Mysteries of Auto Layout, Part 2

Session 219

Jesse Donaldson AppKit Engineer
Kasia Wawer iOS Keyboards Engineer
The Mysteries of Auto Layout

Part 1 — Morning
• Maintainable Layouts
• Changing Constraints
• View Sizing
• Self-Sizing Table View Cells
• Priorities
• Alignment

Part 2 — Afternoon
• The Layout Cycle
• Legacy Layout
• Constraint Creation
• Constraining Negative Space
• Unsatisfiable Constraints
• Resolving Ambiguity
The Mysteries of Auto Layout

Part 1
• Maintainable Layouts
• Changing Constraints
• View Sizing
• Self-Sizing Table View Cells
• Priorities
• Alignment

Part 2 — Afternoon
• The Layout Cycle
• Legacy Layout
• Constraint Creation
• Constraining Negative Space
• Unsatisfiable Constraints
• Resolving Ambiguity
The Layout Cycle

Mystery #7
Inside the Black Box

- Views
- Constraints
- Priorities
- intrinsicContentSize
Inside the Black Box

Views
Constraints
Priorities
intrinsicContentSize

Layout Engine
Inside the Black Box

- Views
- Constraints
- Priorities
- intrinsicContentSize

Layout Engine

Your Layout
Inside the Black Box

Layout Engine
The Layout Cycle

- Constraints Change
- Deferred Layout Pass
- Application Run Loop
The Layout Cycle

Constraints Change

Deferred Layout Pass

Application Run Loop

Options
- Show Extra Options
- Auspicious
- Brouhaha
- Chronograph
- Defenestrate
The Layout Cycle

- Constraints Change
- Deferred Layout Pass
- Application Run Loop

Options
- Show Extra Options
- Auspicious
- Brouhaha
- Chronograph
- Defenestrate
The Layout Cycle

Constraints Change → Deferred Layout Pass

Application Run Loop

Options
- Show Extra Options
- Auspicious
- Brouhaha
- Chronograph
- Defenestrate
The Layout Cycle

- Constraints Change
- Deferred Layout Pass
- Application Run Loop

Options:

- Show Extra Options
Constraint Changes

Changes to constraint expressions
• Activating or deactivating
• Setting the constant or priority
• Adding or removing views
Changes to constraint expressions

- Activating or deactivating
- Setting the constant or priority
- Adding or removing views

Engine recomputes the layout

- Engine variables receive new values
- Views call `superview.setNeedsLayout()`
Constraint Changes

Changes to constraint expressions

• Activating or deactivating
• Setting the constant or priority
• Adding or removing views

Engine recomputes the layout

• Engine variables receive new values
• Views call `superview.setNeedsLayout()`
Deferred Layout Pass

Reposition misplaced views
Deferred Layout Pass

Reposition misplaced views
Two passes through the view hierarchy
Deferred Layout Pass

Reposition misplaced views
Two passes through the view hierarchy
• Update constraints
Deferred Layout Pass

Reposition misplaced views

Two passes through the view hierarchy
  • Update constraints
  • Reassign view frames
Deferred Layout Pass

updateConstraints

Request via `setNeedsUpdateConstraints()`
Deferred Layout Pass

updateConstraints

Request via `setNeedsUpdateConstraints()`
Deferred Layout Pass

updateConstraints

Request via `setNeedsUpdateConstraints()`

Often not needed

- Initial constraints in IB
- Separate logic is harder to follow
Deferred Layout Pass

**updateConstraints**

Request via `setNeedsUpdateConstraints()`

Often not needed

- Initial constraints in IB
- Separate logic is harder to follow

Implement it when

- Changing constraints in place is too slow
- A view is making redundant changes
Deferred Layout Pass

*layoutSubviews* aka *layout*

Traverse the view hierarchy, top-down

- Call *layoutSubviews()* (or *layout()* on OS X)
Deferred Layout Pass

`layoutSubviews` aka `layout`

Traverse the view hierarchy, top-down

- Call `layoutSubviews()` (or `layout()` on OS X)
Deferred Layout Pass

`layoutSubviews` aka `layout`

Traverse the view hierarchy, top-down
- Call `layoutSubviews()` (or `layout()` on OS X)

Position the view's subviews
- Copy subview frames from the layout engine
Deferred Layout Pass

`layoutSubviews` aka `layout`

Traverse the view hierarchy, top-down
• Call `layoutSubviews()` (or `layout()` on OS X)

Position the view's subviews
• Copy subview frames from the layout engine
Deferred Layout Pass

`layoutSubviews` aka `layout`

Traverse the view hierarchy, top-down

• Call `layoutSubviews()` (or `layout()` on OS X)

Position the view's subviews

• Copy subview frames from the layout engine

Override `layoutSubviews()` for custom layout

• … but be careful!
Deferred Layout Pass

Overriding `layoutSubviews`

Override when constraints are insufficient
Deferred Layout Pass

Overriding `layoutSubviews`

Override when constraints are insufficient

Some views have already been laid out
Deferred Layout Pass

Overriding `layoutSubviews`

Override when constraints are insufficient

Some views have already been laid out

**DO**

- Invoke `super.layoutSubviews()`
- Invalidate layout within your subtree
Deferred Layout Pass

Overriding `layoutSubviews`

Override when constraints are insufficient
Some views have already been laid out

**DO**
- Invoke `super.layoutSubviews()`
- Invalidate layout within your subtree

**DON'T**
- Call `setNeedsUpdateConstraints()`
- Invalidate layout outside your subtree
- Modify constraints indiscriminately
The Layout Cycle

Remember

• Don’t expect frames to change immediately
• Proceed with caution when overriding `layoutSubviews()`
Interacting with Legacy Layout

Mystery #8
Interacting with Legacy Layout

Positioning by frame versus constraints
Interacting with Legacy Layout

Positioning by frame versus constraints

Sometimes you need to set the frame

• e.g., if you're overriding `layoutSubviews()`
Interacting with Legacy Layout

Positioning by frame versus constraints

Sometimes you need to set the frame

• e.g., if you're overriding `layoutSubviews()`

```swift
var translatesAutoresizingMaskIntoConstraints: Bool
```
Setting the frame automatically generates constraints
Setting the frame automatically generates constraints

