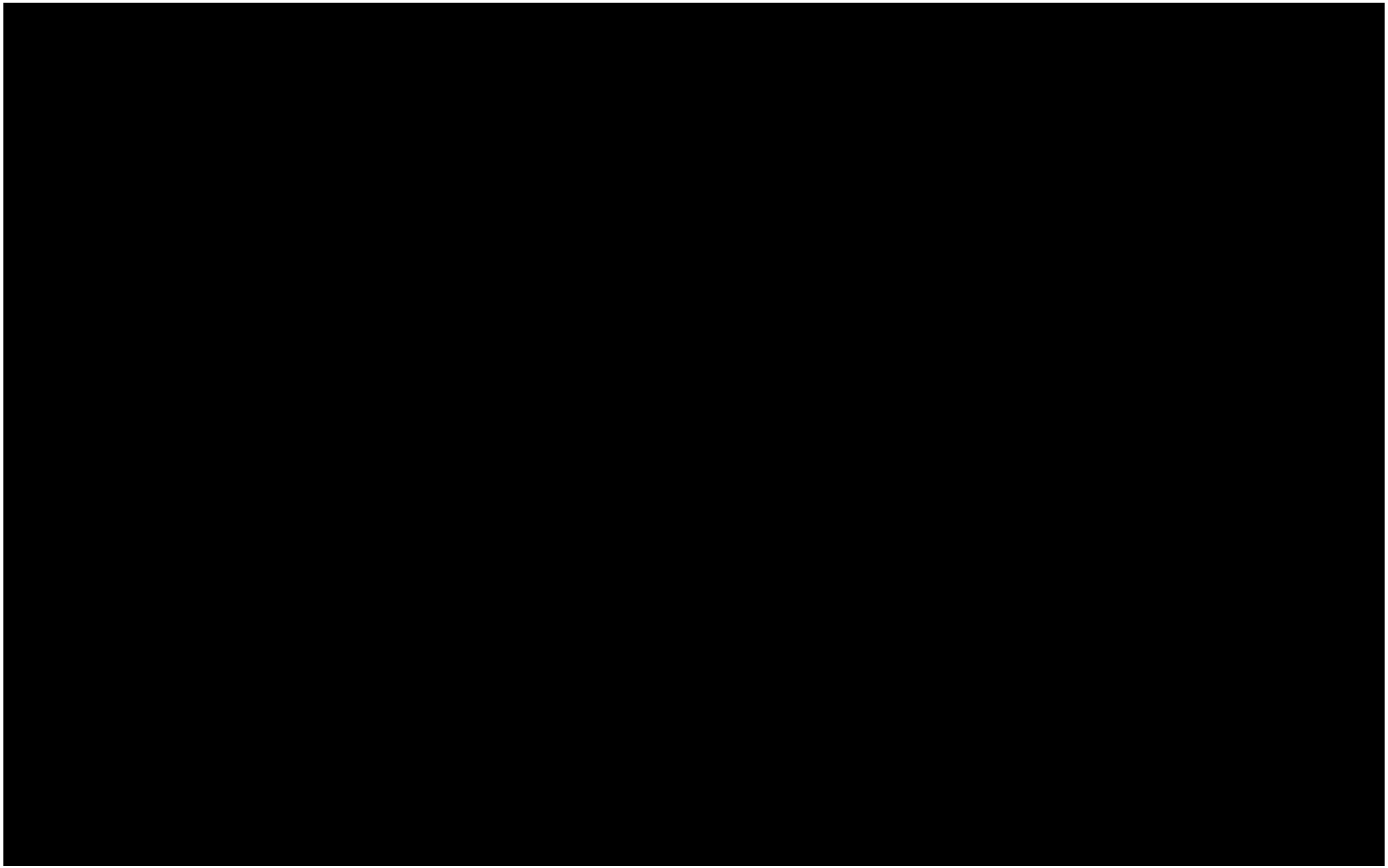
Auto Layout by Example

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

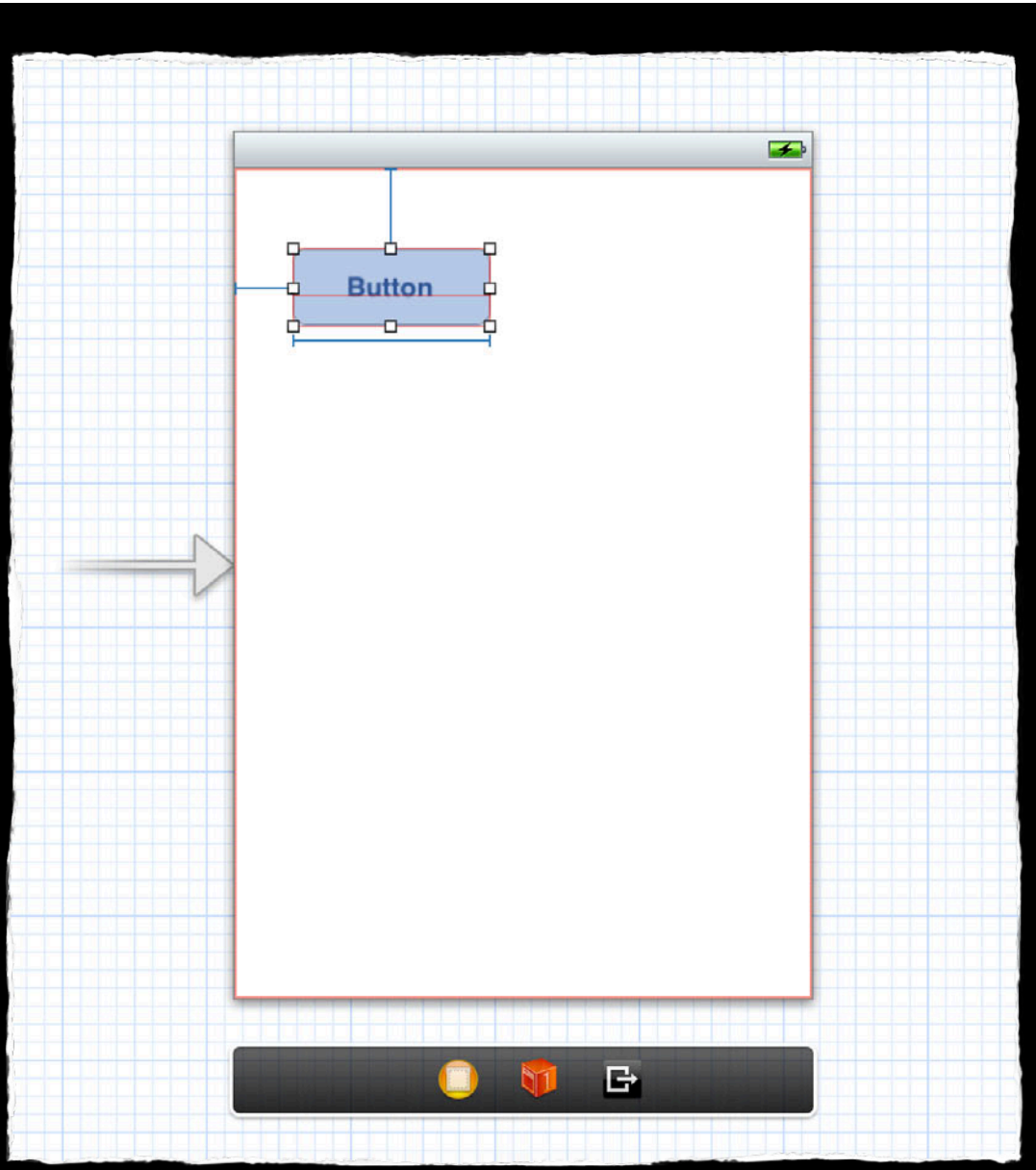
Transitioning to Auto Layout

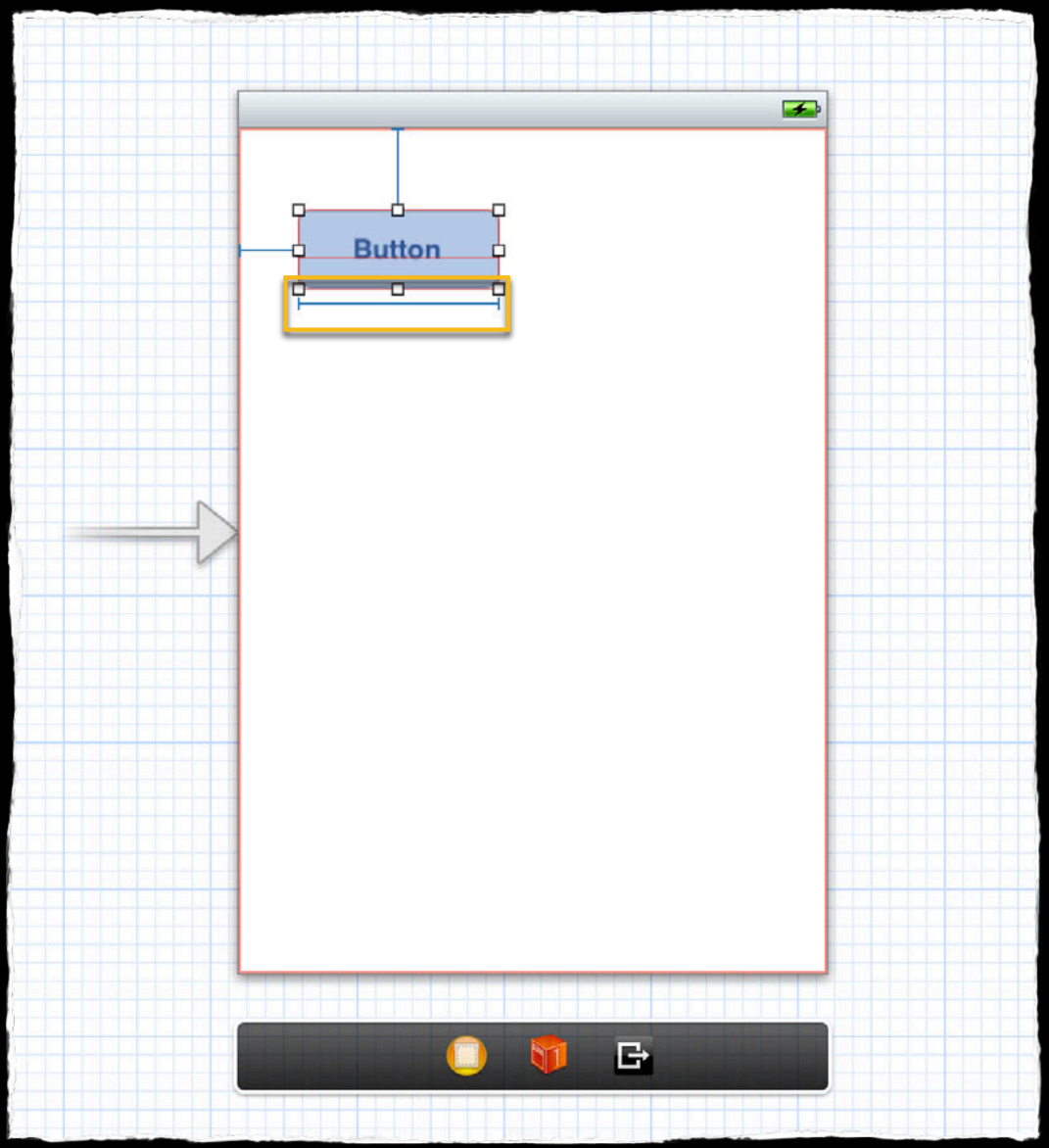
Explicit Widths

Localization

New API

Animation


Button


Button

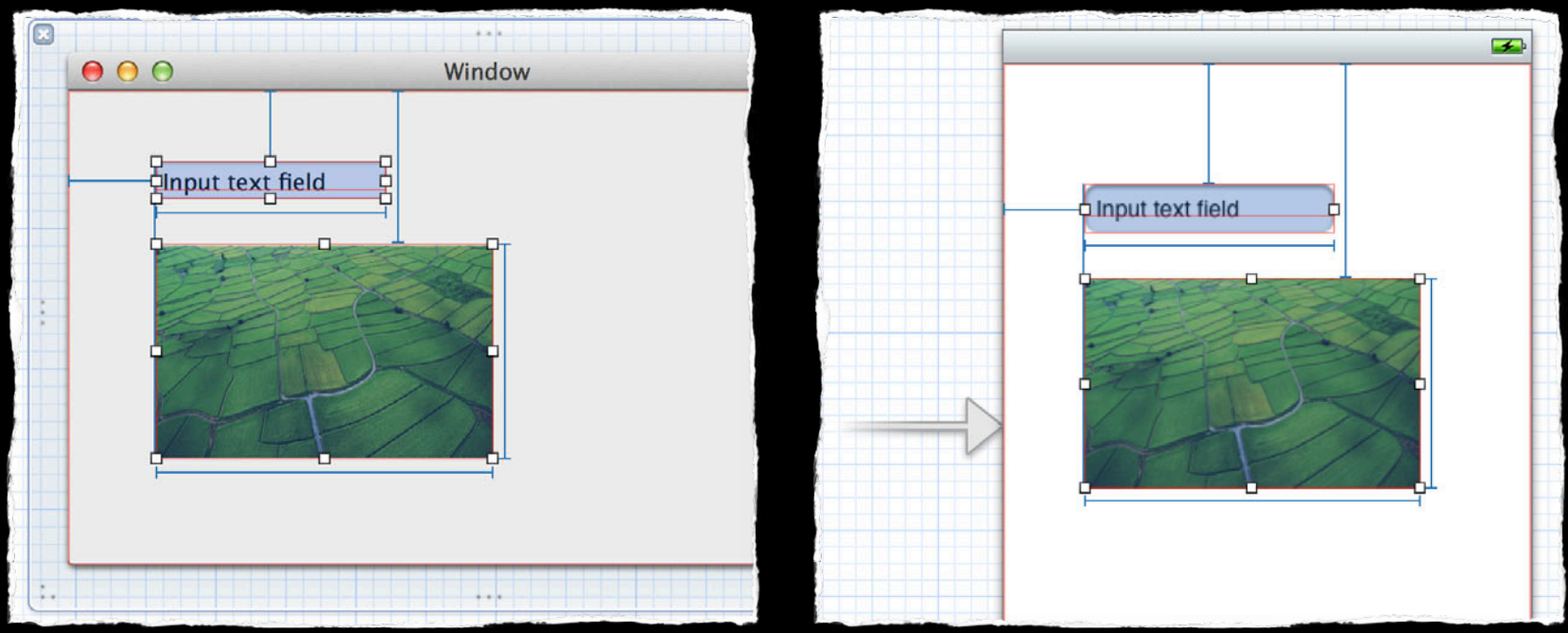
Explicit Widths

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 ≥
- 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

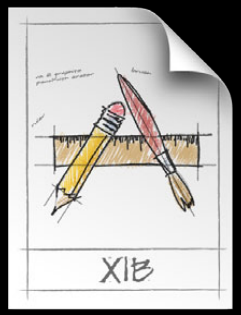
Explicit Widths

Localization

New API

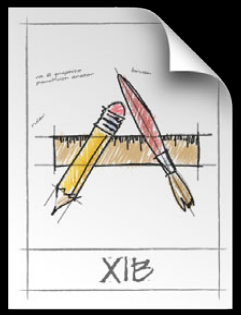
Animation

Localization Before Auto Layout



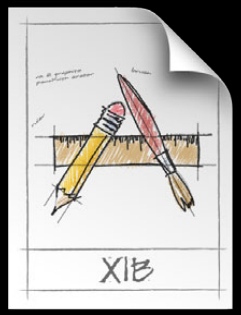
Development Region
(English)

Localization Before Auto Layout



Development Region
(English)

Localization Before Auto Layout



Development Region
(English)

Engineering

Localized Nibs

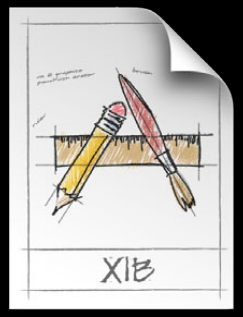
Spanish

French

German

Japanese

Localization Before Auto Layout



Development Region
(English)

Engineering

Localized Nibs

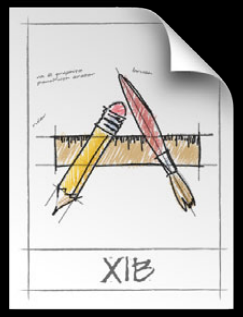
Spanish

French

German

Japanese

Localization Before Auto Layout



Development Region
(English)

Engineering

Localized Nibs

Spanish

French

German

Japanese

Localization Before Auto Layout

Development Region
(English)

Modified Development Region
(English)

