App Frameworks

#WWDC15

Layout and Animation Techniques For WatchKit

Session 216

Miguel Sanchez WatchKit Engineer
Tom Witkin WatchKit Engineer

Agenda

Agenda

Layout Fundamentals

Agenda

Layout Fundamentals

Using Groups

Agenda

Layout Fundamentals

Using Groups

Existing Animations in watchOS 1

Agenda

Layout Fundamentals

Using Groups

Existing Animations in watchOS 1

New Animation API in watchOS 2

Layout Fundamentals

WatchKit Layout Model

WatchKit Layout Model

Same model as WatchKit in watchOS 1

WatchKit Layout Model

Same model as WatchKit in watchOS 1

Different from UIKit and AppKit

WatchKit Layout Model

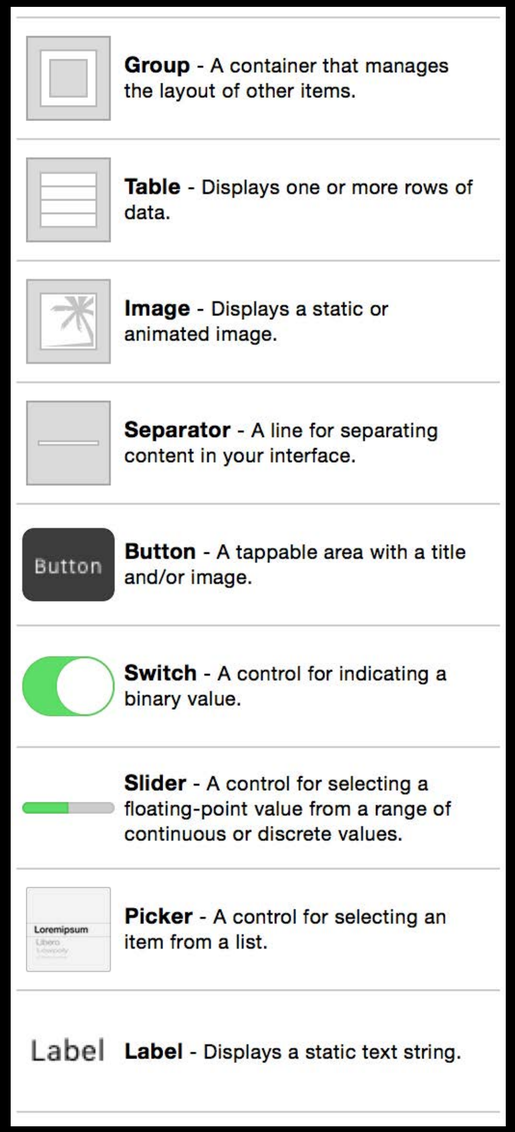
Same model as WatchKit in watchOS 1

Different from UIKit and AppKit