- Set the frame with gleeful abandon!
Setting the frame automatically generates constraints
- Set the frame with gleeful abandon!
- Constraints implement the autoresizingMask
Setting the frame automatically generates constraints

• Set the frame with gleeful abandon!
• Constraints implement the autoresizingMask
• Other views can be constrained to it
Setting the frame automatically generates constraints

- Set the frame with gleeful abandon!
- Constraints implement the `autoresizingMask`
- Other views can be constrained to it

Set to false when using constraints

- Beware—defaults to true for programmatically created views
translatesAutoresizingMaskIntoConstraints

Do not forget to turn this off!

override func viewDidLoad() {
    super.viewDidLoad()

    let b = NSButton()
    b.bezelStyle = .RoundedBezelStyle

    view.addSubview(b)

    NSLayoutConstraint(item:b, attribute:.Top, relatedBy:.Equal, toItem:view, attribute:.Top, multiplier:1, constant:10).active = true

    NSLayoutConstraint(item:b, attribute:.Leading, relatedBy:.Equal, toItem:view, attribute:.Leading, multiplier:1, constant:10).active = true

}
translatesAutoresizingMaskIntoConstraints

Do not forget to turn this off!
translatesAutoresizingMaskIntoConstraints

Do not forget to turn this off!

2015-05-08 09:41:27.668 WWDC 2015[4107:226949] Unable to simultaneously satisfy constraints:

```
"<NSAutoresizingMaskLayoutConstraint:0x6100000810e0 h=---& v=---& H:-(0)-[NSButton:0x618000140160'Button'] (Names: '|':NSView:0x618000120460 )>",
"<NSLayoutConstraint:0x6180000828a0 H:|-(10)-[NSButton:0x618000140160'Button'](LTR) (Names: '|':NSView:0x618000120460 )>"
```

Will attempt to recover by breaking constraint

```
<NSLayoutConstraint:0x6000000825d0 H:-(10)-[NSButton:0x600000140c60'Button'](LTR) (Names: '|':NSView:0x6000001203c0 )>
```
translatesAutoresizingMaskIntoConstraints

Do not forget to turn this off!

2015-05-08 09:41:27.668 WWDC 2015[4107:226949] Unable to simultaneously satisfy constraints:

```
"<NSAutoresizingMaskLayoutConstraint:0x6100000810e0 h=-& v=-& H:-(0)-[NSButton:0x618000140160'Button'] (Names: '|' NSView:0x618000120460 )>",
"<NSLayoutConstraint:0x6180000828a0 H:-(10)-[NSButton:0x618000140160'Button'](LTR) (Names: '|' NSView:0x618000120460 )>"
```

Will attempt to recover by breaking constraint

```
<NSLayoutConstraint:0x6000000825d0 H:-(10)-[NSButton:0x6000000825d0'Button'](LTR) (Names: '|' NSView:0x6000000825d0 )>
```
override func viewDidLoad() {
    super.viewDidLoad()
    let b = NSButton()
    b.bezelStyle = .RoundedBezelStyle
    b.translatesAutoresizingMaskIntoConstraints = false
    view.addSubview(b)

    NSLayoutConstraint(item:b, attribute:.Top, relatedBy:.Equal, toItem:view, attribute:.Top, multiplier:1, constant:10).active = true
    NSLayoutConstraint(item:b, attribute:.Leading, relatedBy:.Equal, toItem:view, attribute:.Leading, multiplier:1, constant:10).active = true
}

translatesAutoresizingMaskIntoConstraints

Do not forget to turn this off!
translatesAutoresizingMaskIntoConstraints

Do not forget to turn this off!

override func viewDidLoad() {
    super.viewDidLoad()
    let b = NSButton()
    b.bezelStyle = .RoundedBezelStyle
    b.translatesAutoresizingMaskIntoConstraints = false
    view.addSubview(b)

    NSLayoutConstraint(item:b, attribute:.Top, relatedBy:.Equal, toItem:view, attribute:.Top, multiplier:1, constant:10).active = true
    NSLayoutConstraint(item:b, attribute:.Leading, relatedBy:.Equal, toItem:view, attribute:.Leading, multiplier:1, constant:10).active = true
}

translatesAutoSizeingMaskIntoConstraints

Remember

• Use when setting the frame directly
• Otherwise, don't forget to turn this off!
Constraint Creation
Mystery #9
override func viewDidLoad() {
    super.viewDidLoad()

    let b = NSButton()
b.bezelStyle = .RoundedBezelStyle
b.translatesAutoresizingMaskIntoConstraints = false
view.addSubview(b)

    NSLayoutConstraint(item:b, attribute:.Top, relatedBy:.Equal, toItem:view, attribute:.Top, multiplier:1, constant:10).active = true
    NSLayoutConstraint(item:b, attribute:.Leading, relatedBy:.Equal, toItem:view, attribute:.Leading, multiplier:1, constant:10).active = true
}

NSLayoutConstraint(item:b, attribute:.Top, relatedBy:.Equal, toItem:view, attribute:.Top, multiplier:1, constant:10).active = true
NSLayoutConstraint(item:b, attribute:.Leading, relatedBy:.Equal, toItem:view, attribute:.Leading, multiplier:1, constant:10).active = true

}
override func viewDidLoad() {
    super.viewDidLoad()

    let b = NSButton()
    b.bezelStyle = .RoundedBezelStyle
    b.translatesAutoresizingMaskIntoConstraints = false
    view.addSubview(b)

    NSLayoutConstraint(item: b, attribute:.Top, relatedBy:.Equal, toItem: view, attribute:.Top, multiplier:1, constant:10).active = true
    NSLayoutConstraint(item: b, attribute:.Leading, relatedBy:.Equal, toItem: view, attribute:.Leading, multiplier:1, constant:10).active = true

    NSLayoutConstraint(item:b, attribute:.Top, relatedBy:.Equal, toItem:view, attribute:.Top, multiplier:1, constant:10).active = true
    NSLayoutConstraint(item:b, attribute:.Leading, relatedBy:.Equal, toItem:view, attribute:.Leading, multiplier:1, constant:10) .active = true
}

Layout Constraint Creation
Layout Constraint Creation

NSLayoutConstraint(item:b, attribute:.Top, relatedBy:.Equal, toItem:view, attribute:.Top, multiplier:1, constant:10)
NSLayoutConstraint(item:b, attribute:.Leading, relatedBy:.Equal, toItem:view, attribute:.Leading, multiplier:1, constant:10)
Layout Constraint Creation