Engineering

Localized Nibs

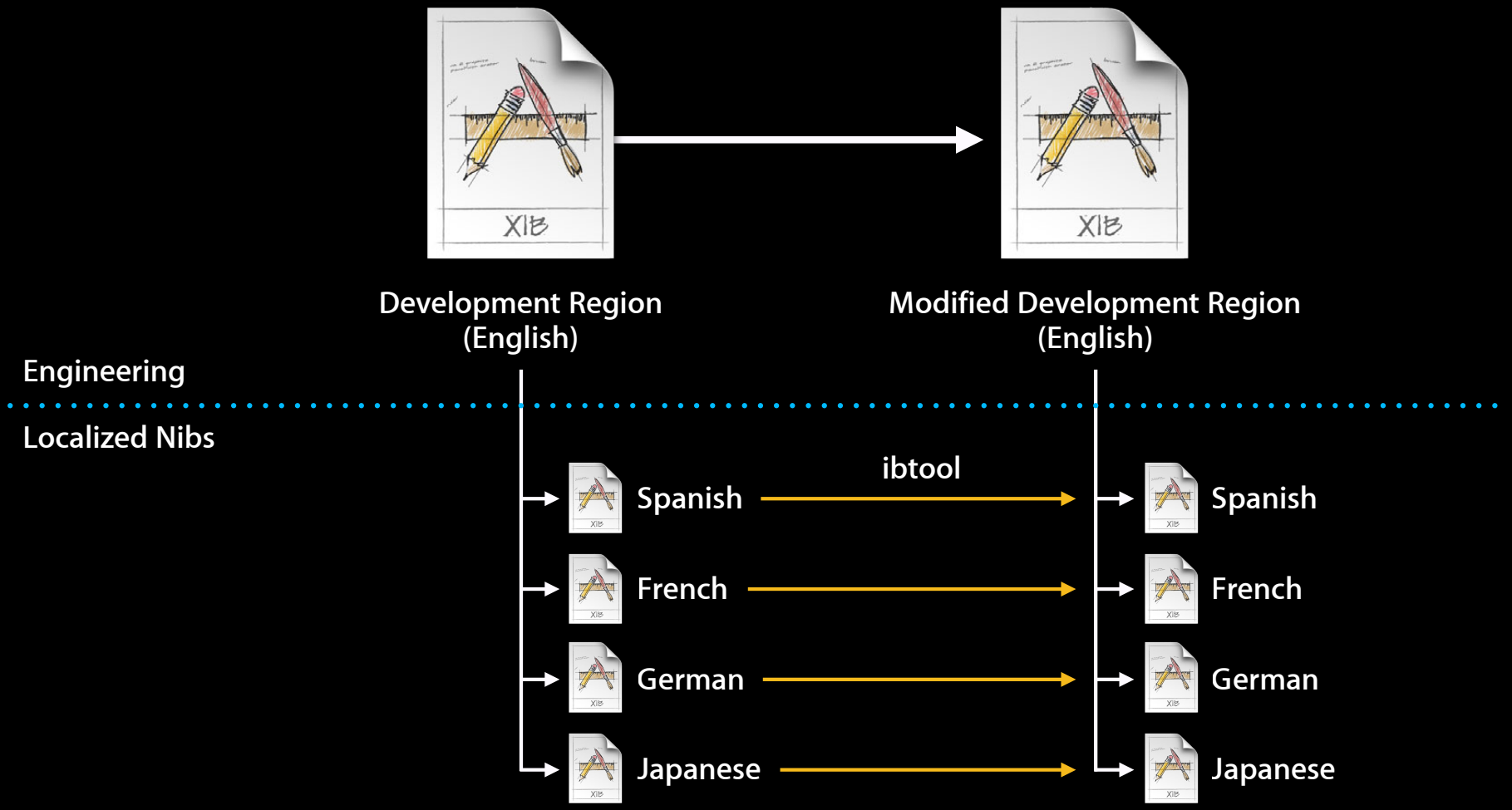
Spanish

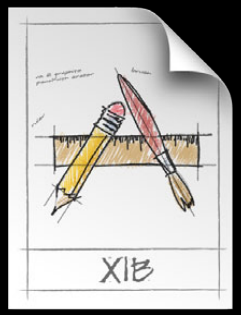
French

German

Japanese

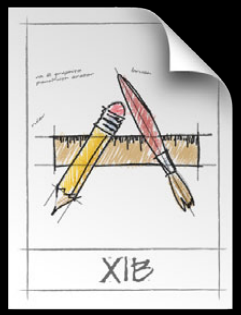
Localization Before Auto Layout





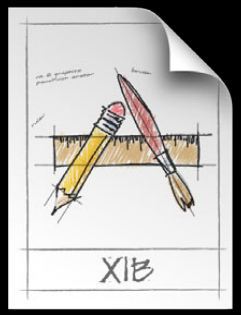
Development Region
(English)

Localization with Auto Layout



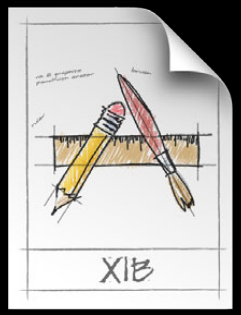
Development Region
(English)

Localization with Auto Layout



Development Region
(English)

Localization with Auto Layout



Development Region
(English)

Engineering

Localized Strings

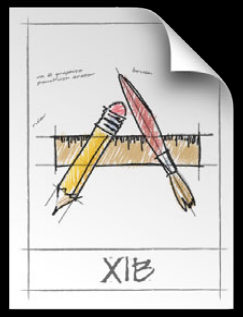
Spanish

French

German

Japanese

Localization with Auto Layout



Development Region
(English)

Engineering

Localized Strings

Spanish

French

German

Japanese

Localization with Auto Layout

Development Region
(English)

Modified Development Region
(English)

Engineering

Localized Strings

Spanish

French

German

Japanese

Localization with Auto Layout

Development Region
(English)

Modified Development Region
(English)

Engineering

Localized Strings

- Spanish
- French
- German
- Japanese

New strings might
not be needed

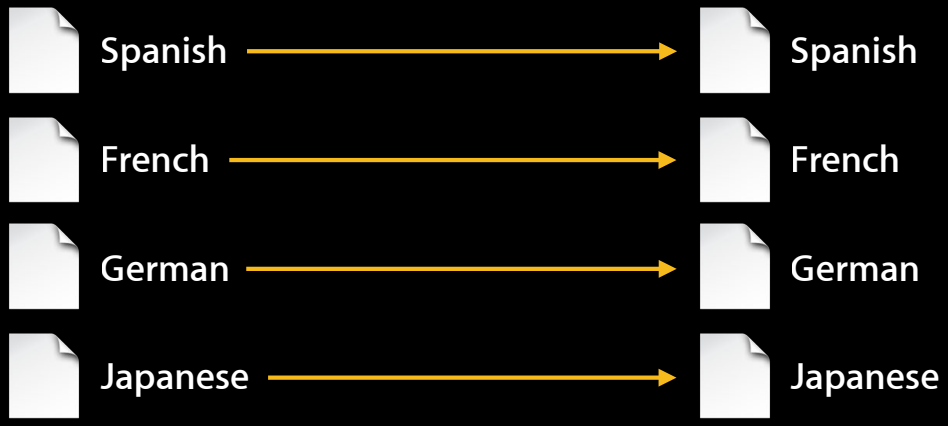
Localization with Auto Layout

Development Region
(English)

Modified Development Region
(English)

Engineering

Localized Strings

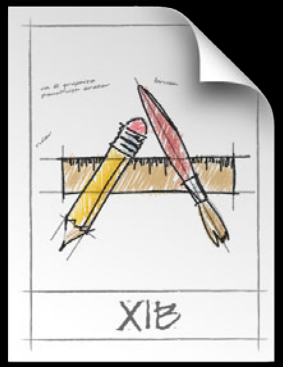


How Does That Work?

Localization and Auto Layout at runtime

How Does That Work?

