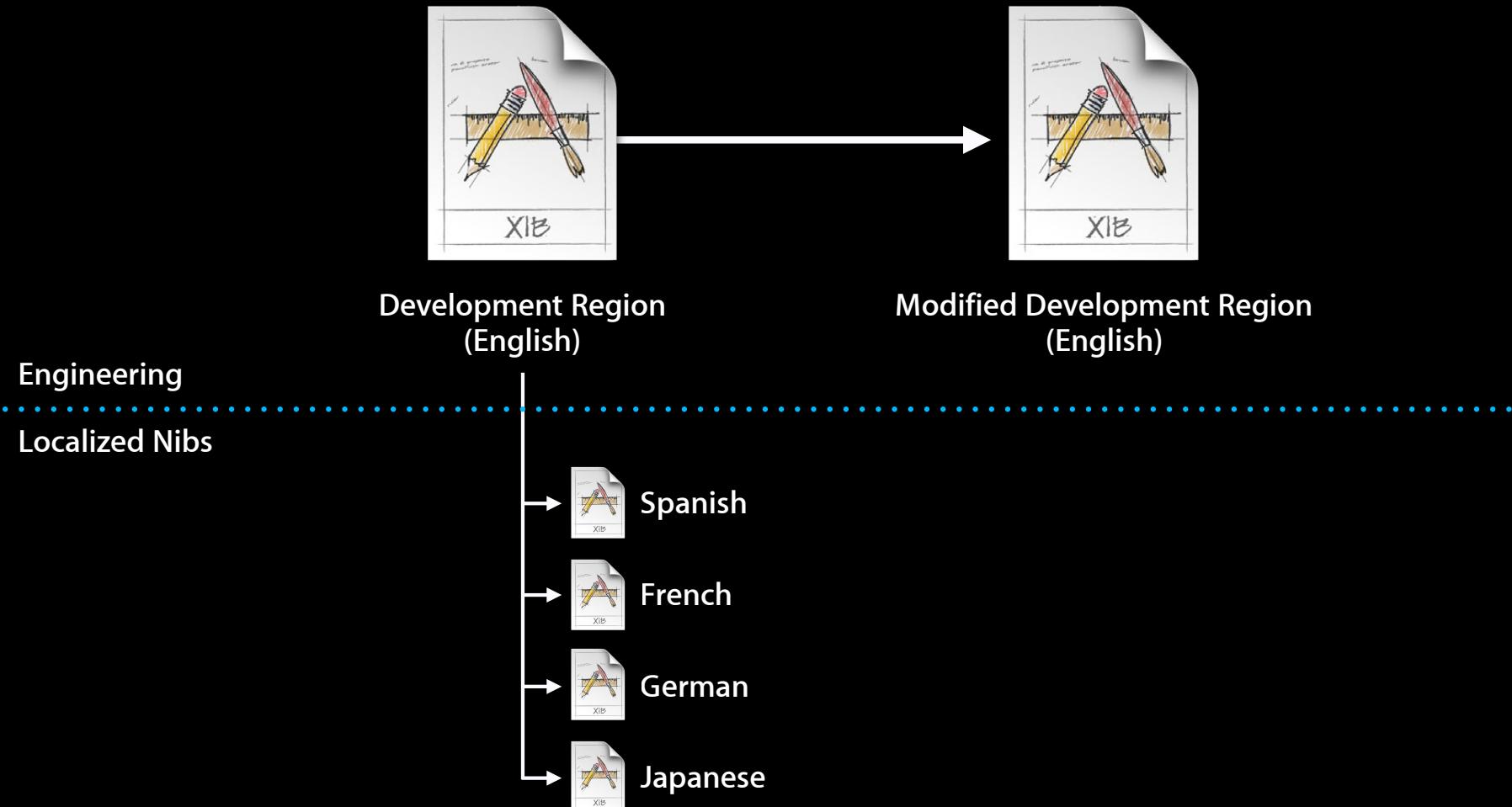
Development Region
(English)

Engineering

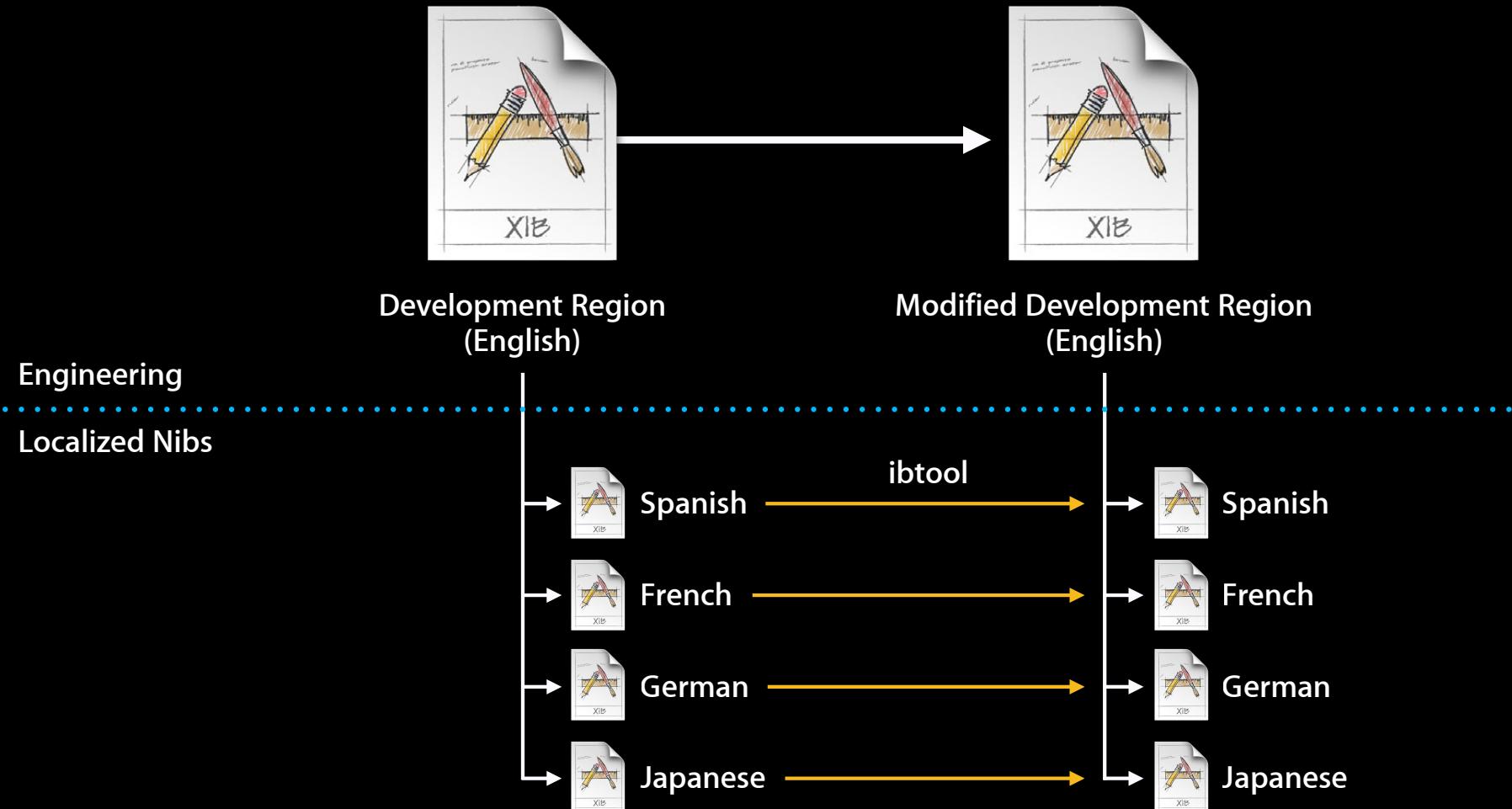
Localized Nibs

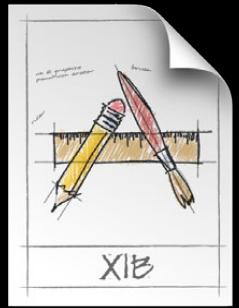


Localization Before Auto Layout



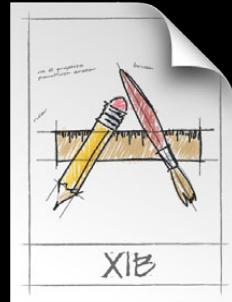
Localization Before Auto Layout





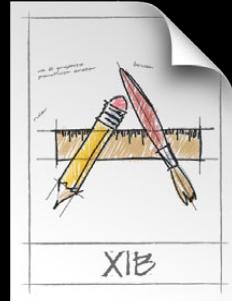
Development Region
(English)