Layout anchors

```
NSLayoutConstraint(item:b, attribute:.Top, relatedBy:.Equal, toItem:view, attribute:.Top, multiplier:1, constant:10)
NSLayoutConstraint(item:b, attribute:.Leading, relatedBy:.Equal, toItem:view, attribute:.Leading, multiplier:1, constant:10)
```

```
b.topAnchor.constraintEqualToAnchor(view.topAnchor, constant:10)
b.leadingAnchor.constraintEqualToAnchor(view.leadingAnchor, constant:10)
```
Layout Constraint Creation

Layout anchors

```objective-c
[NSLayoutConstraint constraintWithItem:b attribute:NSLayoutAttributeTop relatedBy:NSLayoutRelationEqual toItem:self.view attribute:NSLayoutAttributeTop multiplier:1 constant:10];
[NSLayoutConstraint constraintWithItem:b attribute:NSLayoutAttributeLeading relatedBy:NSLayoutRelationEqual toItem:self.view attribute:NSLayoutAttributeLeading multiplier:1 constant:10];

[b.topAnchor constraintEqualToAnchor:self.view.topAnchor constant:10];
[b.leadingAnchor constraintEqualToAnchor:self.view.leadingAnchor constant:10];
```
Layout Constraint Creation
Layout anchors
Cannot set a location equal to a constant

[v1.leadingAnchor constraintEqualToConstant:100];

// Error: may not respond to method
Layout Constraint Creation

Layout anchors

Cannot set a location equal to a constant

```swift
[v1.leadingAnchor constraintEqualToConstant:100];
// Error: may not respond to method
```

Cannot relate a location to a size

```swift
[v1.leadingAnchor constraintEqualToAnchor:v2.widthAnchor];
// Error: incompatible pointer type
```
Constraining Negative Space

Mystery #10
Constraining Negative Space

Equal spacing between buttons

Centering a group
Constraining Negative Space

Equal spacing between buttons

Centering a group
**NSLayoutGuide / UILayoutGuide**

**UILayoutGuide** represents a rectangle in the layout engine

Constrain just like a view

```swift
let guide = UILayoutGuide()
view.addLayoutGuide(guide)
```
NSLayoutGuide / UILayoutGuide

Layout anchors are not available for margins

UIView now exposes layoutMarginsGuide

```swift
var layoutMarginsGuide: UILayoutGuide
```
Debugging Your Layout
Mysteries of Auto Layout, part 2

Kasia Wawer iOS Keyboards Engineer
Has This Ever Happened to You?
Has This Ever Happened to You?

Quiz question

Send Answer
Has This Ever Happened to You?

Layout spec

Quiz question

Build and run

(Not so much)

Probably at least one of the constraints in the following list is one you don't want.

```swift
("<_UILayoutSupportConstraint:0x7ffe9ad11a80 V:[UILayoutGuide:0x7ffe9ad10650(20)]"",
"<_UILayoutSupportConstraint:0x7ffe9ad10ba0 V:|-(0)-[UILayoutGuide:0x7ffe9ad10650] (Names: '|':UIView:0x7ffe9c81b720 )">

"<NSLayoutConstraint:0x7ffe9acbef60 'saturnWidth' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0 )">

"<NSLayoutConstraint:0x7ffe9acbef60 'imageHorizontal' saturn.leading == UIView:0x7ffe9c81b720.leadingMargin (Names: saturn:0x7ffe9acb8cb0 )">

"<NSLayoutConstraint:0x7ffe9acbef60 'verticalLayout' V:[UILayoutGuide:0x7ffe9ad10650]-(NSSpace(8))-[saturn] (Names: saturn:0x7ffe9acb8cb0 )>

"<NSLayoutConstraint:0x7ffe9acbef60 'imageMiddle' saturn.centerX == UIView:0x7ffe9c81b720.centerX (Names: saturn:0x7ffe9acb8cb0 )>

"<UILayoutConstraint:0x7ffe9ad10ba0 'labelToTop' V:|-(100)-[UILabel:0x7ffe9c81b720 Which planet is this?] (Names: '|':UIView:0x7ffe9c81b720 )>

"<UILayoutConstraint:0x7ffe9ad10ba0 'imageMiddle' saturn.centerX == UIView:0x7ffe9c81b720 centerX (Names: saturn:0x7ffe9acb8cb0 )>

"<UILayoutConstraint:0x7ffe9ad10ba0 'UIView-Encapsulated-Layout-Width' H:[UIView:0x7ffe9c81b720(375)]>

Will attempt to recover by breaking constraint

<LayoutConstraint:0x7ffe9acbef60 'saturnWidth' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0 )>
```
Unsatisfiable Constraints
Mystery #11
Understanding the Log

```
( "<_UILayoutSupportConstraint:0x7ffe9ad11a80 V:[_UILayoutGuide:
0x7ffe9ad10650(20)]>",
  "<_UILayoutSupportConstraint:0x7ffe9ad10ba0 V:|-(0)-[_UILayoutGuide:
0x7ffe9ad10650]  (Names: '|':UIView:0x7ffe9c81b720 )>",
  "<NSLayoutConstraint:0x7ffe9acbef60 'saturnWidth' saturn.width ==
1.5*saturn.height  (Names: saturn:0x7ffe9acb8cb0 )>",
  "<NSLayoutConstraint:0x7ffe9c905400 'imageHorizontal' saturn.leading
== UIView:0x7ffe9c81b720.leadingMargin  (Names: saturn:
0x7ffe9acb8cb0 )>",
  "<NSLayoutConstraint:0x7fedd3423ae0 'imageHorizontal' UIView:
0x7fedd3607b90.trailingMargin == saturn.trailingMargin>",  (Names: saturn:0x7ffe9acb8cb0 )",
  "<NSLayoutConstraint:0x7ffe9c905aa0 'verticalLayout' V:[_UILayoutGuide:0x7ffe9ad10650]-(NSSpace(8))-[saturn]  (Names: saturn:0x7ffe9acb8cb0 )>",
  "<NSLayoutConstraint:0x7ffe9c905b40 'verticalLayout' V:[saturn]-
(NSSpace(8))-[UILabel:0x7ffe9c03d10'Which planet is this?']  (Names:
saturn:0x7ffe9acb8cb0 )>",
  "<NSLayoutConstraint:0x7ffe9c906050 'labelToTop' V:|-(100)-[UILabel:
0x7ffe9c903d10'Which planet is this?']  (Names: '|':UIView:0x7ffe9c81b720
)>",
  "<NSLayoutConstraint:0x7ffe9c906130 'UIView-Encapsulated-Layout-Width'
H:[ UIView:0x7ffe9c81b720(375)]>")
)

Will attempt to recover by breaking constraint
<NSLayoutConstraint:0x7ffe9acbef60 'saturnWidth' saturn.width ==
1.5*saturn.height  (Names: saturn:0x7ffe9acb8cb0 )>
```