Flow-based layout

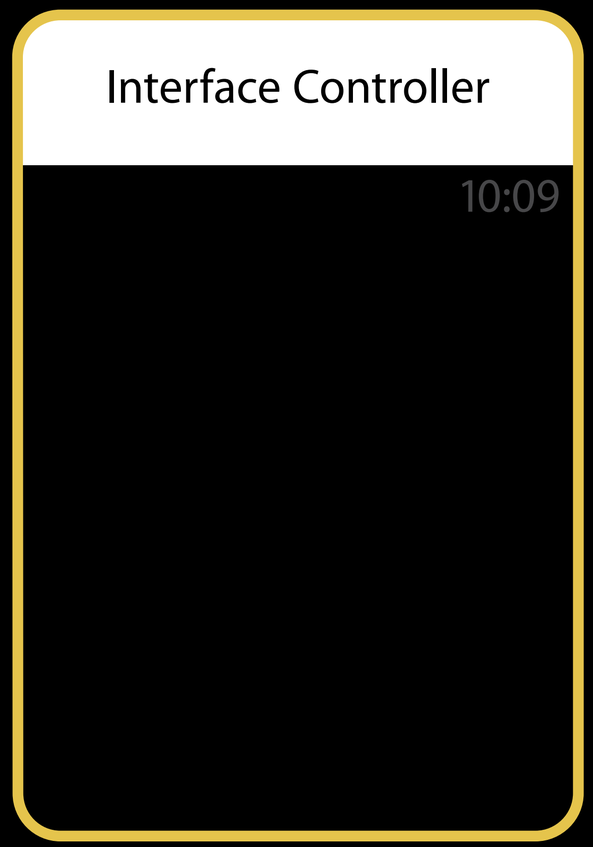
WatchKit Layout Model

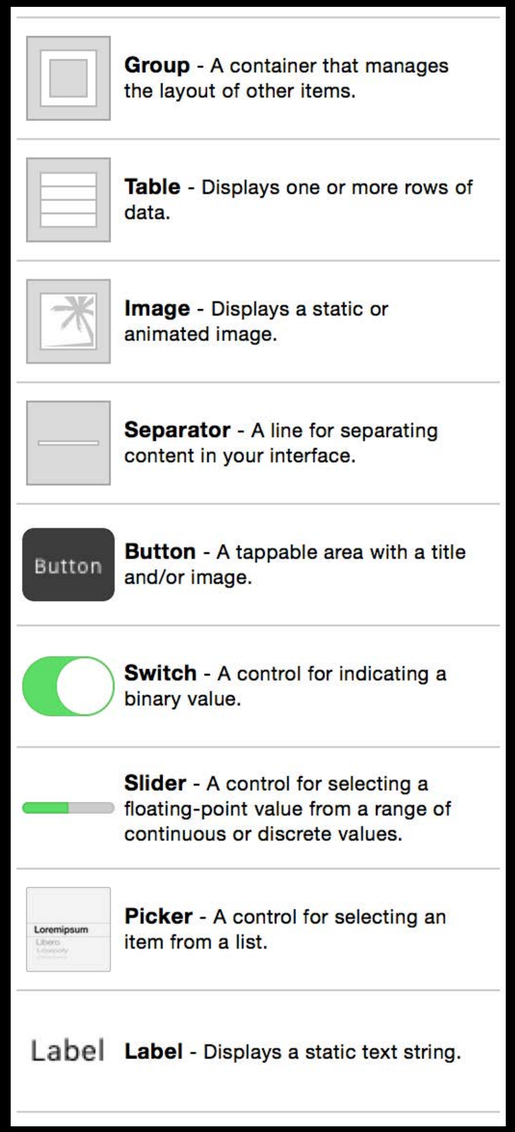
Flow-based layout



WatchKit Layout Model

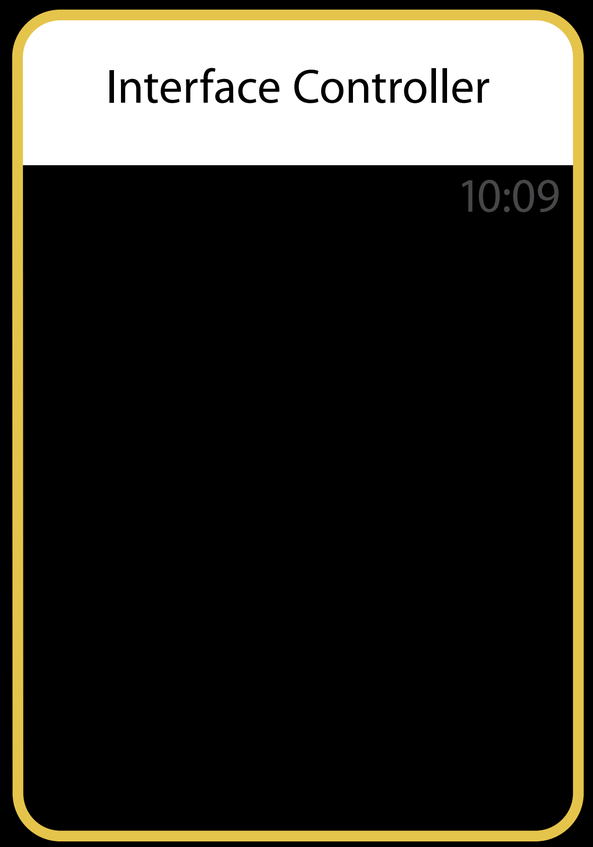
Flow-based layout

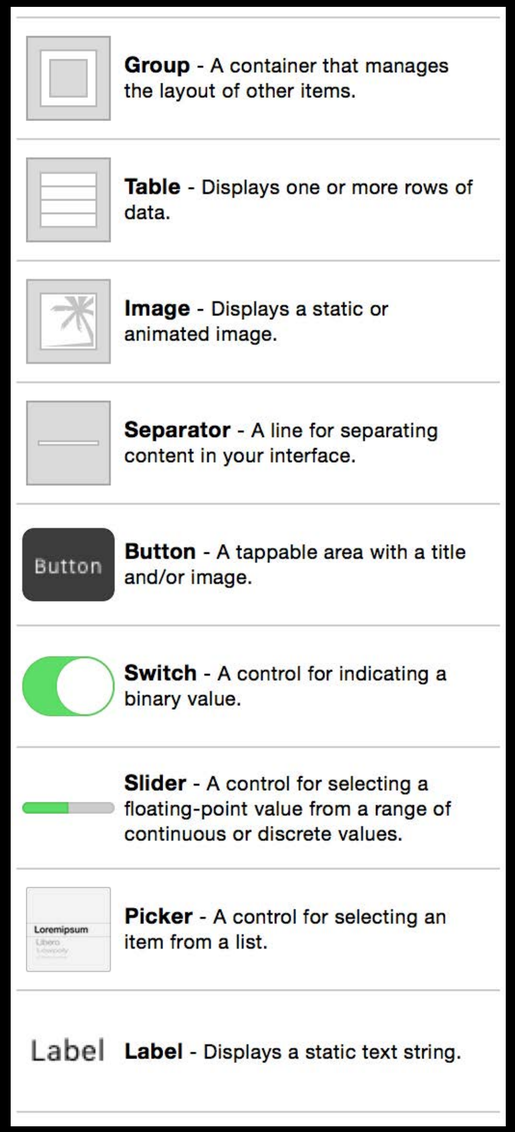




WatchKit Layout Model

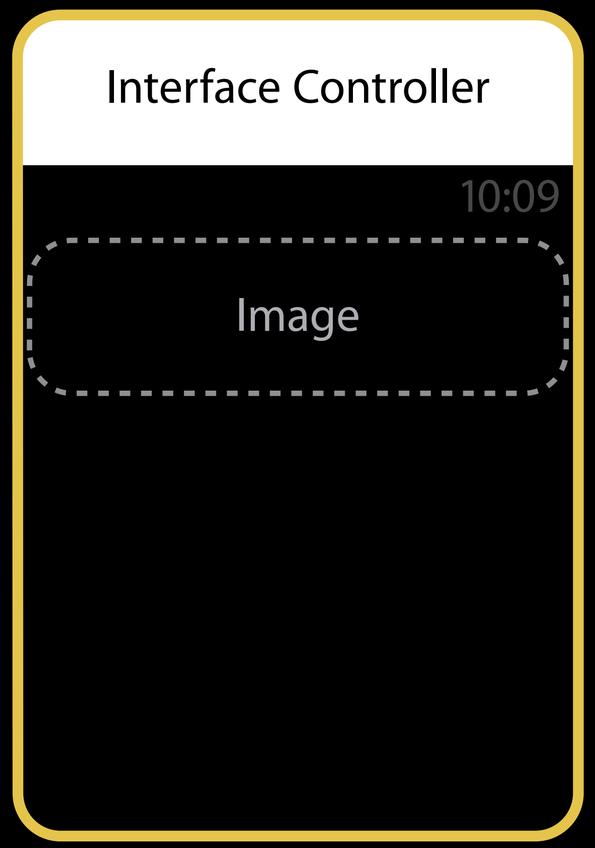
Flow-based layout