Localization with Auto Layout



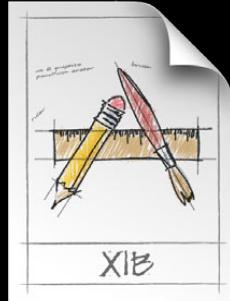
Development Region
(English)

Localization with Auto Layout



Development Region
(English)

Localization with Auto Layout



Development Region
(English)

Engineering

Localized Strings



Spanish



French

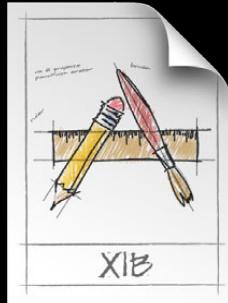


German



Japanese

Localization with Auto Layout



Development Region
(English)

Engineering

Localized Strings



Spanish



French

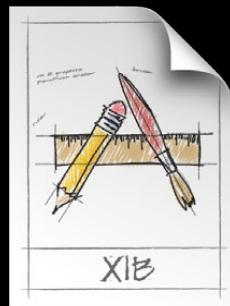


German



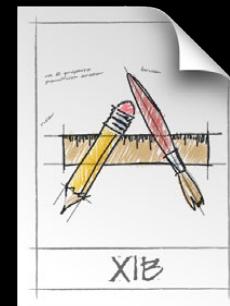
Japanese

Localization with Auto Layout



Development Region
(English)

Engineering



Modified Development Region
(English)

Localized Strings



Spanish



French

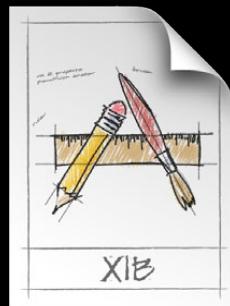


German



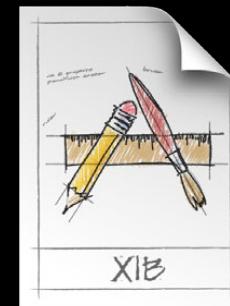
Japanese

Localization with Auto Layout



Development Region
(English)

Engineering



Modified Development Region
(English)

Localized Strings



Spanish



French



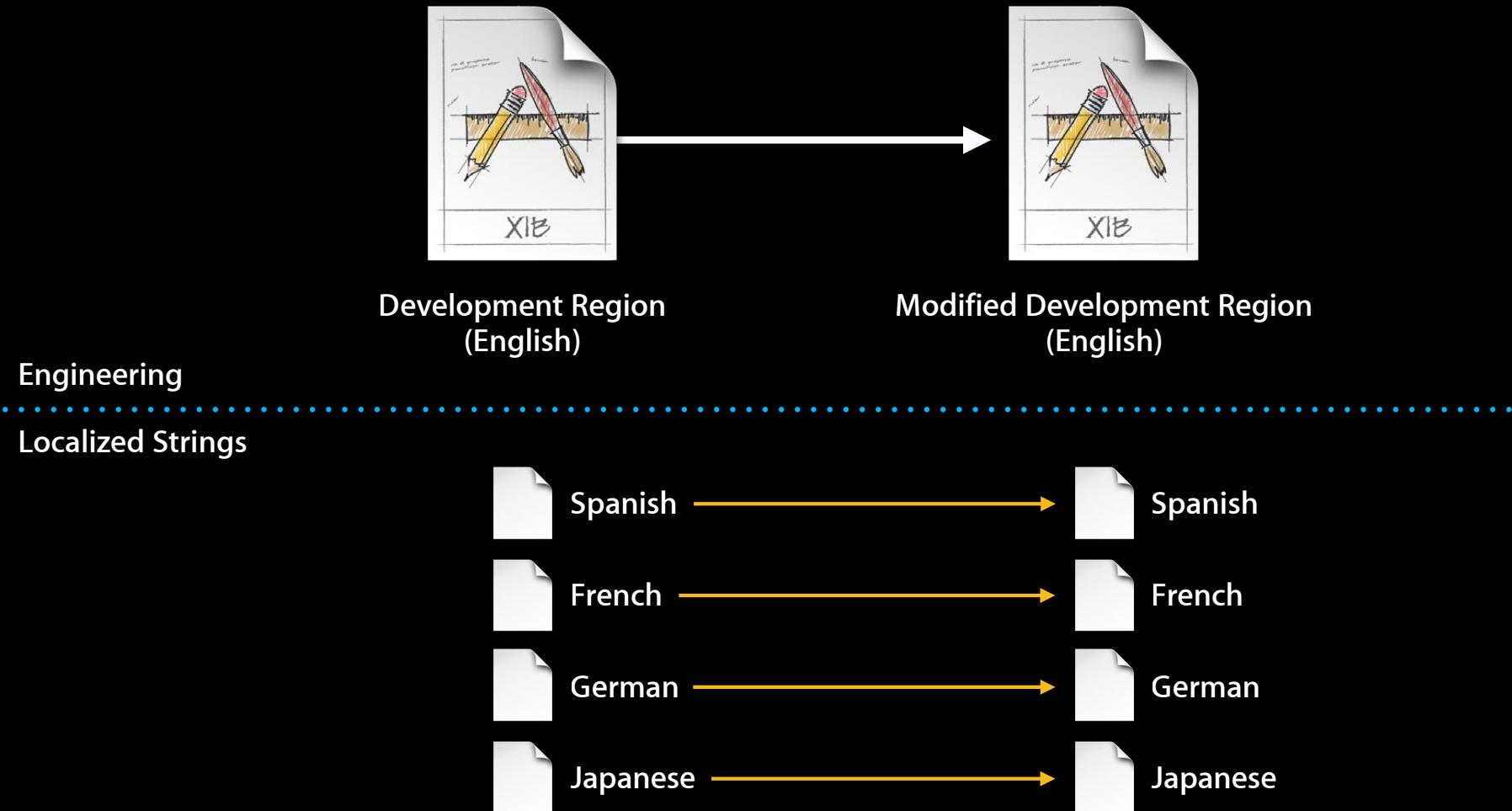
German



Japanese

New strings might
not be needed

Localization with Auto Layout



How Does That Work?

Localization and Auto Layout at runtime

How Does That Work?