Which planet is this?
Understanding the Log

Will attempt to recover by breaking constraint
<NSLayoutConstraint:0x7ffe9acbef60 'saturnWidth' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0 )>

Will attempt to recover by breaking constraint
<NSLayoutConstraint:0x7ffe9acbef60 'saturnWidth' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0 )>

Will attempt to recover by breaking constraint
<NSLayoutConstraint:0x7ffe9acbef60 'saturnWidth' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0 )>

Will attempt to recover by breaking constraint
<NSLayoutConstraint:0x7ffe9acbef60 'saturnWidth' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0 )>

Will attempt to recover by breaking constraint
<NSLayoutConstraint:0x7ffe9acbef60 'saturnWidth' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0 )>

Will attempt to recover by breaking constraint
<NSLayoutConstraint:0x7ffe9acbef60 'saturnWidth' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0 )>

Will attempt to recover by breaking constraint
<NSLayoutConstraint:0x7ffe9acbef60 'saturnWidth' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0 )>

Will attempt to recover by breaking constraint
<NSLayoutConstraint:0x7ffe9acbef60 'saturnWidth' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0 )>

Will attempt to recover by breaking constraint
<NSLayoutConstraint:0x7ffe9acbef60 'saturnWidth' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0 )>
Understanding the Log

Will attempt to recover by breaking constraint

<NSLayoutConstraint:0x7ffe9acbef60 'saturnWidth ' saturn.width == 1.5*saturn.height   (Names: saturn:0x7ffe9acb8cb0 )>
Understanding the Log

```
( "<_UILayoutSupportConstraint:0x7ffe9ad11a80 V:[_UILayoutGuide: 0x7ffe9ad10650(20)]>",
  "<_UILayoutSupportConstraint:0x7ffe9ad10ba0 V:|-(0)-[_UILayoutGuide: 0x7ffe9ad10650] (Names: '|':UIView:0x7ffe9c81b720 )>",
  "<NSLayoutConstraint:0x7ffe9acbef60 'saturnWidth' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0 )>",
  "<NSLayoutConstraint:0x7ffe9c905460 'imageHorizontal' saturn.leading == UIView:0x7ffe9c81b720.leadingMargin (Names: saturn:0x7ffe9acb8cb0 )>",
  "<NSLayoutConstraint:0x7fedd3423ae0 'imageHorizontal' UIView:0x7fedd3607b90.trailingMargin == saturn.trailing> (Names: saturn:0x7ffe9acb8cb0 )>",
  "<NSLayoutConstraint:0x7ffe9c905aa0 'verticalLayout' V:[UILayoutGuide:0x7ffe9ad10650]-(NSSpace(8))-[saturn] (Names: saturn:0x7ffe9acb8cb0 )>",
  "<NSLayoutConstraint:0x7ffe9c905b40 'verticalLayout' V:[saturn]-(NSSpace(8))-[UILabel:0x7ffe9c903d10'Which planet is this?'] (Names: saturn:0x7ffe9acb8cb0 )>",
  "<NSLayoutConstraint:0x7ffe9aca0130 'UIView-Encapsulated-Layout-Width' H:[UIView:0x7ffe9c81b720(375)]>
)

Will attempt to recover by breaking constraint
<NSLayoutConstraint:0x7ffe9acbef60 'saturnWidth' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0 )>
```

Which planet is this?

answer

Send answer
Understanding the Log

Will attempt to recover by breaking constraint

<NSLayoutConstraint:0x7ffe99acbf60 'saturnWidth' saturn.width == 1.5*saturn.height  (Names: saturn:0x7ffe9acb8cb0 )>
Understanding the Log

```
( "<_UILayoutSupportConstraint:0x7ffe9ad11a80 V:[UILayoutGuide:0x7ffe9ad10650(20)]" ,
   "<_UILayoutSupportConstraint:0x7ffe9ad10ba0 V:[UILayoutGuide:0x7ffe9ad10650] (Names: '|':UIView:0x7ffe9c81b720)" ,
   "<NSLayoutConstraint:0x7ffe9acbef60 'saturnWidth ' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0 )" ,
   "<NSLayoutConstraint:0x7ffe9aca0130 'UIView-Encapsulated-Layout-Width' (Names: '|':UIView:0x7ffe9c81b720)
\n<NSLayoutConstraint:0x7ffe9c906050 'labelToTop' V:\[-100]-[UILabel:0x7ffe9acb8cb0]
\n<UILayoutGuide:0x7ffe9ad10650]-(NSSpace(8))-[saturn] (Names: saturn:0x7ffe9acb8cb0)
\n<UILayoutSupportConstraint:0x7ffe9ad10ba0 V:[UILayoutGuide:0x7ffe9ad10650(20)]",
   "<UILayoutSupportConstraint:0x7ffe9ad11a80 V:[UILayoutGuide:0x7ffe9ad10650]
\n<NSLayoutConstraint:0x7ffe9aca0130 'UIView-Encapsulated-Layout-Width' (Names: '|':UIView:0x7ffe9c81b720)
\n<NSLayoutConstraint:0x7ffe9c903d10 'Which planet is this?' (Names: 'Which planet is this?':UIView:0x7ffe9c81b720)
\n<NSLayoutConstraint:0x7ffe9c906050 'labelToTop' V:\[-100]-[UILabel:0x7ffe9acb8cb0]
\n<UILayoutGuide:0x7ffe9ad10650]-(NSSpace(8))-[saturn] (Names: saturn:0x7ffe9acb8cb0)
\n<UILayoutSupportConstraint:0x7ffe9ad10ba0 V:[UILayoutGuide:0x7ffe9ad10650(20)]",
   "<UILayoutSupportConstraint:0x7ffe9ad11a80 V:[UILayoutGuide:0x7ffe9ad10650]",
   "<UILayoutSupportConstraint:0x7ffe9ad10ba0 V:[UILayoutGuide:0x7ffe9ad10650] (Names: '|':UIView:0x7ffe9c81b720)"
)
```