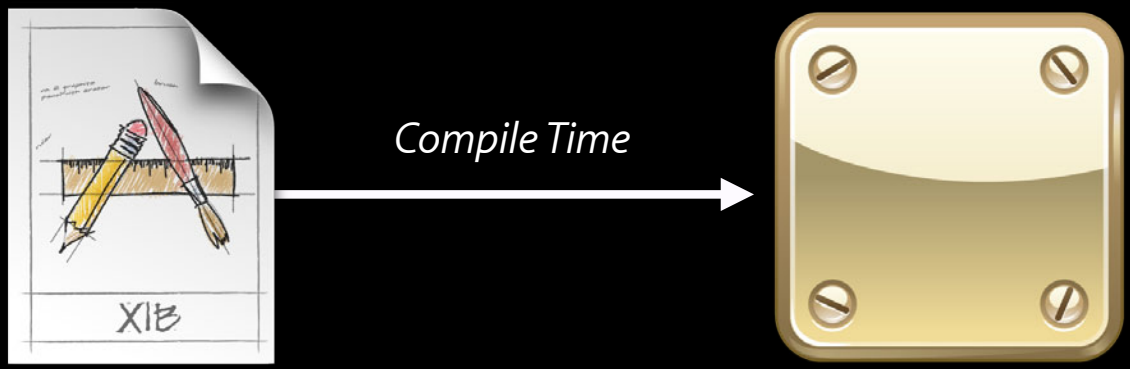
Localization and Auto Layout at runtime



MainMenu.xib

How Does That Work?

Localization and Auto Layout at runtime



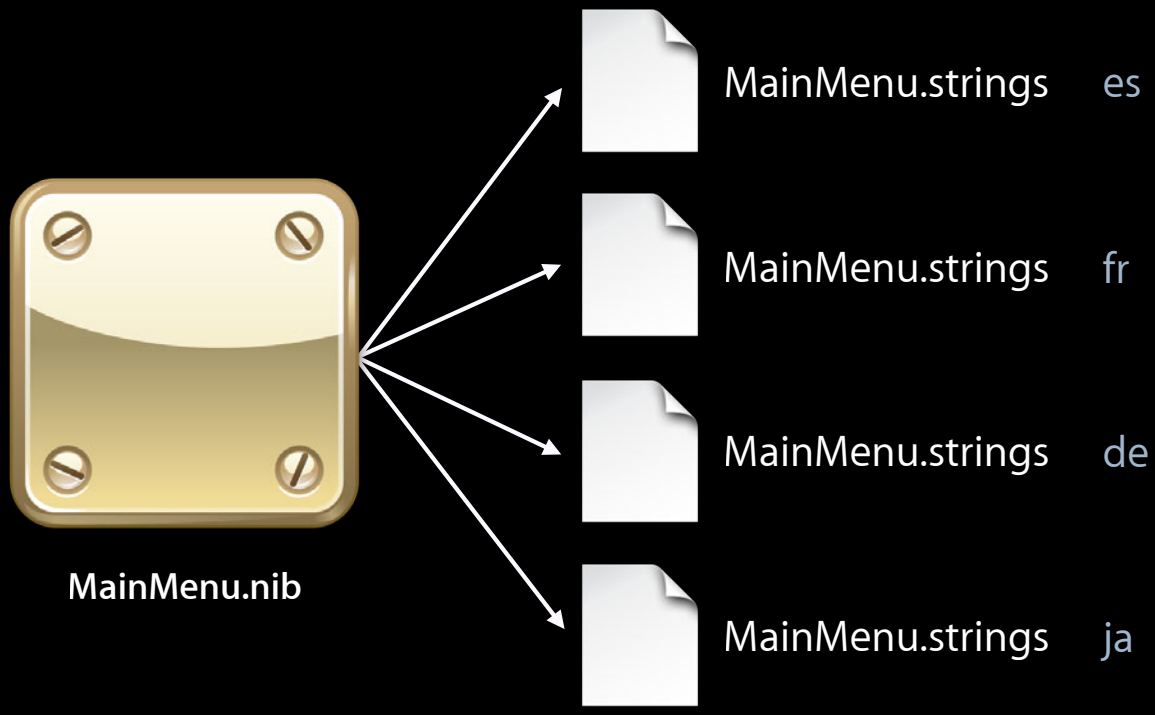
How Does That Work?

Localization and Auto Layout at runtime

MainMenu.nib

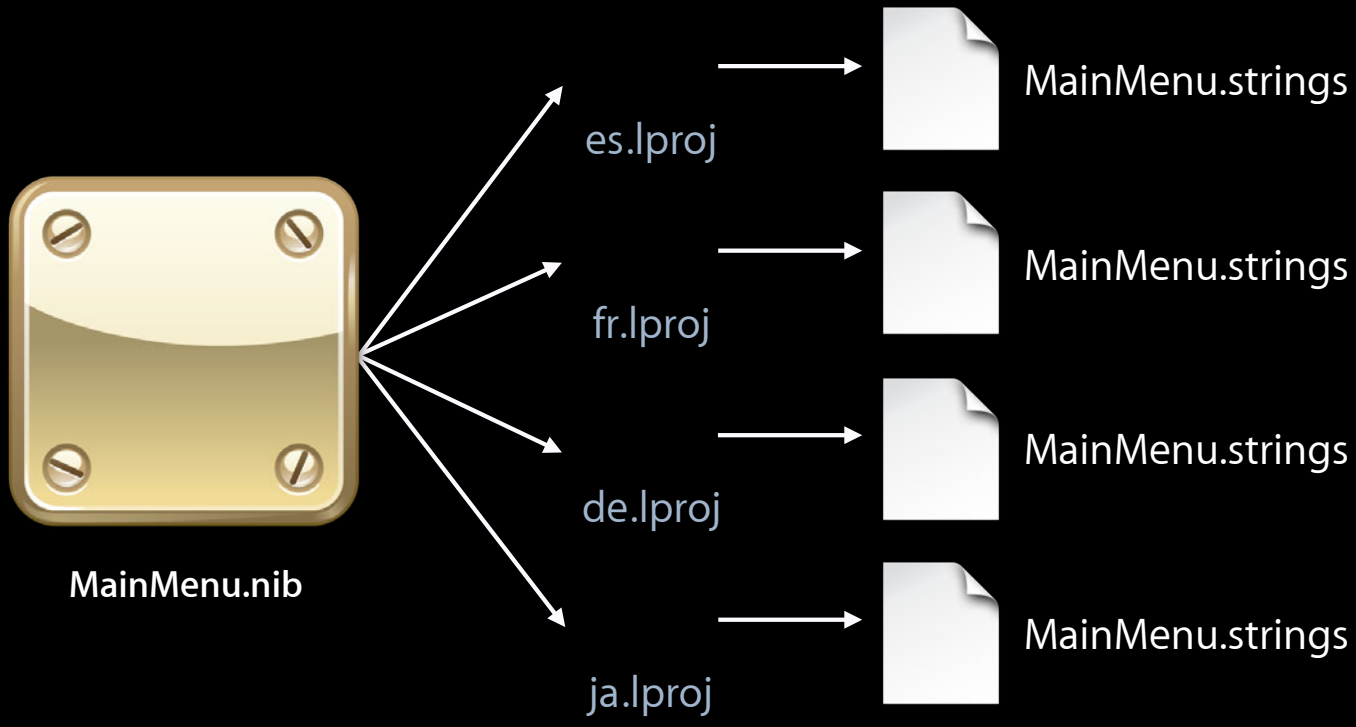
How Does That Work?