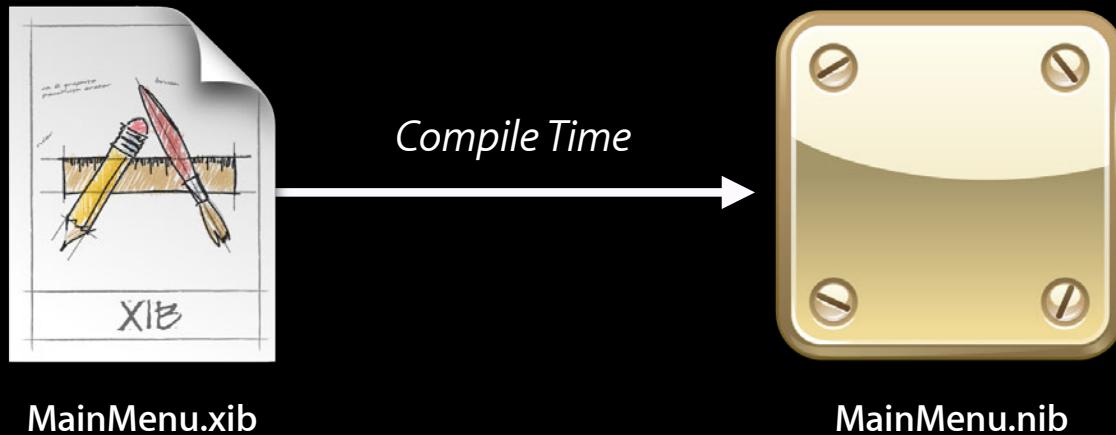
Localization and Auto Layout at runtime



MainMenu.xib

How Does That Work?

Localization and Auto Layout at runtime



How Does That Work?

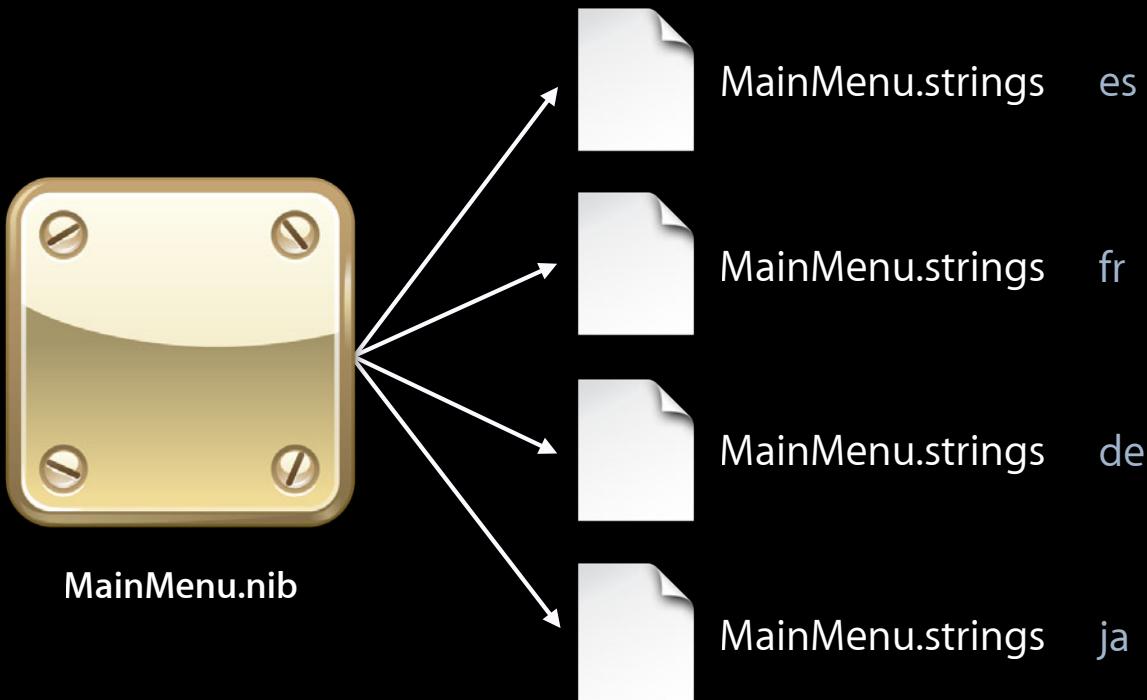
Localization and Auto Layout at runtime



MainMenu.nib

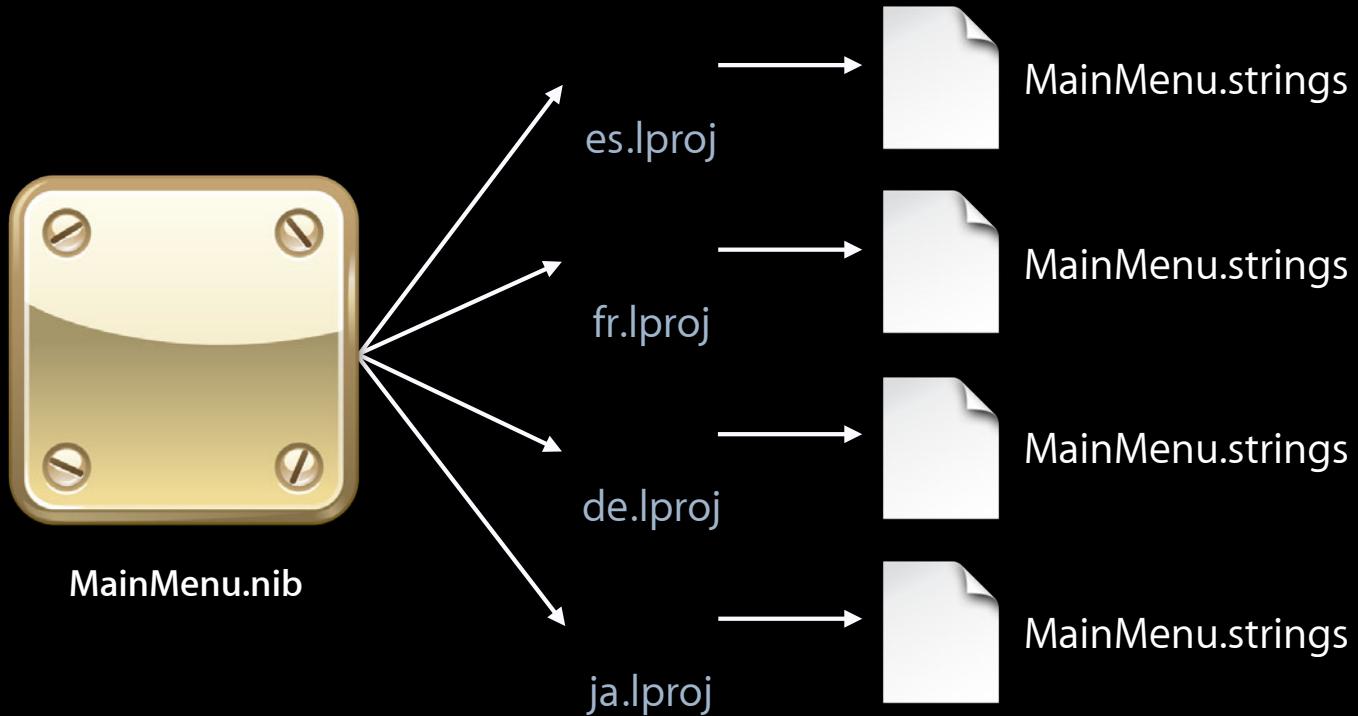
How Does That Work?