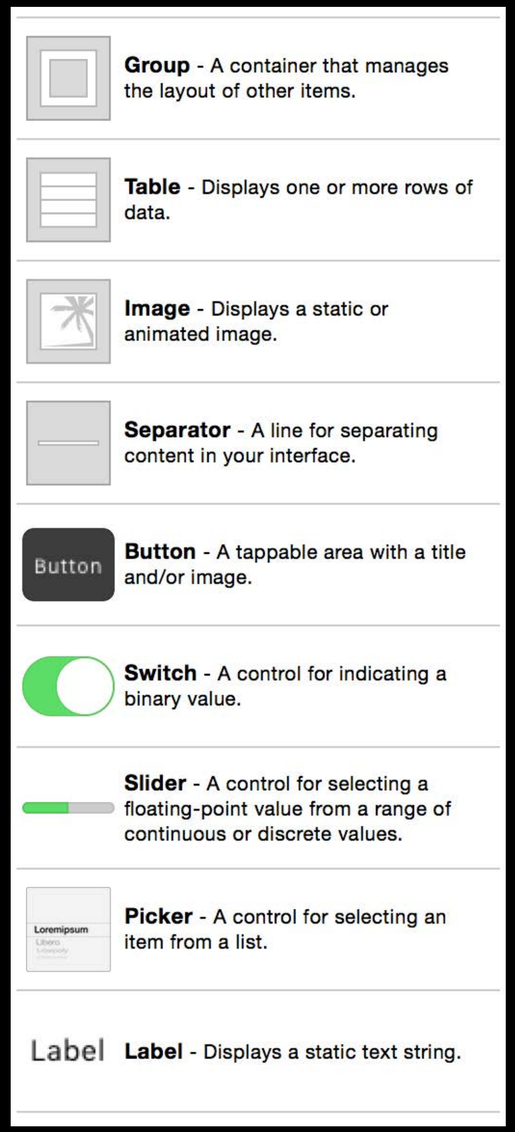




WatchKit Layout Model

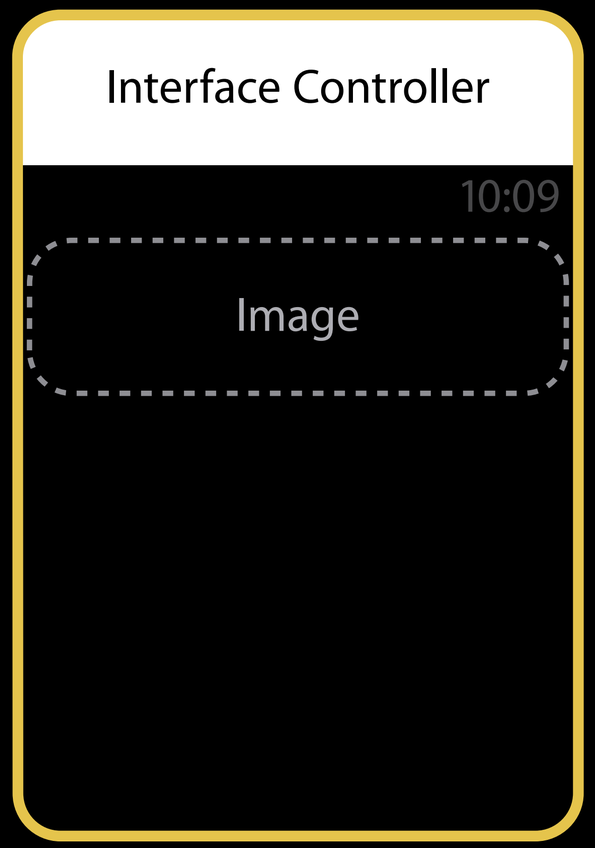
Flow-based layout

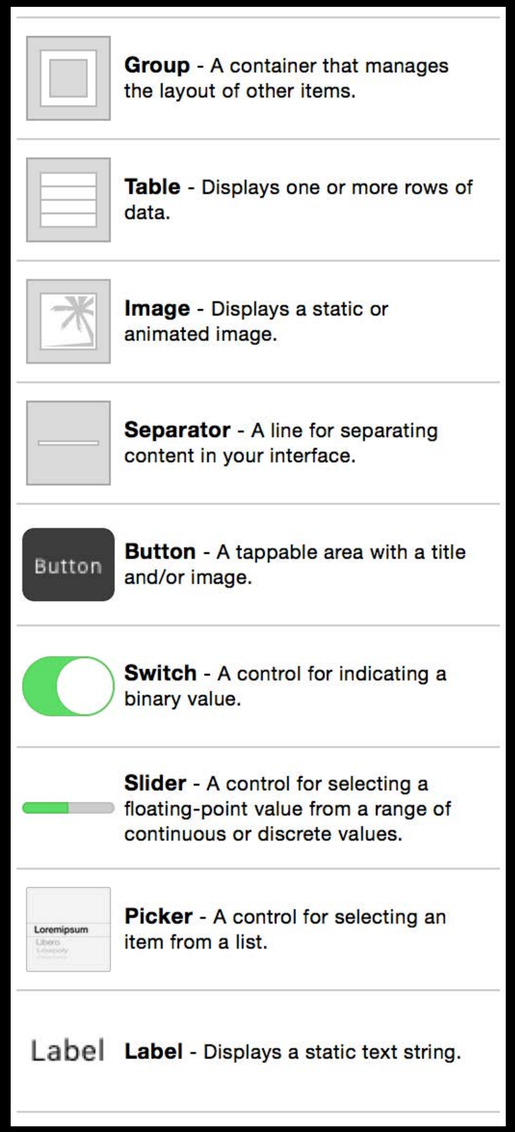




WatchKit Layout Model

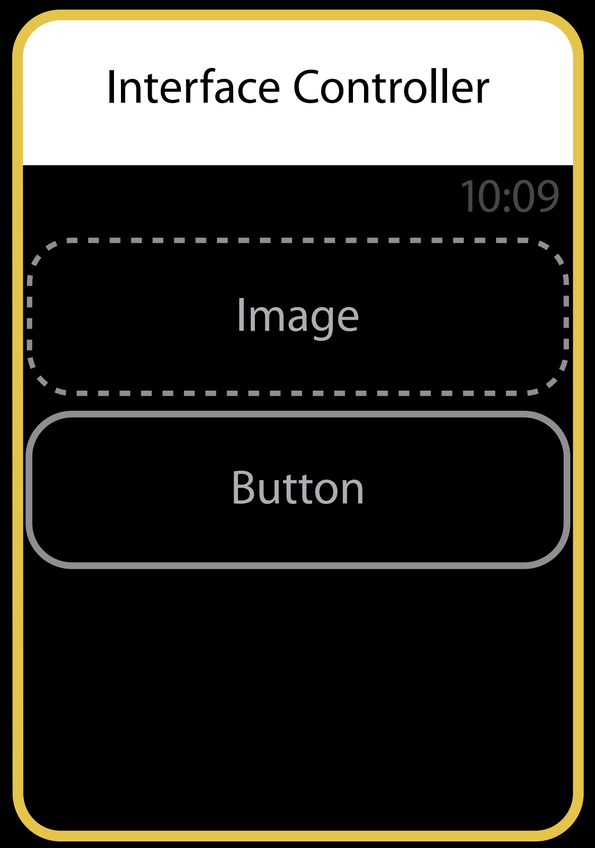
Flow-based layout

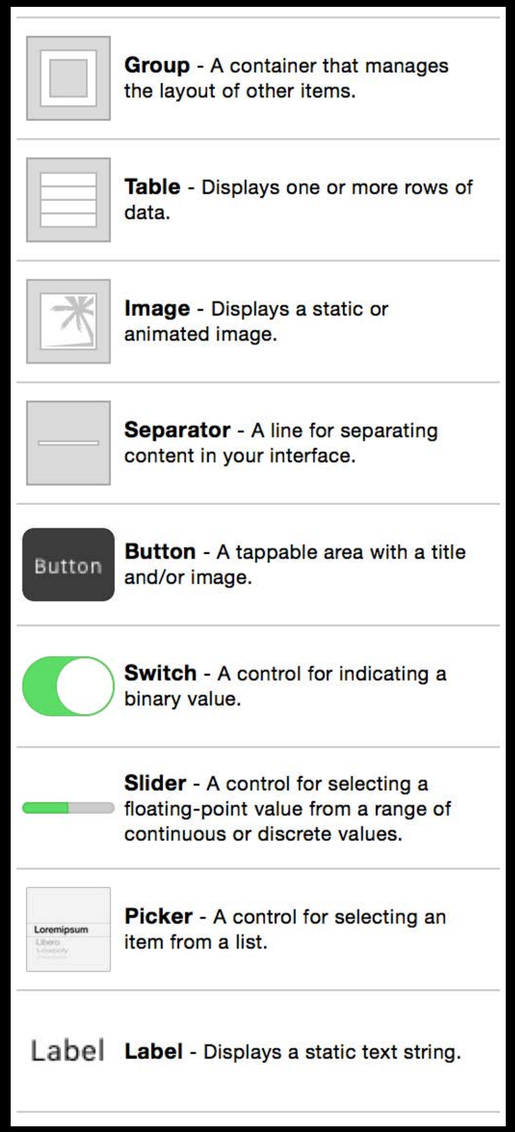




WatchKit Layout Model

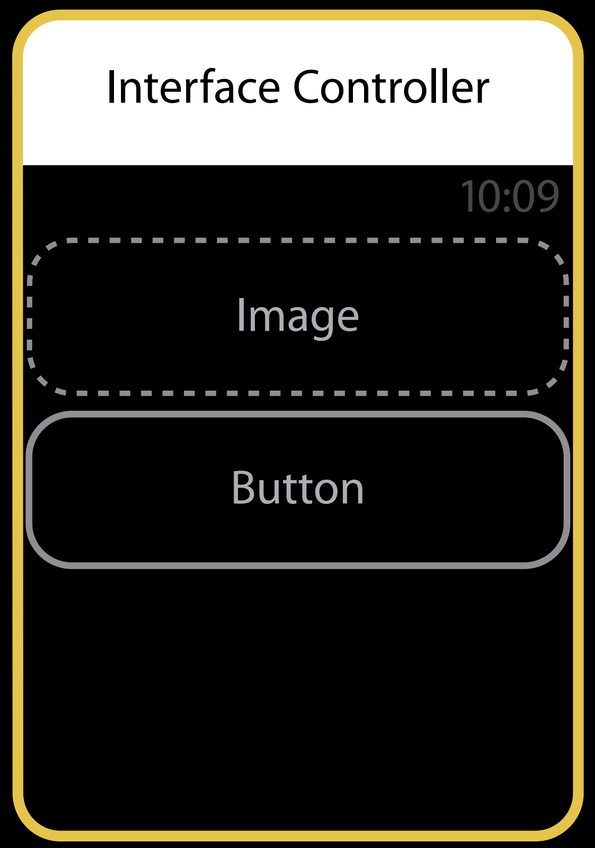