Will attempt to recover by breaking constraint

```
<NSLayoutConstraint:0x7ffe9acbef60 'saturnWidth ' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0 )
```

Which planet is this?
Understanding the Log

```swift
("<UILayoutSupportConstraint:0x7ffe9ad11a80 V:[UILayoutGuide:0x7ffe9ad10650(20)]>",
"<UILayoutSupportConstraint:0x7ffe9ad10ba0 V:-(-0)-[UILayoutGuide:0x7ffe9ad10650] (Names: '|':UIView:0x7ffe9c81b720 )>",
"<NSLayoutConstraint:0x7ffe9acbef60 'saturnWidth ' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0 )>",
"<NSLayoutConstraint:0x7ffe9c906050 'labelToTop' V:|-(100)-[UILabel:0x7ffe9c903d10'Which planet is this?' (Names: '|':UIView:0x7ffe9c81b720 )>",
"<NSLayoutConstraint:0x7ffe9c905b40 'verticalLayout' V:[saturn]-(NSSpace(8))-[saturn:0x7ffe9acb8cb0 ]>",
"<UILayoutGuide:0x7ffe9ad10650-(NSSpace(8))-[saturn:0x7ffe9acb8cb0 ]>",
"<UILayoutSupportConstraint:0x7ffe9ad10ba0 V:-(-0)-[UILayoutGuide:0x7ffe9ad10650] (Names: '|':UIView:0x7ffe9c81b720 )>">

Which planet is this?

Will attempt to recover by breaking constraint
<LayoutConstraint:0x7ffe9acbef60 'saturnWidth ' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0 )>
```
Understanding the Log

```
( "<_UILayoutSupportConstraint:0x7ffe9ad11a80 V:[_UILayoutGuide: 0x7ffe9ad10650(20)]>",
 "<_UILayoutSupportConstraint:0x7ffe9ad1d0ba0 V:|-(0)-[_UILayoutGuide: 0x7ffe9ad10650] (Names: '|':UIView:0x7ffe9c81b720 )>",
 "<NSLayoutConstraint:0x7ffe9acbf60 'saturnWidth ' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0 )>",
 "<NSLayoutConstraint:0x7ffe9c905b40 'verticalLayout' V:[saturn]-[UIView:0x7ffe9c81b720].leadingMargin (Names: saturn:0x7ffe9acb8cb0 )>",
 "<NSLayoutConstraint:0x7fedd3423ae0 'imageHorizontal' UIView:0x7ffe9c903d10'Which planet is this?' (Names: saturn:0x7ffe9acb8cb0 )>",
 "<NSLayoutConstraint:0x7ffe9c905aa0 'verticalLayout' V:[UIView:0x7ffe9c81b720(375)]>",
 "<NSLayoutConstraint:0x7ffe9acbef60 'saturnWidth ' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0 )>",
 "<NSLayoutConstraint:0x7ffe9c906050 'labelToTop' V:|-(100)-[UILabel:0x7ffe9acbf60 'saturnWidth ' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0 )>"
)
```

Will attempt to recover by breaking constraint

```
<NSLayoutConstraint:0x7ffe9acbf60 'saturnWidth ' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0 )>
```
Understanding the Log

```swift
( "<_UILayoutSupportConstraint:0x7ffe9ad11a80 V:[UILayoutGuide: 0x7ffe9ad10650(20)]>", "<_UILayoutSupportConstraint:0x7ffe9ad10ba0 V:-0(0)->[UILayoutGuide: 0x7ffe9ad10650] (Names: '|':UIView:0x7ffe9c81b720 )">", "<NSLayoutConstraint:0x7ffe9acbef60 'saturnWidth ' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0 )>>, "<NSLayoutConstraint:0x7ffe9aca0130 'UIView-Encapsulated-Layout-Width' (Names: UIView:0x7ffe9c81b720(375))>" )< NSLayoutConstraint:0x7ffe9acbef60 'saturnWidth ' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0 )> "<_UILayoutSupportConstraint:0x7ffe9ad11a80 V:[UILayoutGuide: 0x7ffe9ad10650(20)]>", "<_UILayoutSupportConstraint:0x7ffe9ad10ba0 V:-0(0)->[UILayoutGuide: 0x7ffe9ad10650] (Names: '|':UIView:0x7ffe9c81b720 )">", "<NSLayoutConstraint:0x7ffe9acbef60 'saturnWidth ' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0 )>>, "<NSLayoutConstraint:0x7ffe9aca0130 'UIView-Encapsulated-Layout-Width' (Names: UIView:0x7ffe9c81b720(375))>" )

Will attempt to recover by breaking constraint
<NSLayoutConstraint:0x7ffe9acbef60 'saturnWidth ' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0 )>
```

Which planet is this?