Localization and Auto Layout at runtime



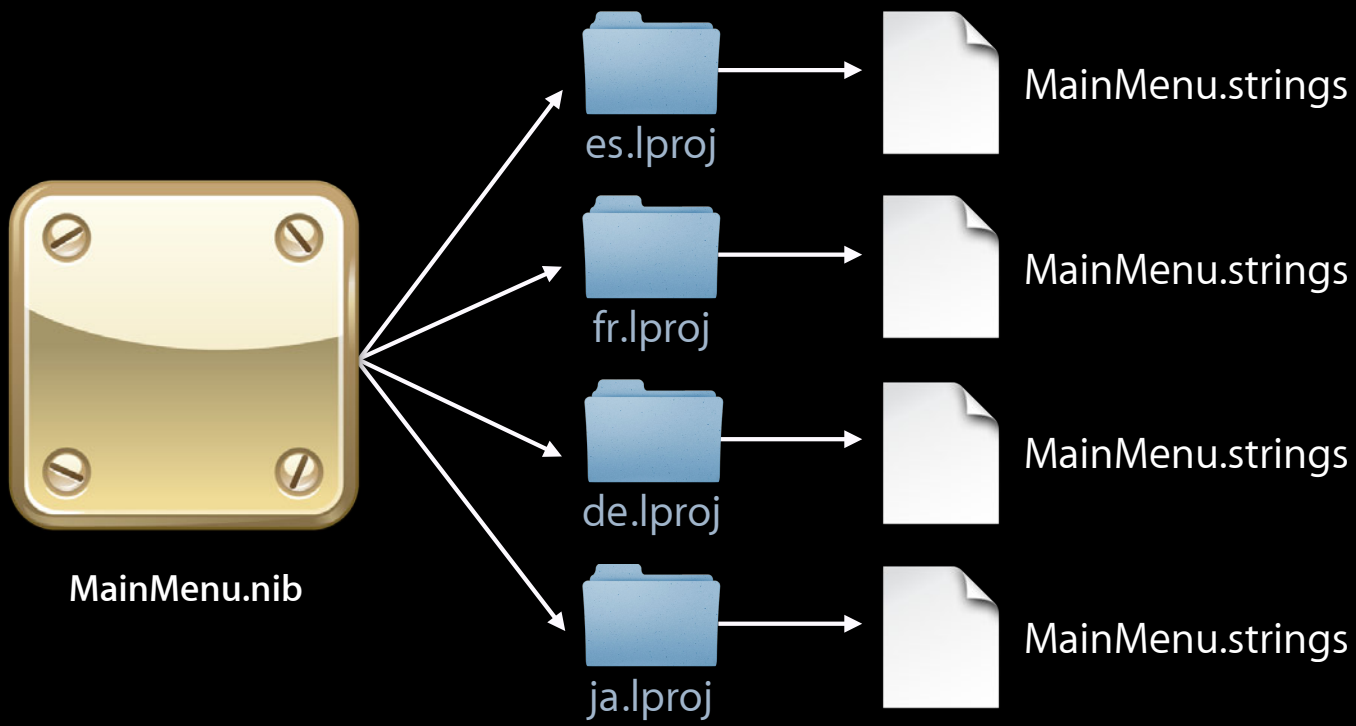
How Does That Work?

Localization and auto layout at runtime



How Does That Work?

Localization and auto layout at runtime



What About MainMenu.nib?

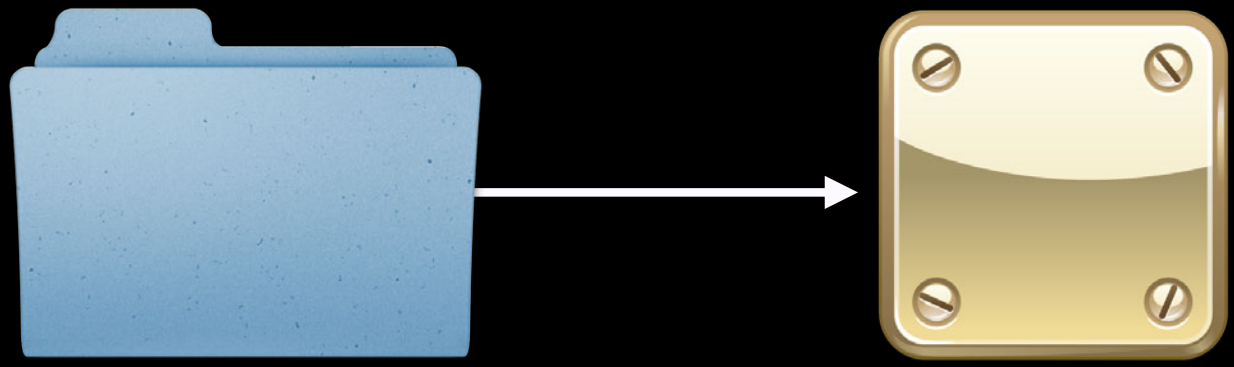
The base localization

Base.lproj

MainMenu.nib

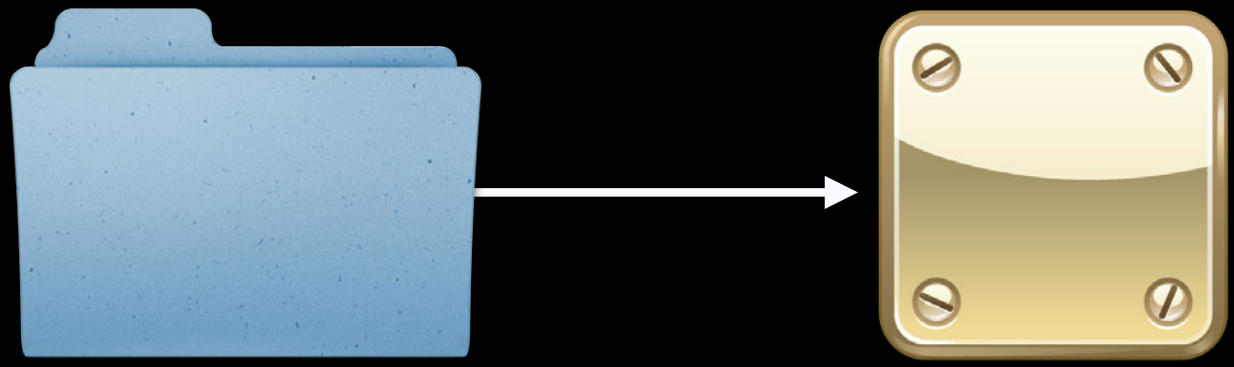
What About MainMenu.nib?

The base localization



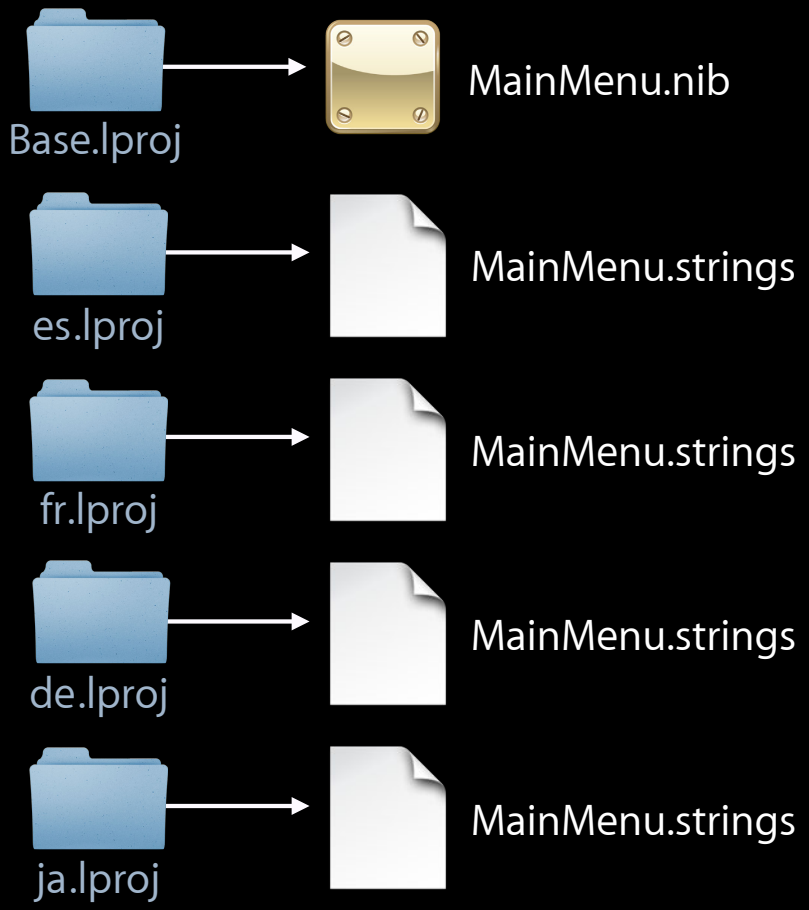
What About MainMenu.nib?