Localization and Auto Layout at runtime



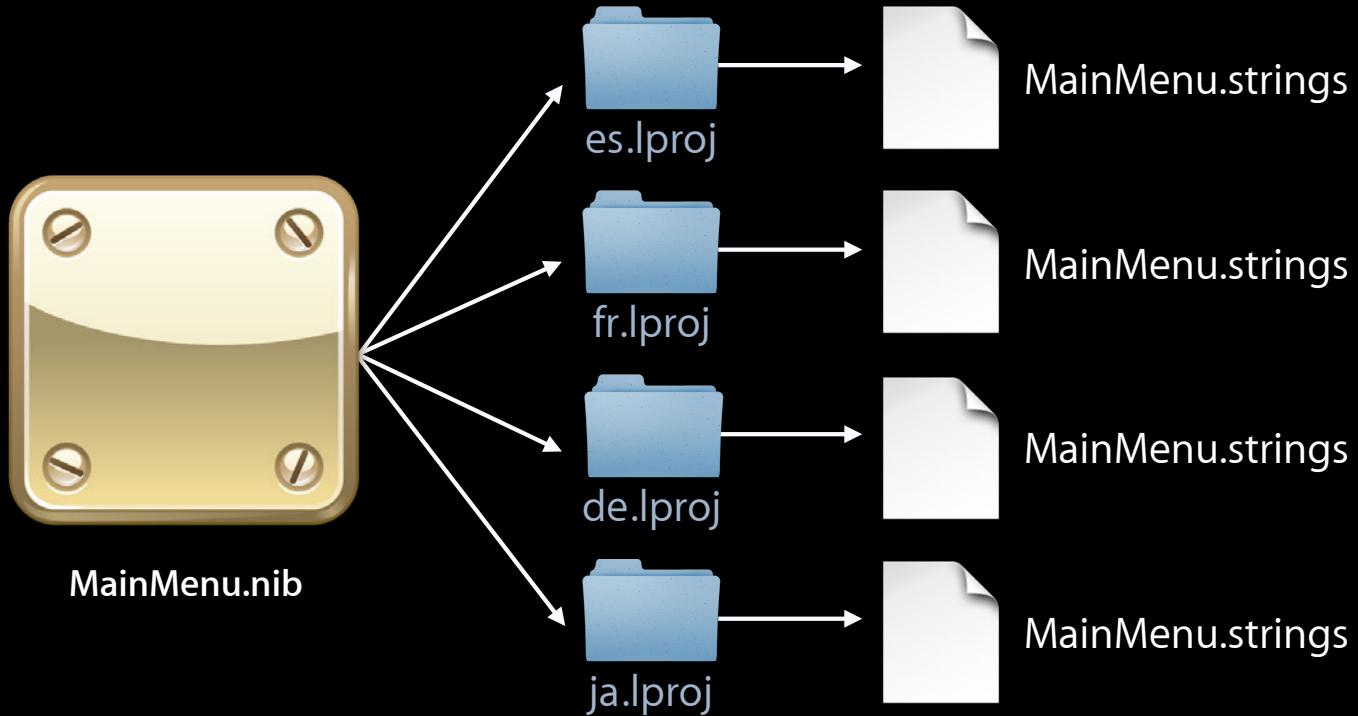
How Does That Work?

Localization and auto layout at runtime



How Does That Work?

Localization and auto layout at runtime



What About MainMenu.nib?

The base localization



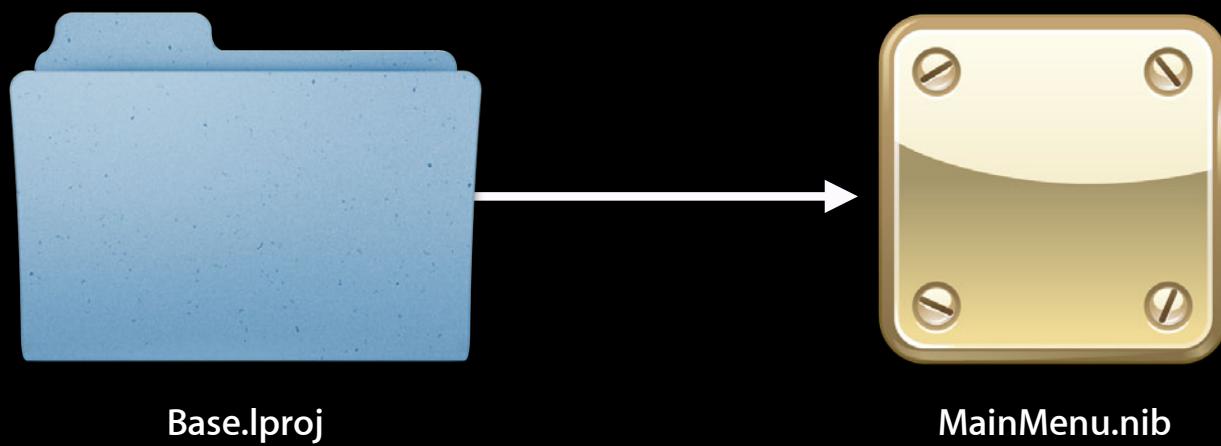
Base.iproj



MainMenu.nib

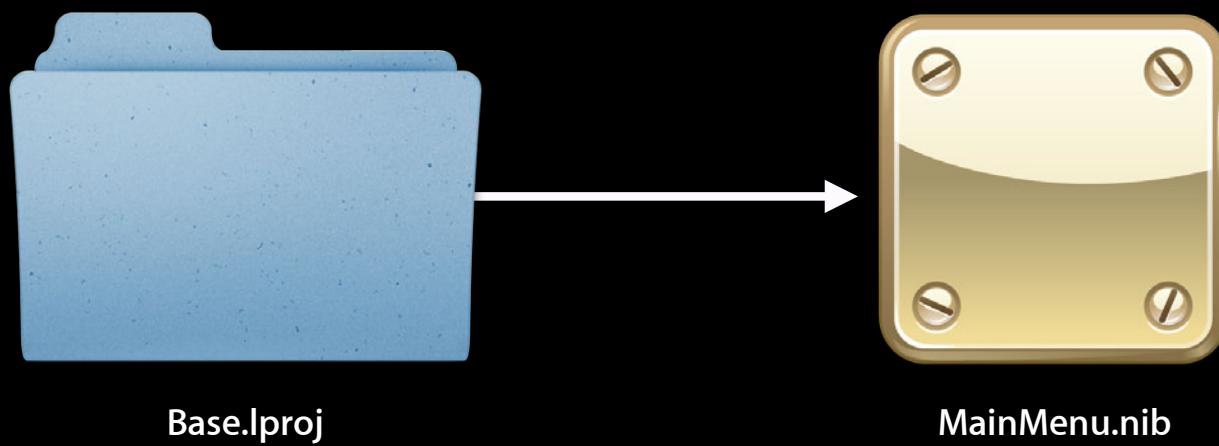
What About MainMenu.nib?

The base localization

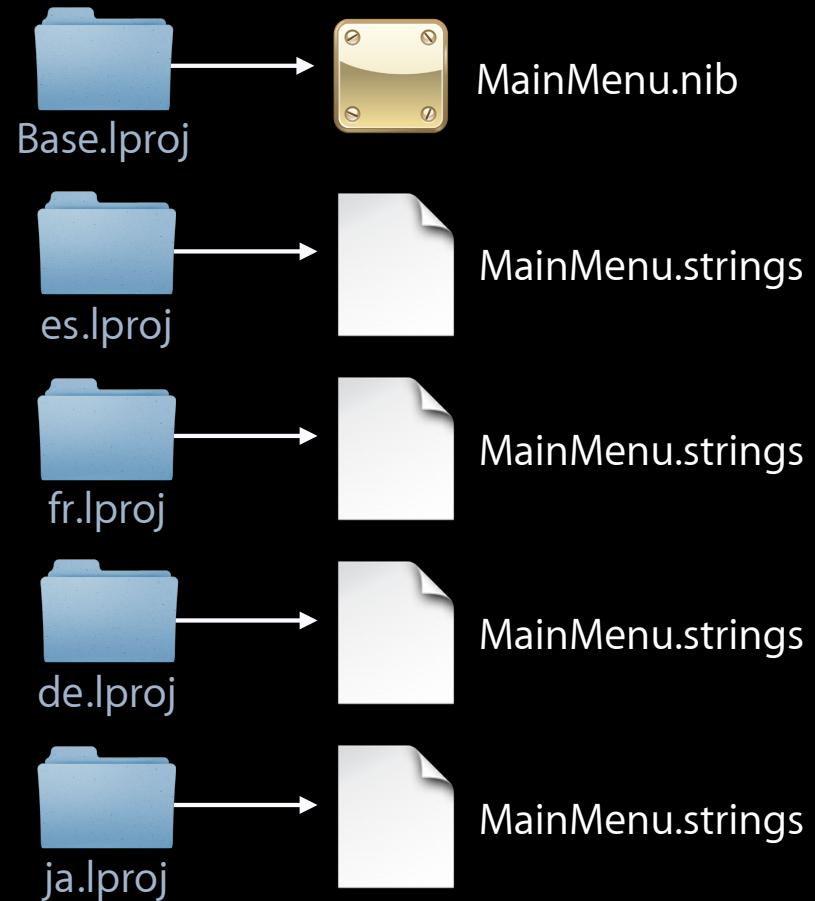


What About MainMenu.nib?