Flow-based layout

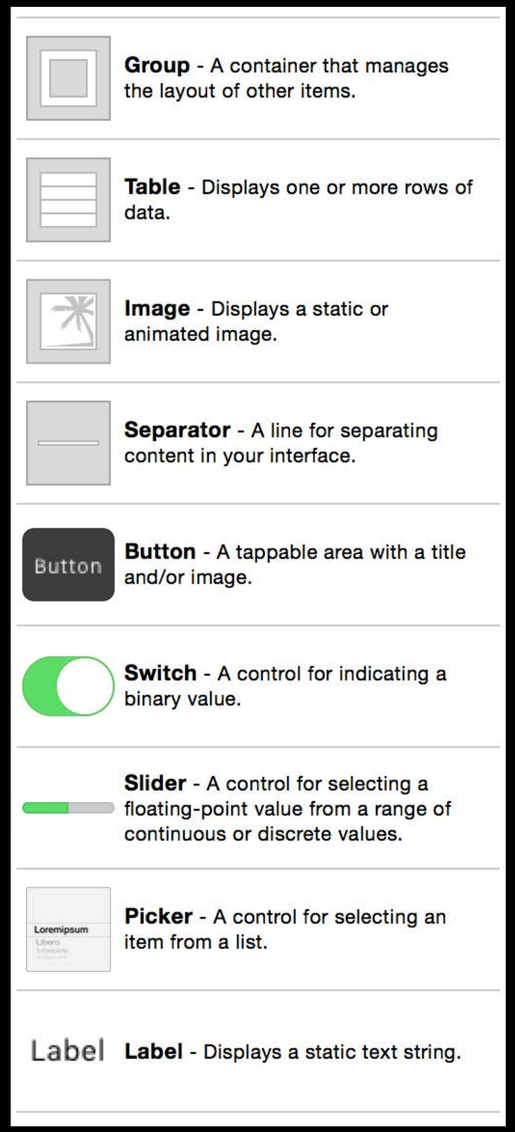




WatchKit Layout Model

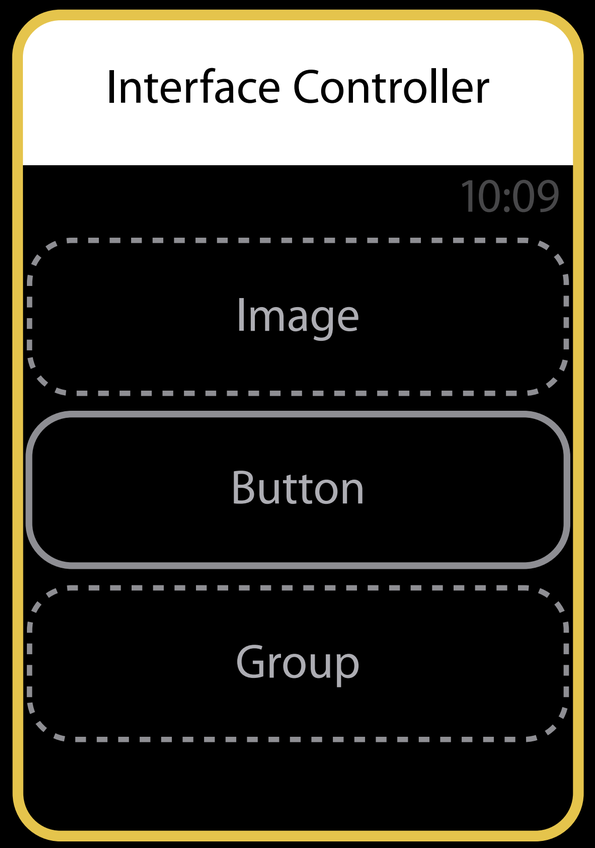
Flow-based layout

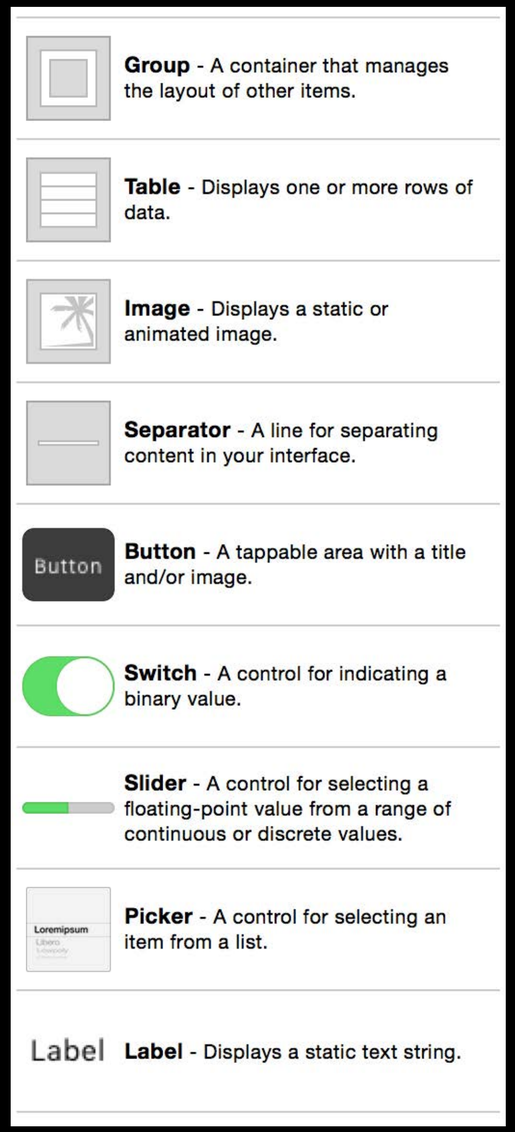




WatchKit Layout Model

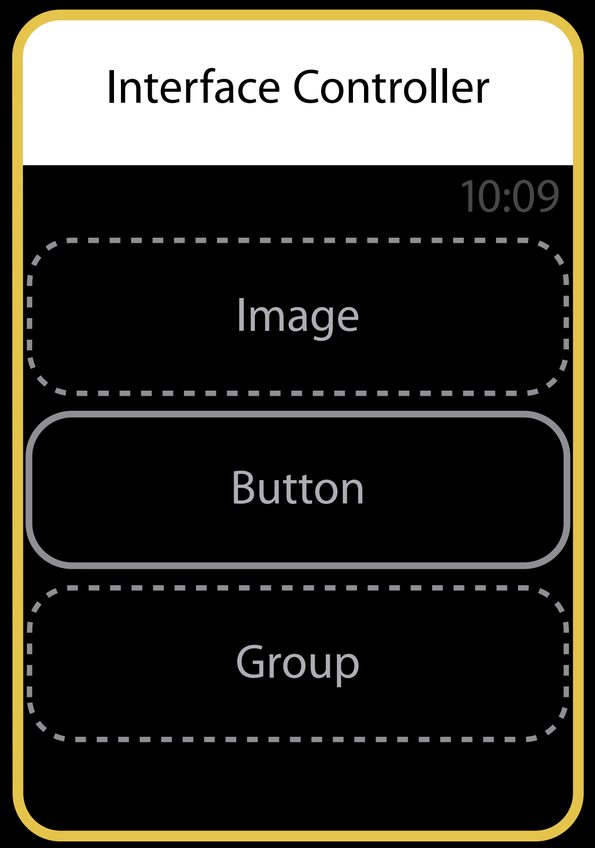
Groups are containers of elements

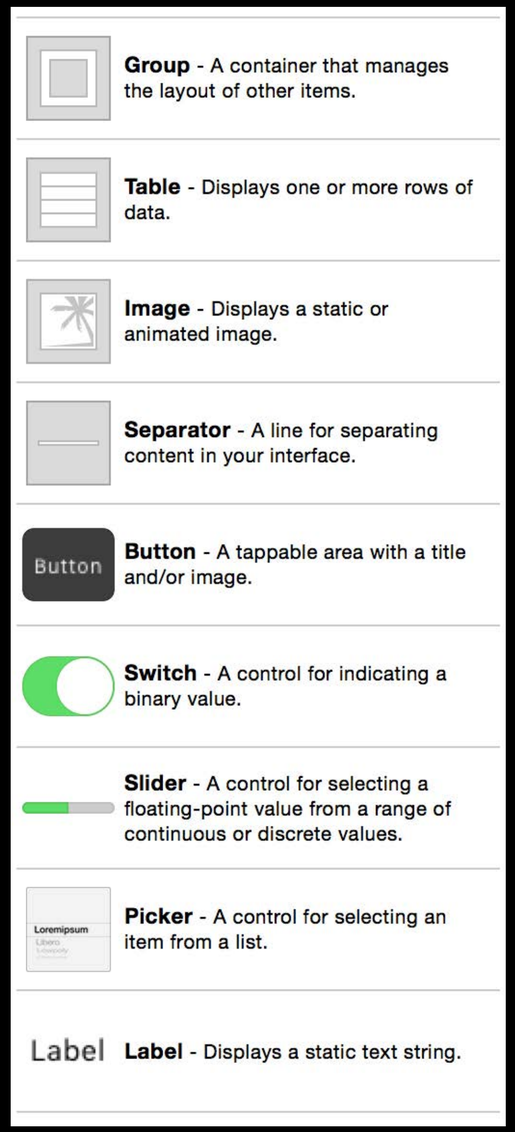




WatchKit Layout Model