The base localization



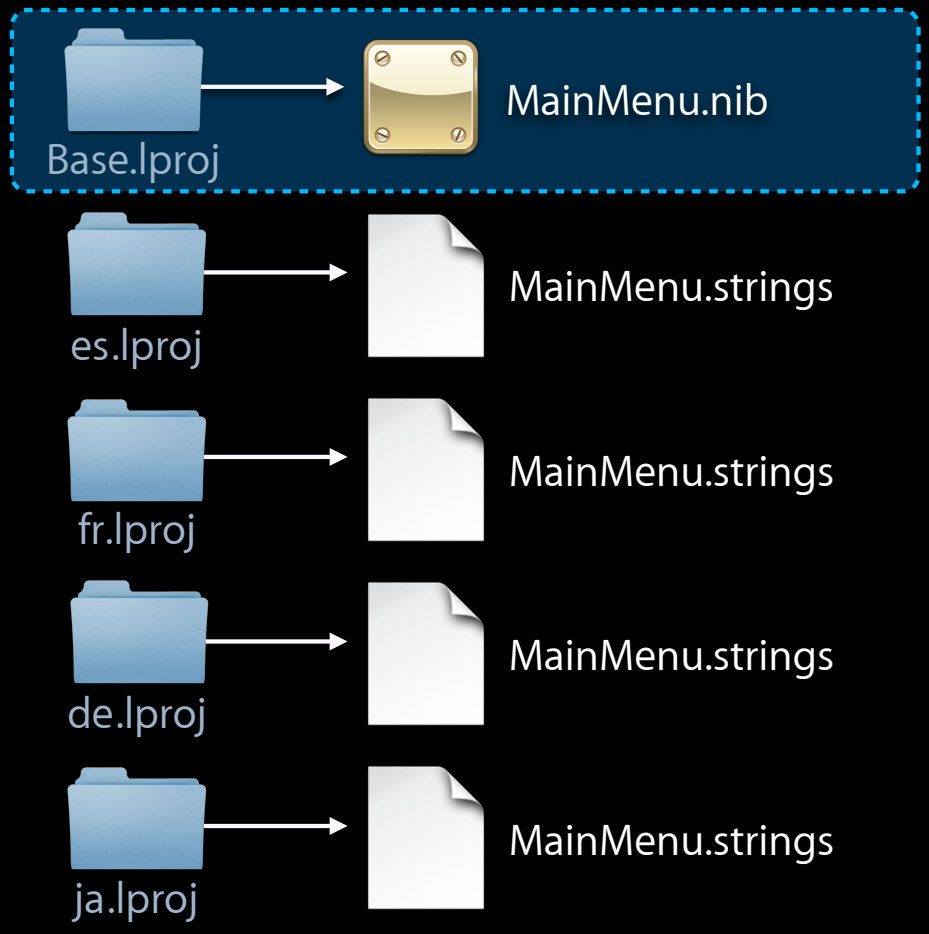
At Runtime

Hello


Base.lproj
MainMenu.nib
es.lproj
MainMenu.strings
fr.lproj
MainMenu.strings
de.lproj
MainMenu.strings
ja.lproj
MainMenu.strings

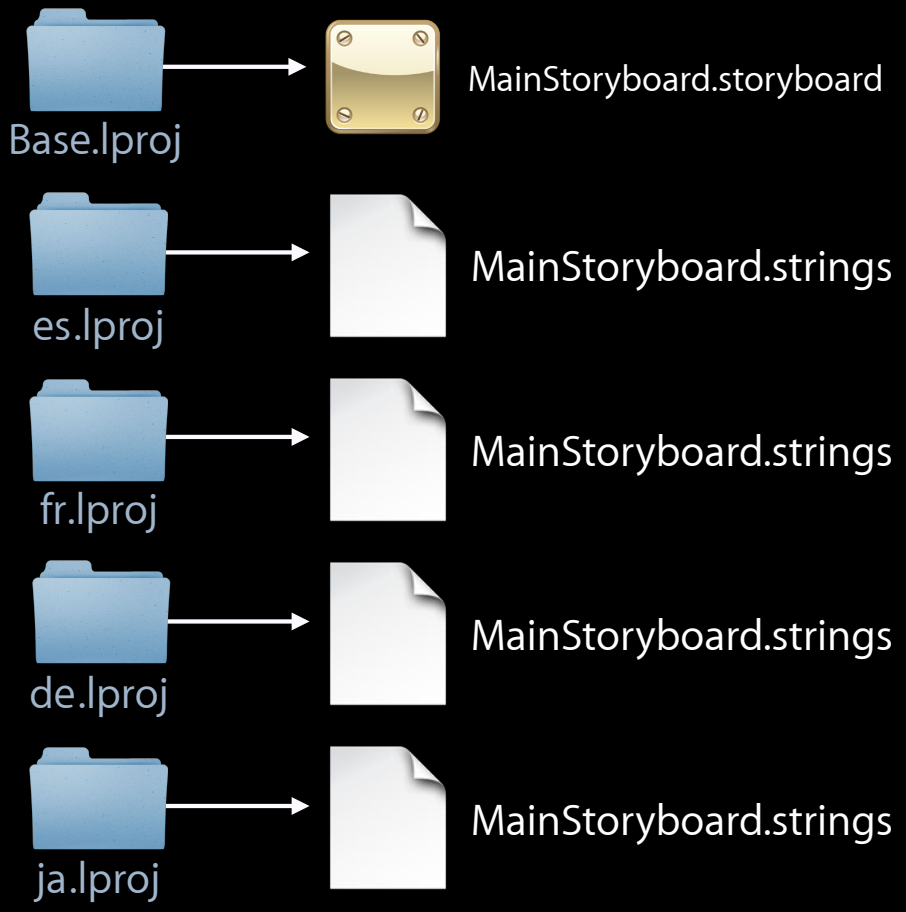
At Runtime

Hello


Base.lproj
MainMenu.nib
es.lproj
MainMenu.strings
fr.lproj
MainMenu.strings
de.lproj
MainMenu.strings
ja.lproj
MainMenu.strings

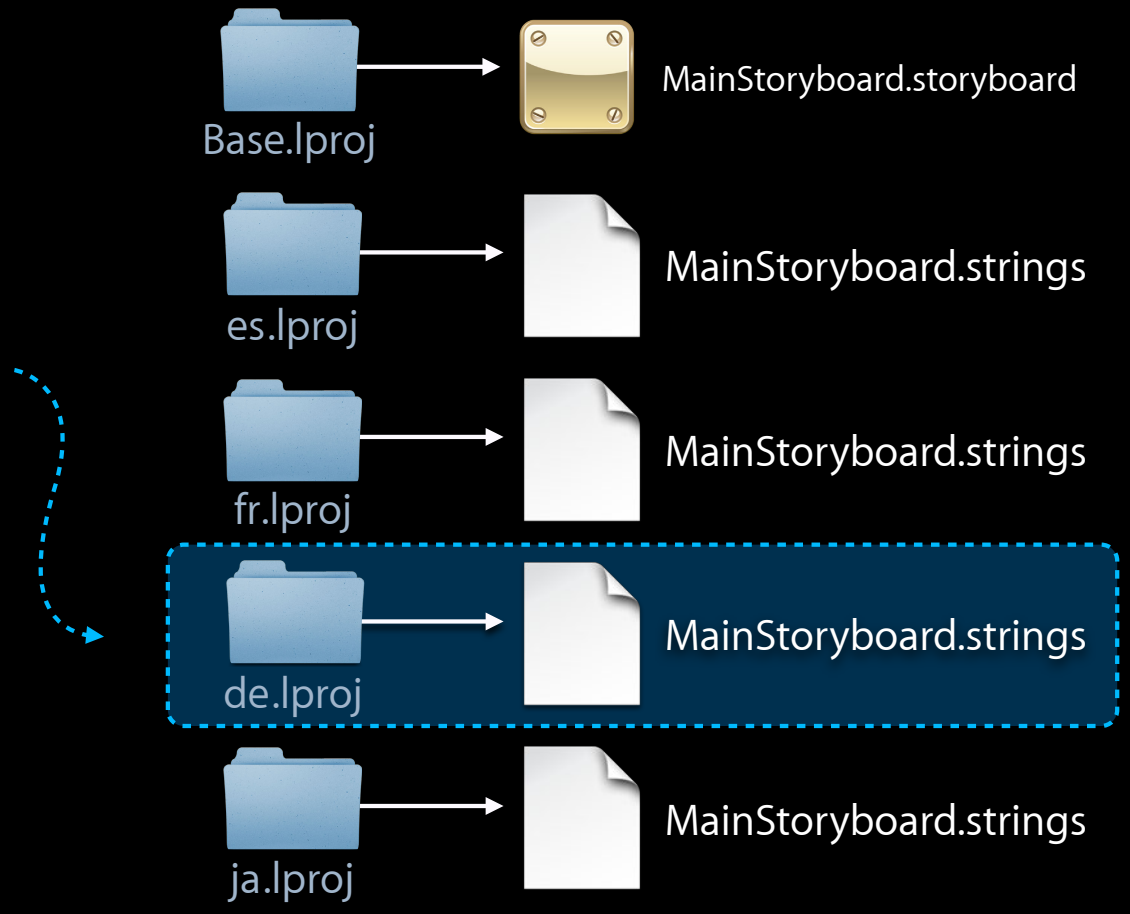
At Runtime

Guten Tag


Base.lproj
MainStoryboard.storyboard
es.lproj
MainStoryboard.strings
fr.lproj
MainStoryboard.strings
de.lproj
MainStoryboard.strings
ja.lproj
MainStoryboard.strings