Send answer
Understanding the Log

```
(<UILayoutSupportConstraint:0x7ffe9ad11a80 V:UILayoutGuide:0x7ffe9ad10650(20))>

(<UILayoutSupportConstraint:0x7ffe9ad10ba0 V:UILayoutGuide:0x7ffe9ad10650(20))>

<UILayoutSupportConstraint:0x7ffe99acb8cb0 V:UILayoutGuide:0x7ffe9ad10650(20)>

< NSLayoutConstraint:0x7ffe9c906050 'labelToTop' V:|-(100)-UILabel:0x7ffe9c903d10'Which planet is this?' (Names: |:UIView:0x7ffe9c81b720)>

< NSLayoutConstraint:0x7ffe9aca0130 'UIView-Encapsulated-Layout-Width' UIView:0x7ffe9acb8cb0(375)>

< NSLayoutConstraint:0x7ffe9aca0130 'UIView-Encapsulated-Layout-Width' UIView:0x7ffe9acb8cb0(375)>

< NSLayoutConstraint:0x7ffe9c905b40 'verticalLayout' V:saturn:0x7ffe9acb8cb0-(NSSpace(8))-saturn:

< NSLayoutConstraint:0x7ffe9c905b40 'verticalLayout' V:saturn:0x7ffe9acb8cb0-(NSSpace(8))-saturn:

< NSLayoutConstraint:0x7ffe9aca0130 'UIView-Encapsulated-Layout-Width' UIView:0x7ffe9acb8cb0(375)>

< NSLayoutConstraint:0x7ffe9c905460 'imageHorizontal' saturn:0x7ffe9acb8cb0(375)>

< NSLayoutConstraint:0x7ffe9c905460 'imageHorizontal' saturn:0x7ffe9acb8cb0(375)>

< NSLayoutConstraint:0x7ffe9c905aa0 'verticalLayout' V:saturn:0x7ffe9acb8cb0(375)>

< NSLayoutConstraint:0x7ffe9c905aa0 'verticalLayout' V:saturn:0x7ffe9acb8cb0(375)>

< NSLayoutConstraint:0x7ffe9c905460 'imageHorizontal' saturn:0x7ffe9acb8cb0(375)>

< NSLayoutConstraint:0x7ffe9c905460 'imageHorizontal' saturn:0x7ffe9acb8cb0(375)>

< NSLayoutConstraint:0x7ffe9c905aa0 'verticalLayout' V:saturn:0x7ffe9acb8cb0(375)>

< NSLayoutConstraint:0x7ffe9c905aa0 'verticalLayout' V:saturn:0x7ffe9acb8cb0(375)>

< NSLayoutConstraint:0x7ffe9c905460 'imageHorizontal' saturn:0x7ffe9acb8cb0(375)>

< NSLayoutConstraint:0x7ffe9c905460 'imageHorizontal' saturn:0x7ffe9acb8cb0(375)>

< NSLayoutConstraint:0x7ffe9c905aa0 'verticalLayout' V:saturn:0x7ffe9acb8cb0(375)>

< NSLayoutConstraint:0x7ffe9c905aa0 'verticalLayout' V:saturn:0x7ffe9acb8cb0(375)>

< NSLayoutConstraint:0x7ffe9c905460 'imageHorizontal' saturn:0x7ffe9acb8cb0(375)>

< NSLayoutConstraint:0x7ffe9c905460 'imageHorizontal' saturn:0x7ffe9acb8cb0(375)>

< NSLayoutConstraint:0x7ffe9c905aa0 'verticalLayout' V:saturn:0x7ffe9acb8cb0(375)>

< NSLayoutConstraint:0x7ffe9c905aa0 'verticalLayout' V:saturn:0x7ffe9acb8cb0(375)>

< NSLayoutConstraint:0x7ffe9c905460 'imageHorizontal' saturn:0x7ffe9acb8cb0(375)>

< NSLayoutConstraint:0x7ffe9c905460 'imageHorizontal' saturn:0x7ffe9acb8cb0(375)>

< NSLayoutConstraint:0x7ffe9c905aa0 'verticalLayout' V:saturn:0x7ffe9acb8cb0(375)>

< NSLayoutConstraint:0x7ffe9c905aa0 'verticalLayout' V:saturn:0x7ffe9acb8cb0(375)>

Will attempt to recover by breaking constraint
< NSLayoutConstraint:0x7ffe99acb8cb0 'saturnWidth' saturn.width == 1.5*saturn.height (Names: saturn:0x7ffe9acb8cb0)>
```

Which planet is this?

Send answer
Understanding the Log
Make it easier with identifiers

"<_UILayoutSupportConstraint:0x14630d40 V:[_UILayoutGuide:0x14538610(0)]>",
"<_UILayoutSupportConstraint:0x14627b90 V:|(0)-[_UILayoutGuide:0x14538610]
(Names: '|':UIView:0x14538470 )>"

"<NSLayoutConstraint:0x146778d0 UIImageView:0x146707c0.height ==
0.6*UIView:0x145831a0.height>",
"<NSLayoutConstraint:0x14677930 UILabel:0x14670f70’Photo caption’.centerY
<= UIView:0x145831a0.centerY>",
"<NSLayoutConstraint:0x146774e0 V:[_UILayoutGuide:0x14580ff0]-(NSSpace(8))-
[UIImageView:0x146707c0]>",
"<NSLayoutConstraint:0x14677550 V:[UIImageView:0x146707c0]-(NSSpace(8))-
[UILabel:0x14670f70’Photo caption’]>"
Understanding the Log

Make it easier with identifiers

"<_UILayoutSupportConstraint:0x14630d40 V:[_UILayoutGuide:0x14538610(0)]>",
"<_UILayoutSupportConstraint:0x14627b90 V:|-(0)-[_UILayoutGuide:0x14538610] (Names: '|':UIView:0x14538470 )>
"<NSLayoutConstraint:0x146778d0 UIImageView:0x146707c0.height == 0.6*UIView:0x145831a0.height>
"<NSLayoutConstraint:0x14677930 UILabel:0x14670f70'Photo caption'.centerY <= UIView:0x145831a0.centerY   (Names: Caption for photo: 0x14644ab0 )>
"<NSLayoutConstraint:0x146774e0 V:[_UILayoutGuide:0x14580ff0]-(NSSpace(8))-[UIImageView:0x146707c0]>
"<NSLayoutConstraint:0x14677550 V:[UIImageView:0x146707c0]-(NSSpace(8))-[UILabel:0x14670f70'Photo caption']>
"<NSLayoutConstraint:0x1464b4d0 'photoHeight' UIImageView:0x14644300.height == 0.6*UIView:0x14538470.height>
"<NSLayoutConstraint:0x1464b530 'captionToCenterY' Caption for photo.centerY <= UIView:0x14538470.centerY   (Names: Caption for photo: 0x14644ab0 )>
"<NSLayoutConstraint:0x1464b0e0 'topVerticalArray' V:[UILayoutGuide:0x14538610]-(NSSpace(8))-[UIImageView:0x14644300]>
"<NSLayoutConstraint:0x1464b150 'topVerticalArray' V:[UIImageView:0x14644300]-(NSSpace(8))-[Caption for photo]   (Names: Caption for photo: 0x14644ab0 )>"
Understanding the Log