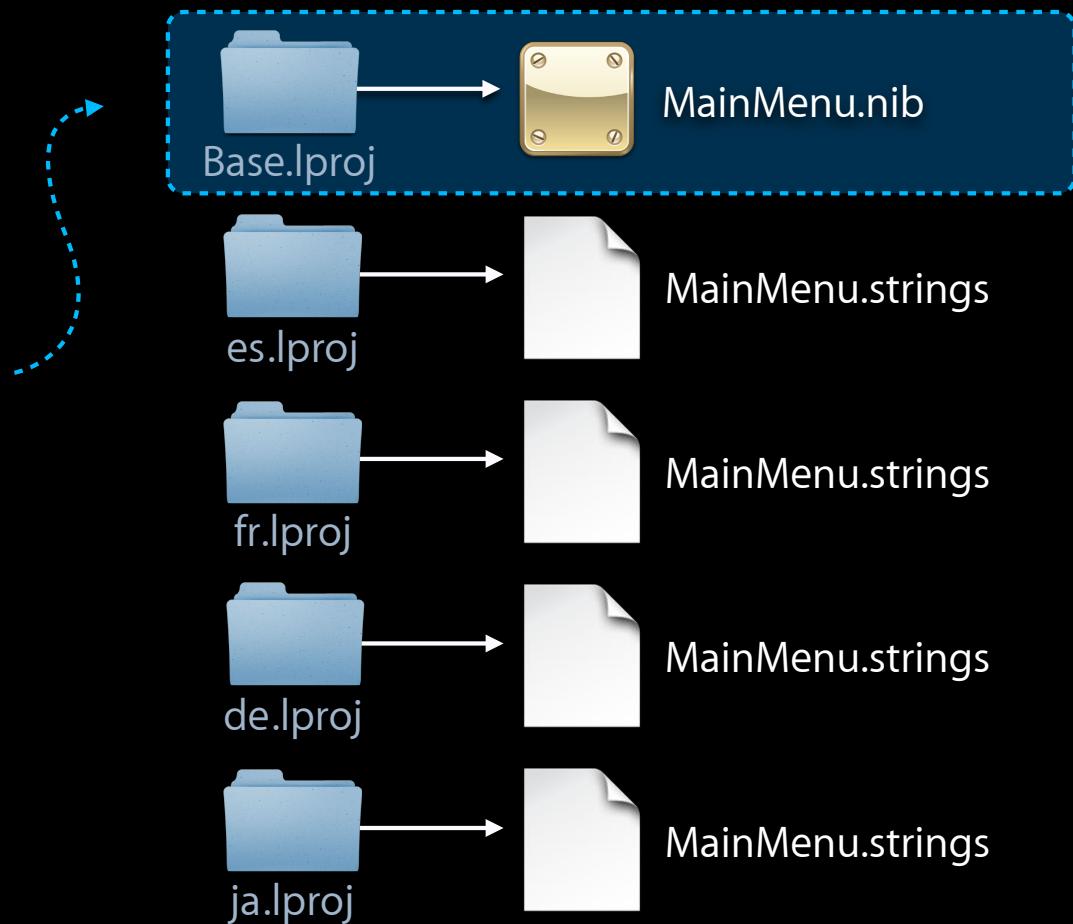
The base localization



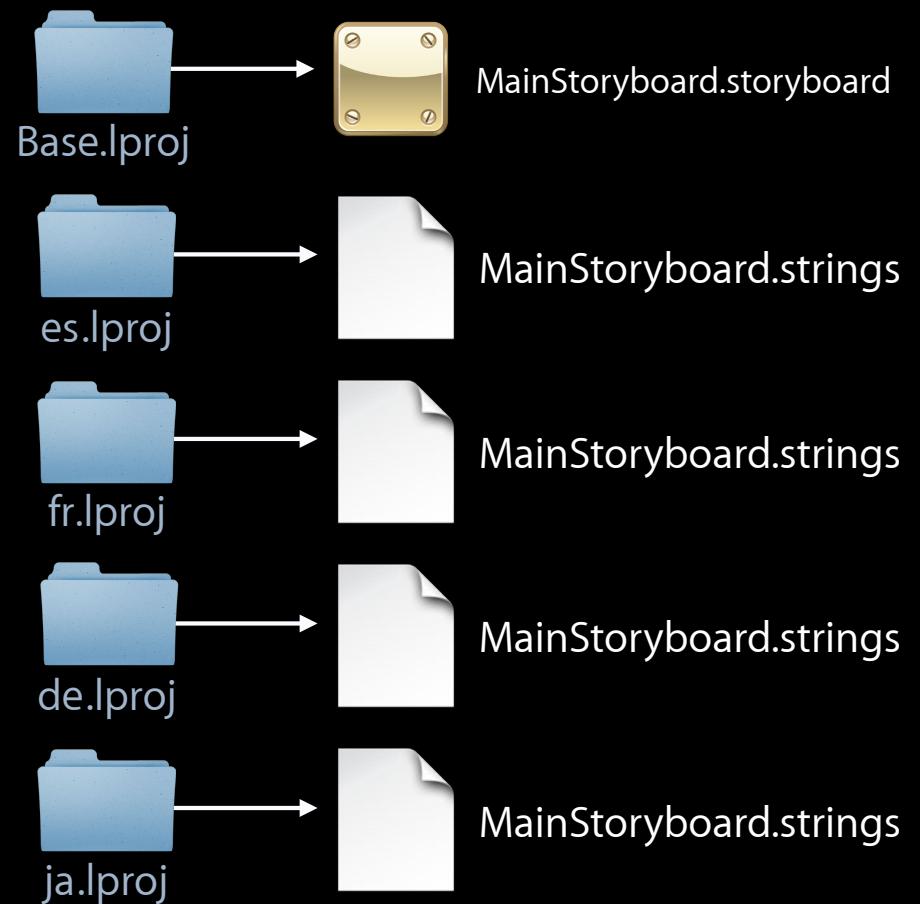
At Runtime



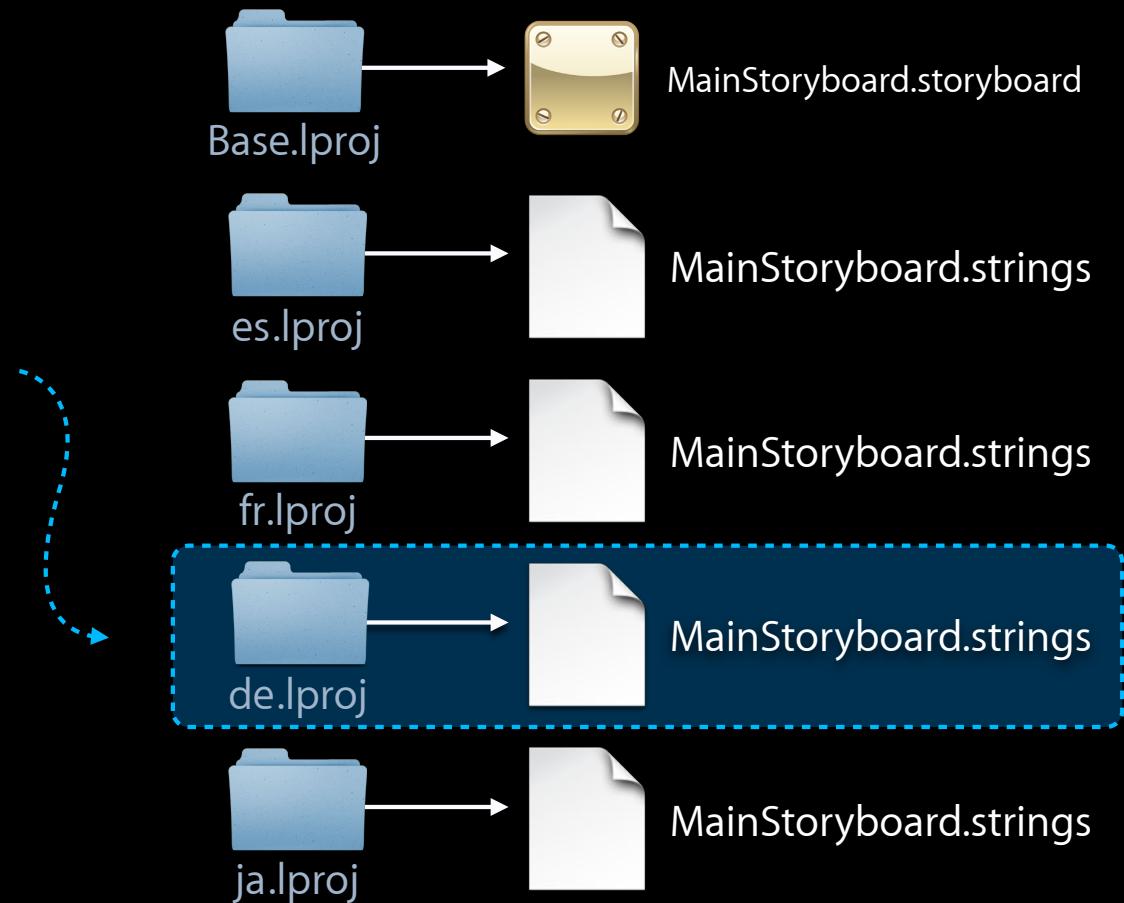
At Runtime



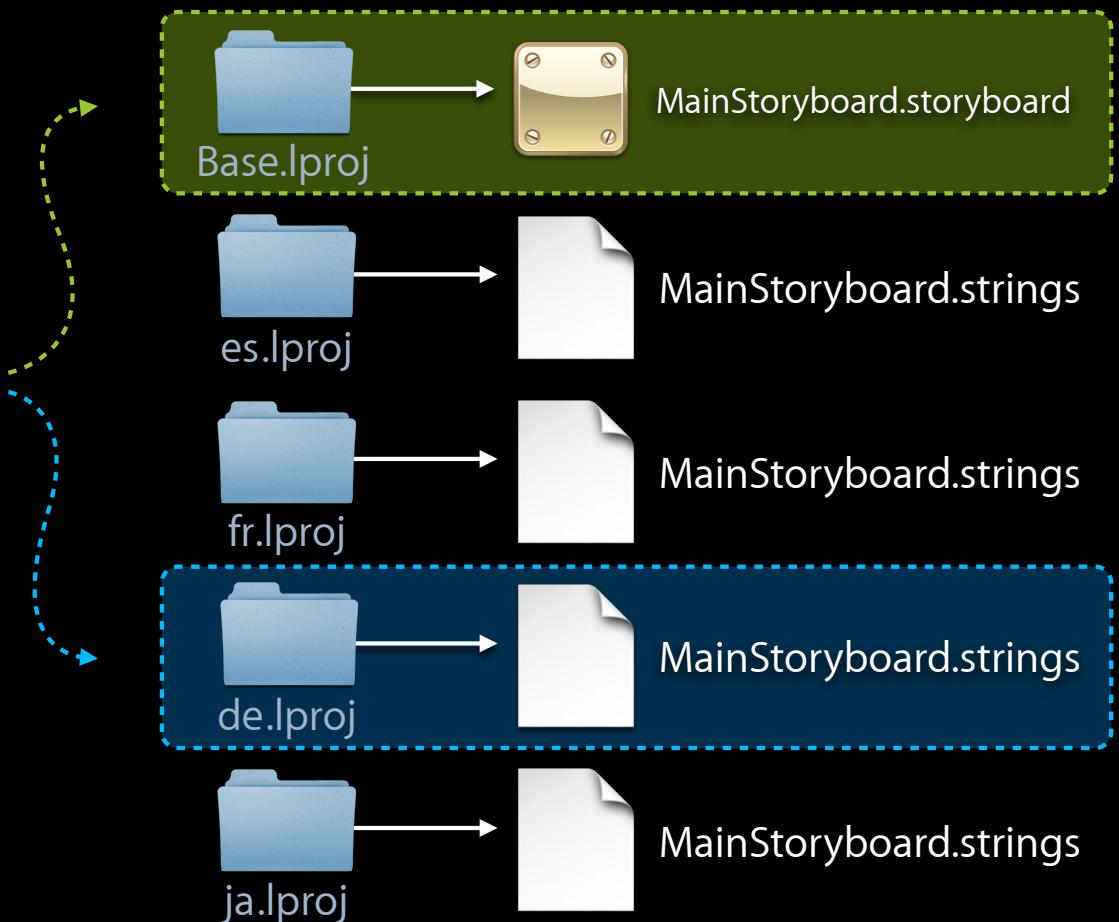
At Runtime



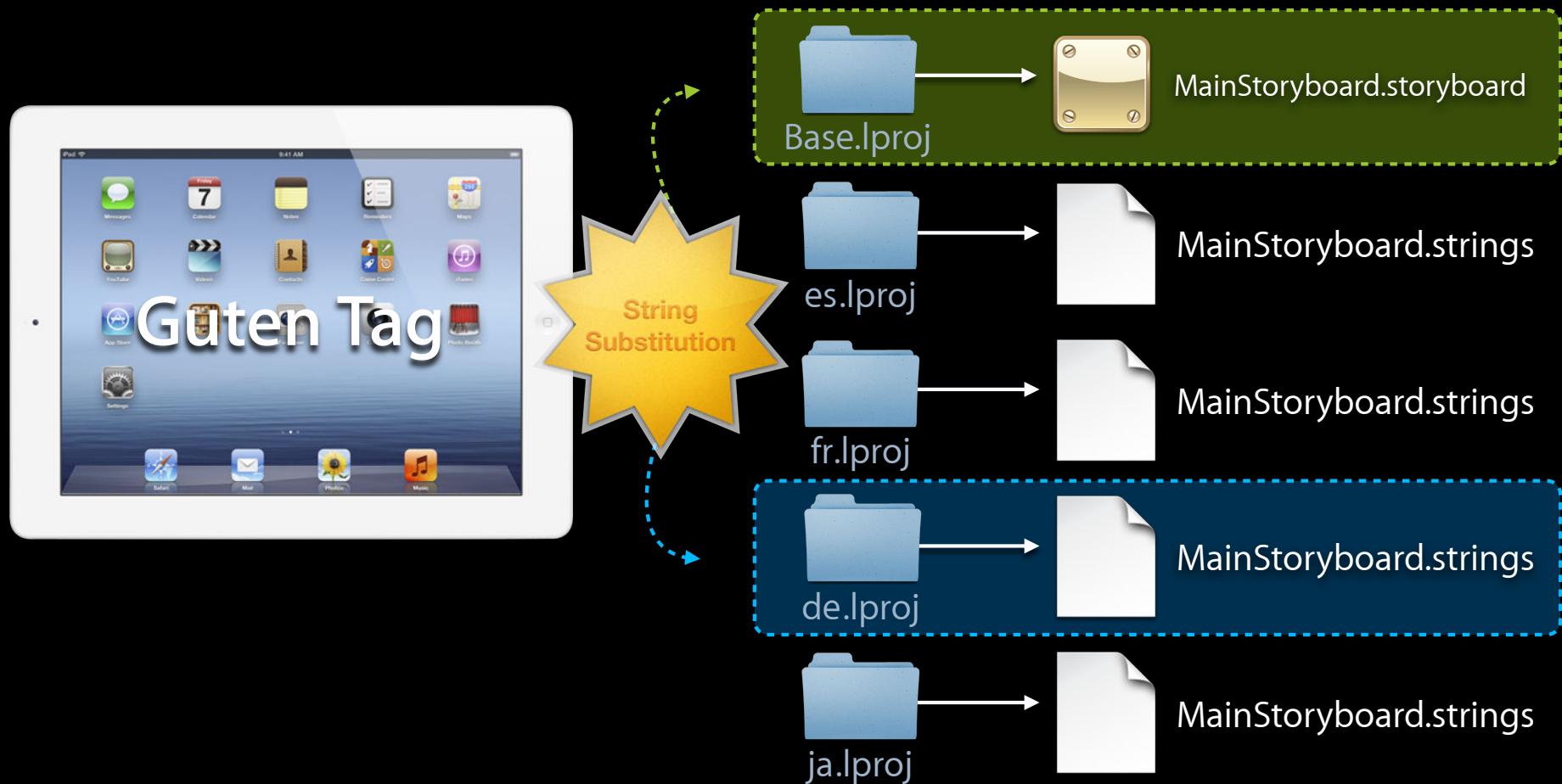
At Runtime



At Runtime



At Runtime



Demo

Localization and Auto Layout

Base Localization

Recap



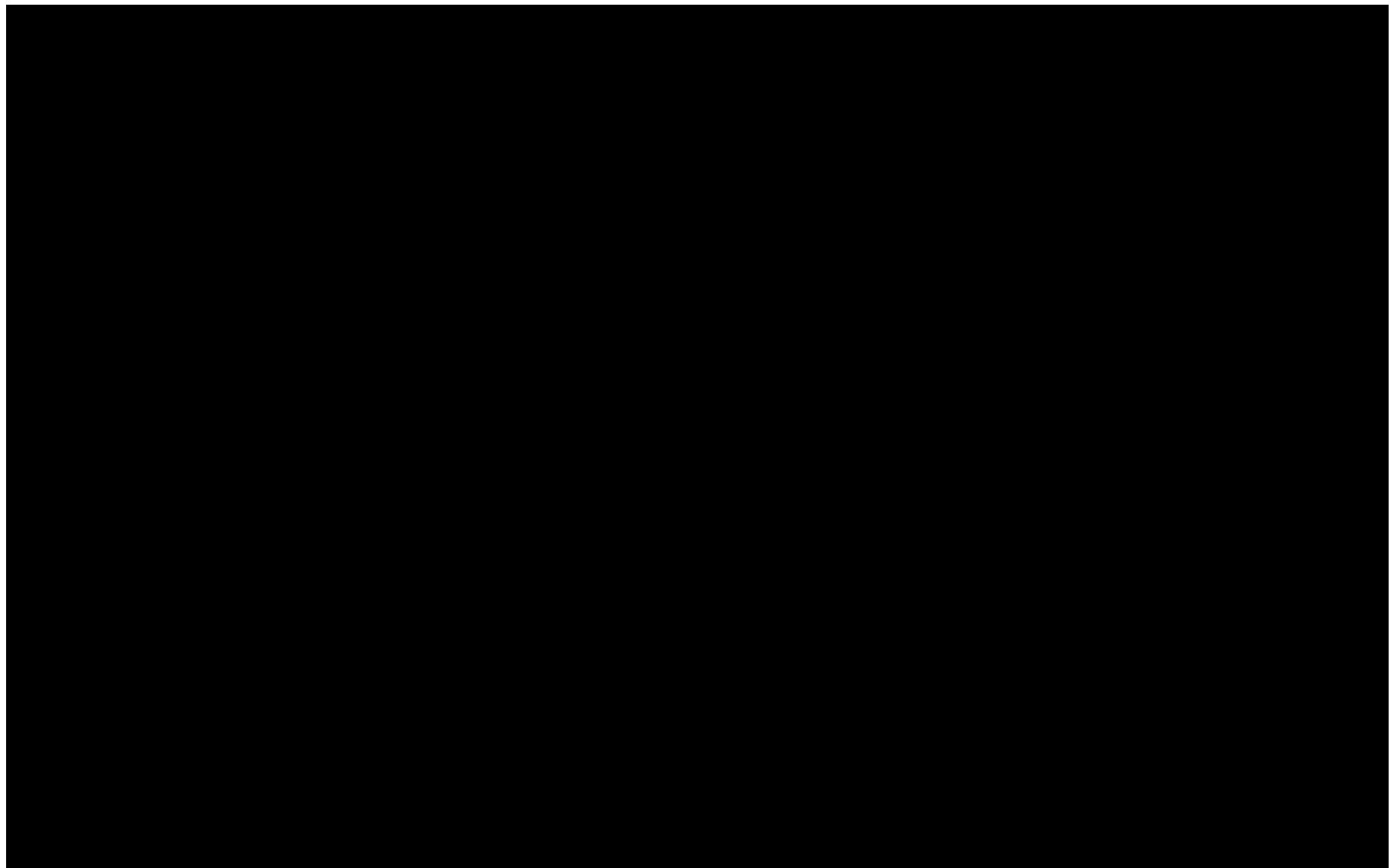
- Base is development localization
 - Not necessarily English
 - Can have English strings file
- Base contains nibs and storyboards with Auto Layout
- For other localizations, one strings file per nib or storyboard

Base Localization

Recap



- At runtime, strings from Base are substituted with strings from active language
- Can have nib or storyboard for specific localization if layout customization is required
- Availability
 - Mountain Lion and iOS 6



Takeaway:
Think ahead!

Thinking Ahead

Dynamic content

Thinking Ahead

Dynamic content

the bridge's unique design touches, including the chevrons connecting the sides of the towers and the arch elements connecting the towers to the roadway. The bridge spans the Golden Gate strait between Marin Headlands to the west and Fort Point. The Bethlehem Steel Company provided the structural steel and the construction management for the bridge. There are over 27,000 strands of wire, equal to about 100 miles of wire, used throughout the bridge. The Golden Gate Bridge cost \$35 million (in 1937 dollars of course!).

LENGTH 8,981 ft (2,737.4 m), about 2 km

WIDTH 90 ft (27.4 m)

HEIGHT 746 ft (227.4 m)

LONGEST SPAN 4,200 ft (1,280.2 m)

HEIGHT 210 ft (64 m)

AREA 1.7 acres (0.69 ha)