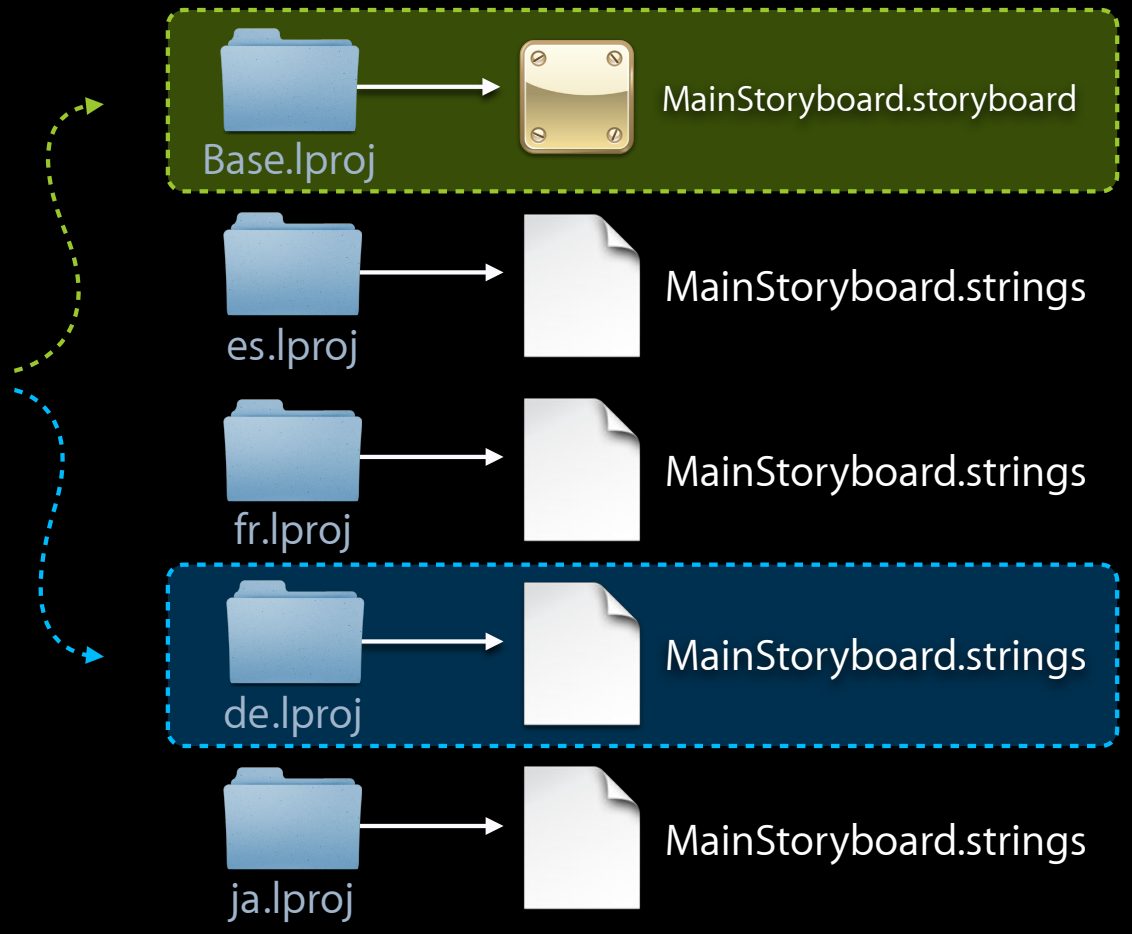
At Runtime

Guten Tag

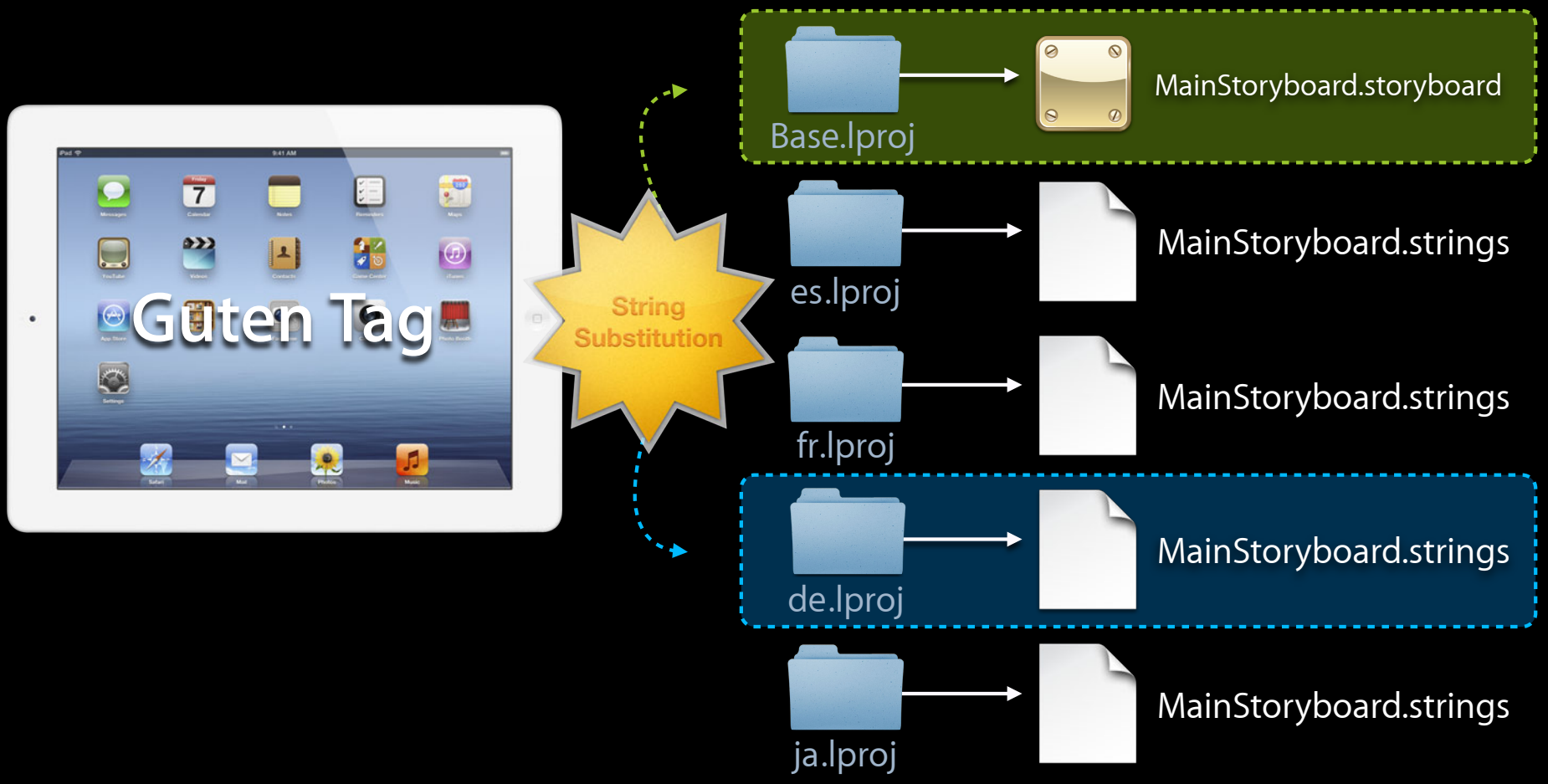

Base.lproj
MainStoryboard.storyboard
es.lproj
MainStoryboard.strings
fr.lproj
MainStoryboard.strings
de.lproj
MainStoryboard.strings
ja.lproj
MainStoryboard.strings

At Runtime

Güten Tag


Base.lproj
MainStoryboard.storyboard
es.lproj
MainStoryboard.strings
fr.lproj
MainStoryboard.strings
de.lproj
MainStoryboard.strings
ja.lproj
MainStoryboard.strings

At Runtime


Guten Tag
String Substitution
Base.lproj
MainStoryboard.storyboard
es.lproj
MainStoryboard.strings
fr.lproj
MainStoryboard.strings
de.lproj
MainStoryboard.strings
ja.lproj
MainStoryboard.strings