Adding identifiers
Understanding the Log

Adding identifiers

Use constraint identifiers
Understanding the Log

Adding identifiers

Use constraint identifiers

Explicit constraints

```
labelToTop.identifier = @"labelToTop";
```
Use constraint identifiers

Explicit constraints

    labelToTop.identifier = @"labelToTop";

Constraints using VFL

    for (NSLayoutConstraint *constraint in verticalLayout)
    {
        constraint.identifier = @"verticalLayout";
    }
Understanding the Log
Adding identifiers

Use constraint identifiers

Explicit constraints

```swift
labelToTop.identifier = @"labelToTop";
```

Constraints using VFL

```swift
for (NSLayoutConstraint *constraint in verticalLayout) {
    constraint.identifier = @"verticalLayout";
}
```

Constraints in Interface Builder
Understanding the Log

Tips
Understanding the Log

Tips

Set accessibility identifiers

• Identifies views in logs
Understanding the Log

Tips

Set accessibility identifiers

• Identifies views in logs

Set identifiers on layout guides
Understanding the Log

Tips

Set accessibility identifiers
• Identifies views in logs

Set identifiers on layout guides

Add as you go
Understanding the Log

Tips

Set accessibility identifiers
• Identifies views in logs

Set identifiers on layout guides

Add as you go

View one axis at a time
• `constraintsAffectingLayoutForAxis`: on iOS
• `constraintsAffectingLayoutForOrientation`: on OS X
Demo

Unsatisfiable constraints
Understanding the Log
Understanding the Log

Start from the bottom
Understanding the Log

Start from the bottom

Check `translatesAutoresizingMaskIntoConstraints`
Understanding the Log

Start from the bottom
Check `translatesAutoresizingMaskIntoConstraints`
Set identifiers
Understanding the Log

Start from the bottom

Check `translatesAutoresizingMaskIntoConstraints`

Set identifiers

Use `constraintsAffectingLayoutForAxis:`
Resolving Ambiguity

Mystery #12
Ambiguous Layouts

Why doesn’t my layout look right?
Ambiguous Layouts

Why doesn’t my layout look right?

Possible causes

• Too few constraints
Ambiguous Layouts
Why doesn’t my layout look right?

Possible causes
• Too few constraints
Ambiguous Layouts
Why doesn’t my layout look right?

Possible causes
• Too few constraints
Ambiguous Layouts
Why doesn’t my layout look right?

Possible causes
• Too few constraints
Ambiguous Layouts
Why doesn’t my layout look right?

Possible causes
• Too few constraints
Ambiguous Layouts

Why doesn’t my layout look right?

Possible causes

• Too few constraints
• Conflicting priorities
Ambiguous Layouts
Why doesn’t my layout look right?

Possible causes
• Too few constraints
• Conflicting priorities
Ambiguous Layouts
Why doesn’t my layout look right?

Possible causes
• Too few constraints
• Conflicting priorities
Ambiguous Layouts

Why doesn’t my layout look right?

Possible causes

• Too few constraints
• Conflicting priorities
Ambiguous Layouts
Why doesn’t my layout look right?

Possible causes
• Too few constraints
• Conflicting priorities
Ambiguous Layouts
Why doesn’t my layout look right?

Possible causes

- Too few constraints
- Conflicting priorities
Ambiguous Layouts
Why doesn’t my layout look right?

Possible causes
• Too few constraints
• Conflicting priorities

Content hugging priorities

Both:
contentHuggingPriority = 250
compressionResistancePriority = 750
Ambiguous Layouts
Why doesn’t my layout look right?

Possible causes

• Too few constraints
• Conflicting priorities

Content hugging priorities

Both:
contentHuggingPriority = 250
compressionResistancePriority = 750
Ambiguous Layouts
Why doesn’t my layout look right?

Possible causes
• Too few constraints
• Conflicting priorities

Content hugging priorities

Button:
contentHuggingPriority = 249
Ambiguous Layouts
Why doesn’t my layout look right?

Possible causes
• Too few constraints
• Conflicting priorities

Content hugging priorities

Button:
contentHuggingPriority = 249
Ambiguous Layouts
Why doesn’t my layout look right?

Possible causes
• Too few constraints
• Conflicting priorities

Content hugging priorities

Button:
contentHuggingPriority = 251
Ambiguous Layouts
Why doesn’t my layout look right?

Possible causes
• Too few constraints
• Conflicting priorities

Content hugging priorities

Button:
contentHuggingPriority = 251
Ambiguous Layouts
Why doesn’t my layout look right?

Possible causes
• Too few constraints
• Conflicting priorities

Button:
contentHuggingPriority = 251
Ambiguous Layouts

Why doesn’t my layout look right?

Possible causes

• Too few constraints
• Conflicting priorities

Content hugging priorities

answer

Send answer

Button:
contentHuggingPriority = 251
Resolving Ambiguity

Diagnostic tools
Resolving Ambiguity

Diagnostic tools

Red and yellow icons in IB
Resolving Ambiguity
Diagnostic tools

Red and yellow icons in IB
Resolving Ambiguity

Diagnostic tools

Red and yellow icons in IB
Resolving Ambiguity

Diagnostic tools

Red and yellow icons in IB

```objective-c
(lldb) po [self.view _autolayoutTrace]
UIWindow:0x7fe7434a3fe0
 -UIView:0x7fe7434a8140
  -_UILayoutGuide:0x7fe7434a84f0
  -_UILayoutGuide:0x7fe7434a90d0
   -Mercury:0x7fe7434a7790
   -Venus:0x7fe743639380
   -Earth:0x7fe74363aae0
   -Mars:0x7fe74363bed0
   -Jupiter:0x7fe74363ce30
   -Saturn:0x7fe74363e220- AMBIGUOUS LAYOUT for Saturn.minX{id: 165}
   -Uranus:0x7fe74363f690
   -Neptune:0x7fe743640d60

Legend:
* - is laid out with auto layout
+ - is laid out manually, but is represented in the layout engine
because translatesAutoresizingMaskIntoConstraints = YES
• - layout engine host
```