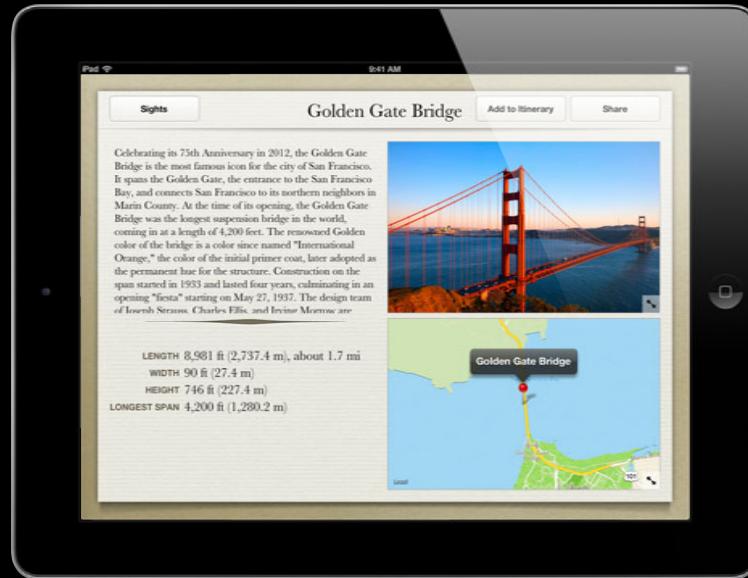
BUILT 1933

ARCHITECT Brown, Arthur Jr

ARCHITECTURAL STYLE Art Deco

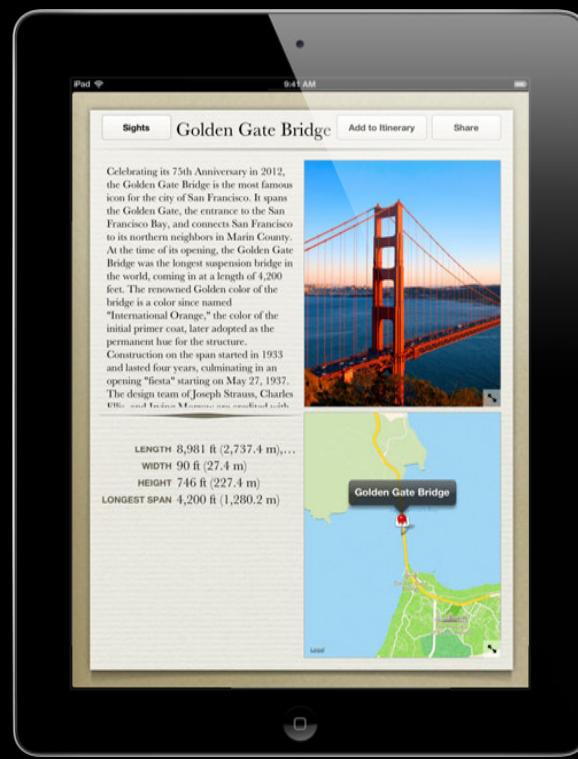
Think Ahead

Device rotation



Think Ahead

Device rotation



Think Ahead Localization

Think Ahead Localization

Golden Gate Bridge

Celebrating its 75th Anniversary in 2012, the Golden Gate Bridge is the most famous icon for the city of San Francisco. It spans the Golden Gate, the entrance to the San Francisco Bay, and connects San Francisco to its northern neighbors in Marin County. At the time of its opening, the Golden Gate Bridge was the longest suspension bridge in the world, coming in at a length of 4,200 feet. The renowned Golden color of the bridge is a color since named "International Orange," the color of the initial primer coat, later adopted as the permanent hue for the structure. Construction on the span started in 1933 and lasted four years, culminating in an opening "festa" starting on May 27, 1937. The design team of Joseph Strauss, Charles Ellis, and Irving Morrow are credited with many of the bridge's unique design touches, including the chevrons connecting the sides of the towers and the arch element that frames Fort Point. The Bethlehem Steel Company provided most of the material and the construction management for the entire project. There are over 27,000 strands of wire, equal to about 80,000 miles, used throughout the bridge. The Golden Gate Bridge cost \$35 million (in 1937 dollars).

LENGTH 8,981 ft (2,737.4 m), about 1.7 mi
 WIDTH 90 ft (27.4 m)
 HEIGHT 746 ft (227.4 m)
 LONGEST SPAN 4,200 ft (1,280.2 m)



Click

جسر غولدن غيت [إضافة إلى خط سير الرحلة]

الرمز الأكثر شهرة لمدينة سان فرانسيسكو، ويربط بين سان فرانسيسكو بجزئها في شمال مقاطعة مارين، في وقت افتتاحها، وكان جولدن جيت يربط أطول جسر معلق في العالم، إذ تأني في يطول 4200 متر، اللون التقليدي الشهير للجسر هو اللون منذ اسمه "أورازت البوية." ذات اللون من معطف التعبدي الإلري الذي اعتمدت في وقت لاحق كما هو دائم للبيكل البالى، في فترة بدات في عام 1933 واستمرت أربع سنوات، ويلات درونتها ذي لفظ "عبد" اعتباراً من يوم 27 مايو 1937.

وتقيد فريق التصميم من شتراوس يوسف، ليس شتاينر، وهو روبرت إبرهارت مع العديد من المسئات الجسر تم تصميم قرير من نوعه، بما في ذلك عالمي الافتراض الذي يربط بين الجانبين من الأبراج والعنصري قوس تلك النقطة فوق إلخارات. قدمت شركة الحديد والصلب بيت لحم 27000 معلم المواد وإدارة إنشاءات المشروع بذلك، هناك أكثر من 70,000 جداريان من الأسلوب، أي ما يعادل حوالي 80,000 كيلومتر، وتستخدم في جميع أنحاء الجسر، يكفي جسر البوية الذهبية 35 مليون دولار في عام 1937 دولاراً مطعماً.

طول 2,737.4 متراً (حوالى 2,73 كيلومتر)
 عرض 27.4 متراً
 ارتفاع 227.4 متراً
 أطول فندة 1,280.2 متراً



Click to play movie

Think Ahead

Helpful debugging defaults

- Double all localized strings

NSDoubleLocalizedStrings YES

- Simulate right to left

AppleTextDirection YES

NSForceRightToLeftWritingDirection YES

- Draw view alignment rects

NSViewShowAlignmentRects YES

UIViewShowAlignmentRects YES

Transitioning to Auto Layout

Explicit Widths

Localization

New API

Animation

Transitioning to Auto Layout

Explicit Widths

Localization

New API

Animation

Transitioning to Auto Layout

Explicit Widths

Localization

New API

Animation

Priorities

- Priorities are for expressing required or optional behaviors
 - Content hugging and compression resistance priorities
 - User constraints (e.g. centered title... if possible)
 - Split view holding priority

Priorities

- By default, constraints are required (priority value 1000)
- If less than 1000, indicates optional constraint
- System will try to get as close as possible to fulfilling all constraints
- System has some predefined, special priorities

Demo
New API

New API

Recap

- NSSplitView
 - [NSSplitView setHoldingPriority:forSubviewAtIndex:]
- Text Wrapping
 - [UILabel setPreferredMaxLayoutWidth:]
 - [NSTextField setPreferredMaxLayoutWidth:]
- Use first layout width for preferred max layout width in IB

Transitioning to Auto Layout

Explicit Widths

Localization

New API

Animation

Transitioning to Auto Layout

Explicit Widths

Localization

New API

Animation

Transitioning to Auto Layout

Explicit Widths

Localization

New API

Animation

Demo
Animation

Animation

Recap

- Animating the constant (OS X Lion and OS X Mountain Lion)

```
[[myConstraint animator] setConstant:10.0];
```

- Animation using CoreAnimation (OS X Mountain Lion and iOS 6)

```
[NSAnimationContext runAnimationGroup:^(NSAnimationContext *ctx) {  
    [ctx setAllowsImplicitAnimation:YES];  
    ...  
    [view layoutSubtreeIfNeeded];  
} completionHandler:NULL];  
  
[UIView animateWithDuration:2.0 animations:^{  
    ...  
    [view layoutIfNeeded];  
}];
```

Transitioning to Auto Layout

Explicit Widths

Localization

New API

Animation

Transitioning to Auto Layout

Explicit Widths

Localization

New API

Animation

Transitioning to Auto Layout

- Explicit Widths

- Localization

- New API

- Animation

Takeaway

Think ahead

- Dynamic content at runtime
- Device rotation or window resize
- Localization
- Use debugging defaults

More Information

Jake Behrens

UI Frameworks Evangelist

behrens@apple.com

Michael Jurewitz

Developer Tools and Performance Evangelist

jury@apple.com

Documentation

Mac OS X Human Interface Guidelines

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

Apple Developer Forums

<http://devforums.apple.com>

Related Sessions

Introduction to Auto Layout for iOS and OS X

Mission
Tuesday 10:15PM

Best Practices for Mastering Auto Layout

Mission
Thursday 9:00AM

Labs

Auto Layout Lab

App Services Lab B
Thursday 2:00PM

 WWDC2012