Demo

Localization and Auto Layout

Base Localization

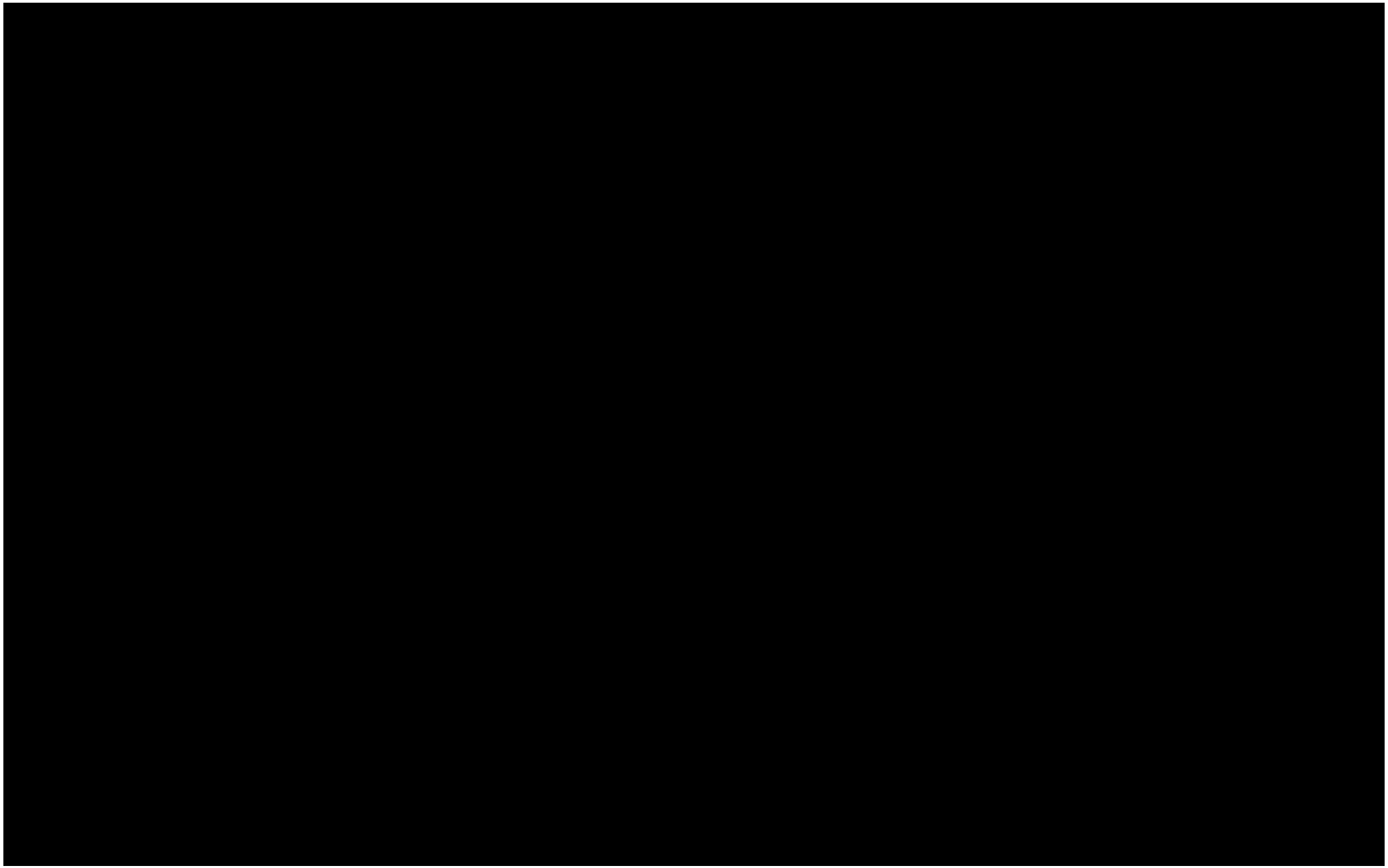
Recap

- 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



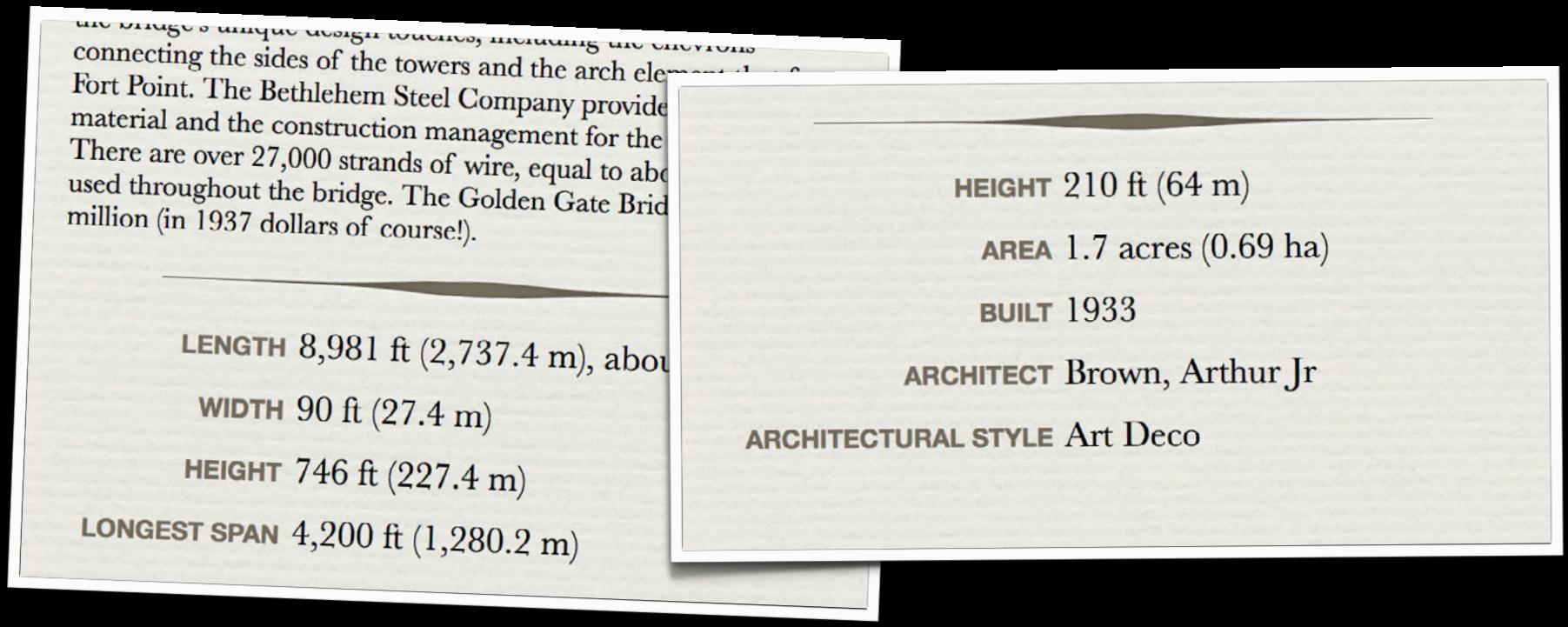
Takeaway:
Think ahead!

Thinking Ahead

Dynamic content

Thinking Ahead

Dynamic content

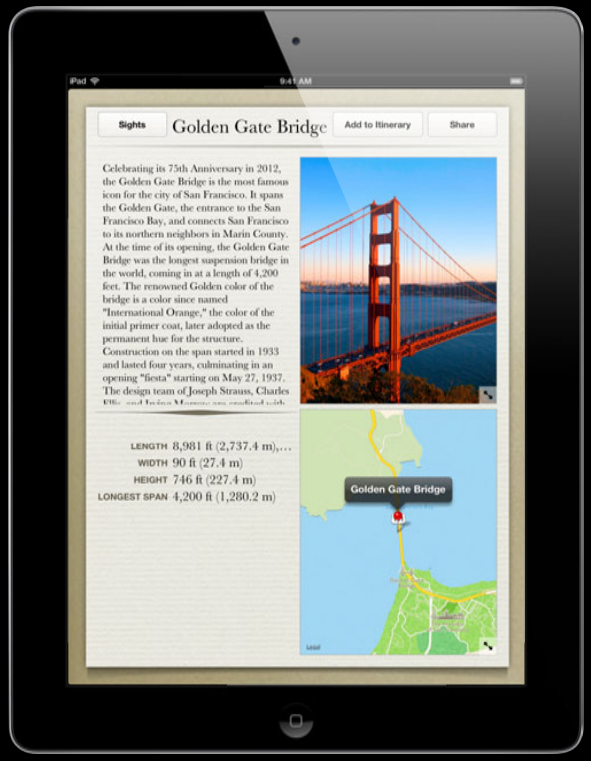


Think Ahead

Device rotation

Think Ahead

Device rotation



Think Ahead

Localization

Think Ahead

Localization

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