(lldb)
Resolving Ambiguity

Diagnostic tools

Red and yellow icons in IB

_autolayoutTrace

Select Debug > View Debugging
Resolving Ambiguity

Diagnostic tools

Red and yellow icons in IB

_selectAutolayoutTrace

Select Debug > View Debugging
Resolving Ambiguity

Diagnostic tools

Red and yellow icons in IB

_autolayoutTrace

Select Debug > View Debugging
Resolving Ambiguity

Diagnostic tools

Red and yellow icons in IB

_autolayoutTrace

Select Debug > View Debugging
Resolving Ambiguity
Diagnostic tools

Red and yellow icons in IB
_autolayoutTrace
Select Debug > View Debugging
Look in the view debugger
Resolving Ambiguity

Diagnostic tools

Red and yellow icons in IB

_select autolayoutTrace_

Select Debug > View Debugging

Look in the view debugger
Resolving Ambiguity

Diagnostic tools

Red and yellow icons in IB

_autolayoutTrace

Select Debug > View Debugging

Look in the view debugger

exerciseAmbiguityInLayout
Resolving Ambiguity

Diagnostic tools

Red and yellow icons in IB

_selectDebugView

Select Debug > View Debugging

Look in the view debugger

exerciseAmbiguityInLayout
Resolving Ambiguity
Diagnostic tools

Red and yellow icons in IB
_autolayoutTrace
Select Debug > View Debugging
Look in the view debugger
exerciseAmbiguityInLayout
Demo
Ambiguous layouts
Debugging Your Layout
Debugging Your Layout

Think about what information the engine needs
Debugging Your Layout

Think about what information the engine needs
Use the logs when constraints are unsatisfiable
Debugging Your Layout

Think about what information the engine needs
Use the logs when constraints are unsatisfiable
• Add identifiers for constraints and views
Debugging Your Layout

Think about what information the engine needs
Use the logs when constraints are unsatisfiable
  • Add identifiers for constraints and views
Check for ambiguity regularly
Debugging Your Layout

Think about what information the engine needs
Use the logs when constraints are unsatisfiable
• Add identifiers for constraints and views
Check for ambiguity regularly
Use tools to help resolve issues
Debugging Your Layout

Think about what information the engine needs
Use the logs when constraints are unsatisfiable
• Add identifiers for constraints and views
Check for ambiguity regularly
Use tools to help resolve issues
• Icons in Interface Builder
Debugging Your Layout

Think about what information the engine needs

Use the logs when constraints are unsatisfiable

• Add identifiers for constraints and views

Check for ambiguity regularly

Use tools to help resolve issues

• Icons in Interface Builder

• View debugger
Debugging Your Layout

Think about what information the engine needs
Use the logs when constraints are unsatisfiable
• Add identifiers for constraints and views
Check for ambiguity regularly
Use tools to help resolve issues
• Icons in Interface Builder
• View debugger
• Methods in lldb
Summary

Mysteries revealed

Part 1, Morning

• Maintainable Layouts
• Changing Constraints
• View Sizing
• Self-Sizing Table View Cells
• Priorities
• Alignment
Summary
Mysteries revealed

Part 1
• Maintainable Layouts
• Changing Constraints
• View Sizing
• Self-Sizing Table View Cells
• Priorities
• Alignment

Part 2, Afternoon
Summary

Mysteries revealed

Part 1
• Maintainable Layouts
• Changing Constraints
• View Sizing
• Self-Sizing Table View Cells
• Priorities
• Alignment

Part 2, Afternoon
• The Layout Cycle
Summary

Mysteries revealed

Part 1
• Maintainable Layouts
• Changing Constraints
• View Sizing
• Self-Sizing Table View Cells
• Priorities
• Alignment

Part 2, Afternoon
• The Layout Cycle
• Legacy Layout
Summary

Mysteries revealed

Part 1
- Maintainable Layouts
- Changing Constraints
- View Sizing
- Self-Sizing Table View Cells
- Priorities
- Alignment

Part 2, Afternoon
- The Layout Cycle
- Legacy Layout
- Constraint Creation
Summary
Mysteries revealed

Part 1
• Maintainable Layouts
• Changing Constraints
• View Sizing
• Self-Sizing Table View Cells
• Priorities
• Alignment

Part 2, Afternoon
• The Layout Cycle
• Legacy Layout
• Constraint Creation
• Constraining Negative Space
Summary

Mysteries revealed

Part 1
- Maintainable Layouts
- Changing Constraints
- View Sizing
- Self-Sizing Table View Cells
- Priorities
- Alignment

Part 2, Afternoon
- The Layout Cycle
- Legacy Layout
- Constraint Creation
- Constrained Negative Space
- Unsatisfiable Constraints
Summary

Mysteries revealed

Part 1
- Maintainable Layouts
- Changing Constraints
- View Sizing
- Self-Sizing Table View Cells
- Priorities
- Alignment

Part 2, Afternoon
- The Layout Cycle
- Legacy Layout
- Constraint Creation
- Constraining Negative Space
- Unsatisfiable Constraints
- Resolving Ambiguity
More Information

Documentation and Videos
Swift Language Documentation
http://developer.apple.com/swift

Technical Support
Apple Developer Forums
http://developer.apple.com/forums

Sample Code
AstroLayout

General Inquiries
Paul Marcos, App Frameworks Evangelist
pmarcos@apple.com
<table>
<thead>
<tr>
<th>Session</th>
<th>Location</th>
<th>Date &amp; Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mysteries of Auto Layout, Part 1</td>
<td>Presidio</td>
<td>Thursday 11:00AM</td>
</tr>
<tr>
<td>What’s New in Cocoa</td>
<td>Presidio</td>
<td>Tuesday 1:30PM</td>
</tr>
<tr>
<td>What’s New in UIKit Dynamics and Visual Effects</td>
<td>Mission</td>
<td>Friday 10:00AM</td>
</tr>
<tr>
<td>Cocoa Touch Best Practices</td>
<td>Presidio</td>
<td>Friday 1:30PM</td>
</tr>
<tr>
<td>What’s New in Internationalization</td>
<td>Pacific Heights</td>
<td>Friday 9:00 AM</td>
</tr>
<tr>
<td>New UIKit Support for International User Interfaces</td>
<td>Nob Hill</td>
<td>Thursday 2:30PM</td>
</tr>
<tr>
<td>Interface Builder and Auto Layout Lab</td>
<td>Developer Tools Lab C</td>
<td>Thursday 2:30PM</td>
</tr>
</tbody>
</table>