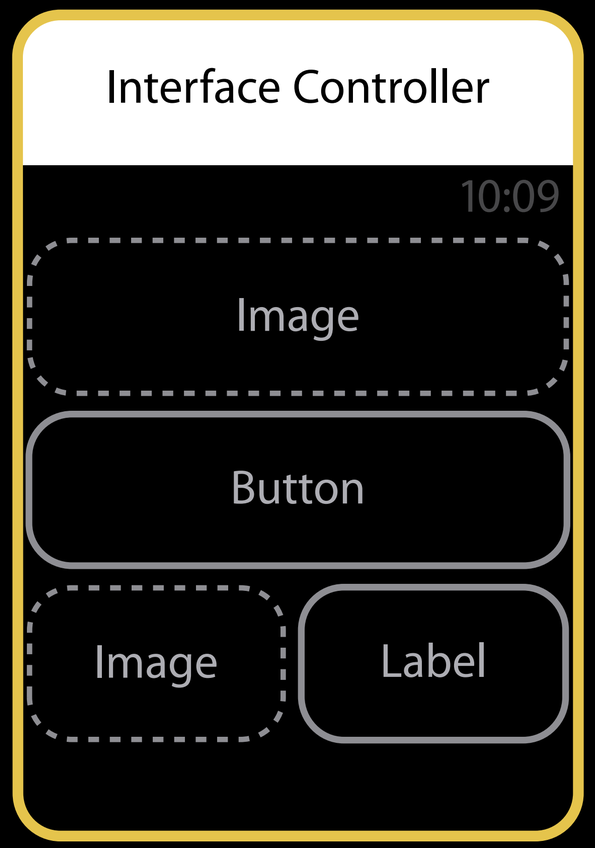
Groups are containers of elements

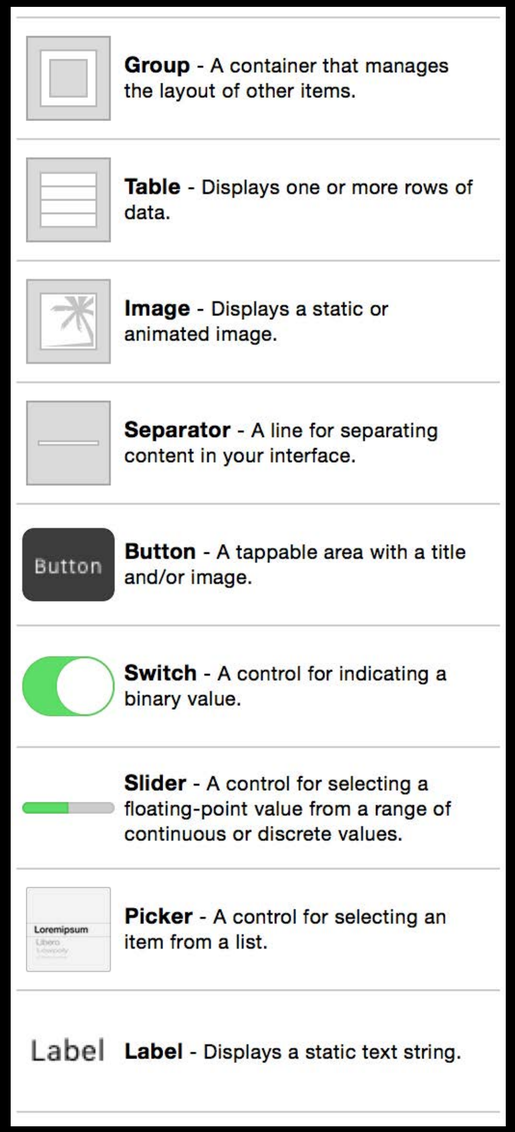




WatchKit Layout Model

Groups are containers of elements





WatchKit Layout Programming Model

WatchKit Layout Programming Model

You don’t write object creation code

WatchKit Layout Programming Model

You don’t write object creation code

Fine tuned control of:

WatchKit Layout Programming Model

You don't write object creation code

Fine tuned control of:

- Layout hierarchy

WatchKit Layout Programming Model

You don't write object creation code

Fine tuned control of:

- Layout hierarchy
- Alignment and sizing

WatchKit Layout Programming Model

You don't write object creation code

Fine tuned control of:

- Layout hierarchy
- Alignment and sizing
- Animation

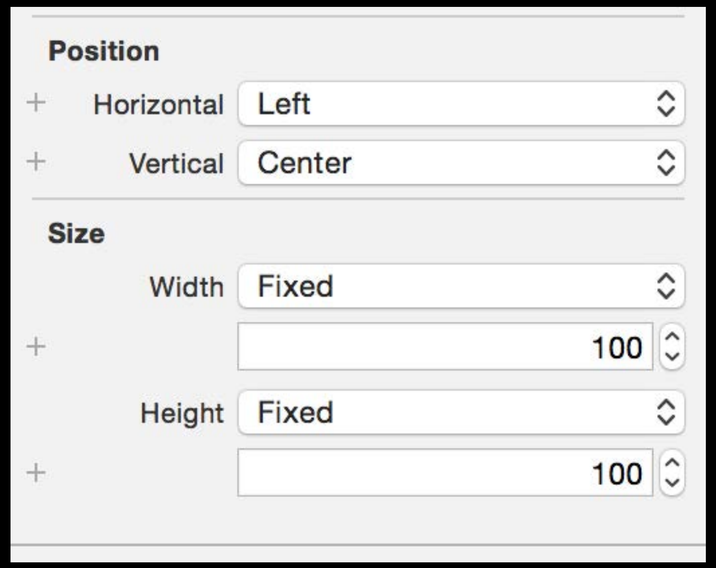
Alignment and Sizing

Alignment and Size

Properties on WKInterfaceObject

Alignment and Size

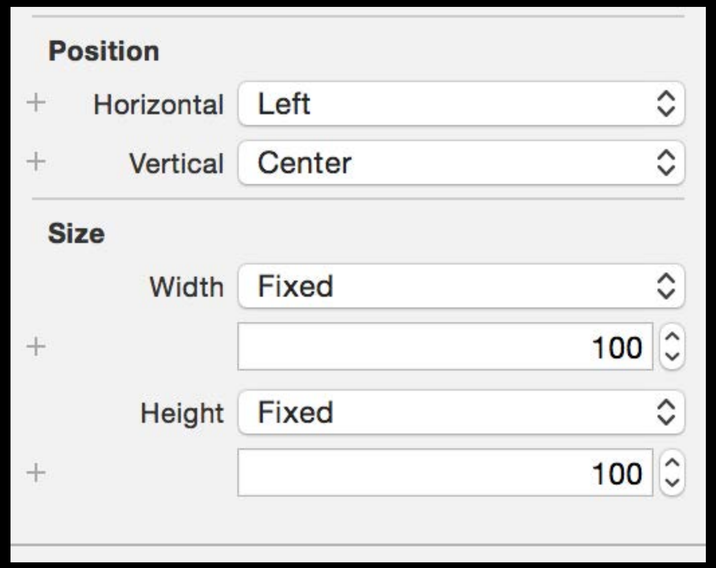
Properties on WKInterfaceObject



Alignment and Size

Properties on WKInterfaceObject

Alignment in containing object

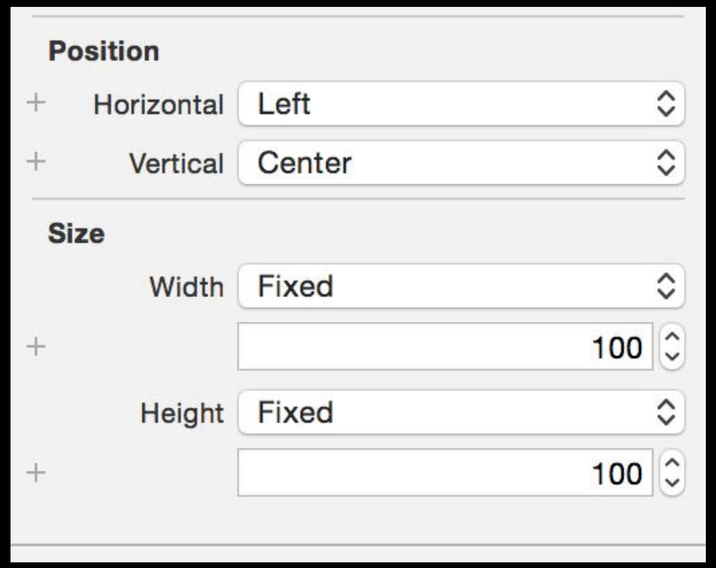


Alignment and Size

Properties on WKInterfaceObject

Alignment in containing object

- Horizontal and vertical

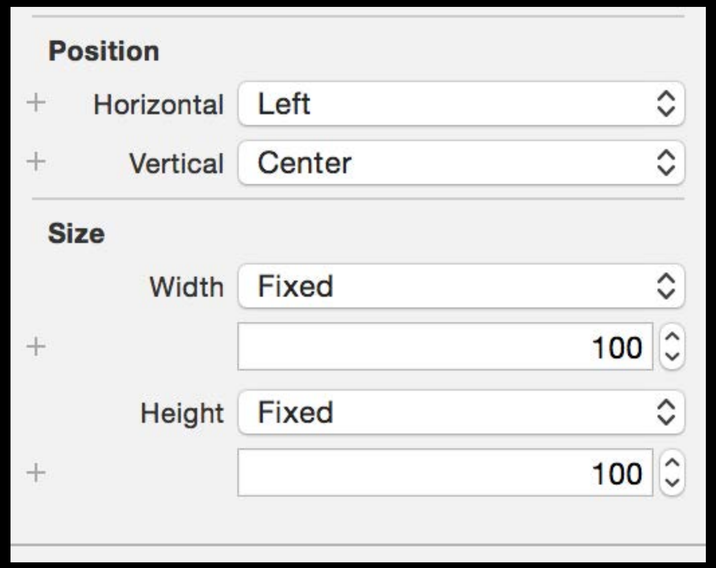


Alignment and Size

Properties on WKInterfaceObject

Alignment in containing object

- Horizontal and vertical
- Left, center, or right



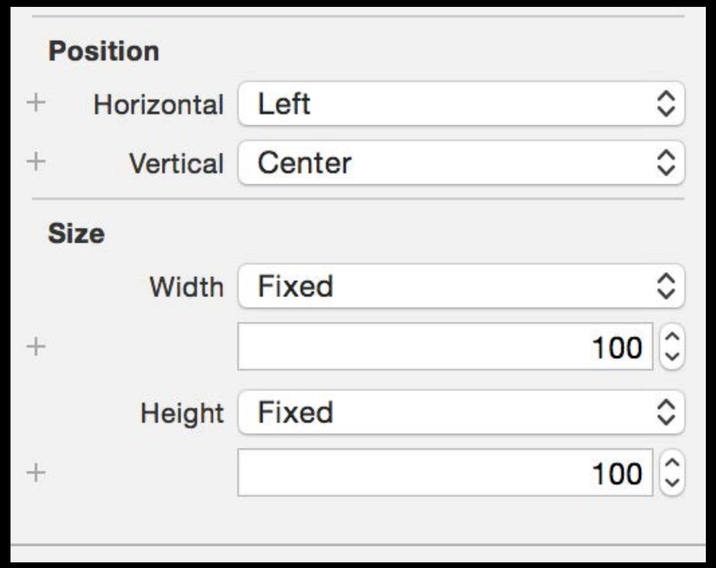
Alignment and Size

Properties on WKInterfaceObject

Alignment in containing object

- Horizontal and vertical
- Left, center, or right

Size



Alignment and Size

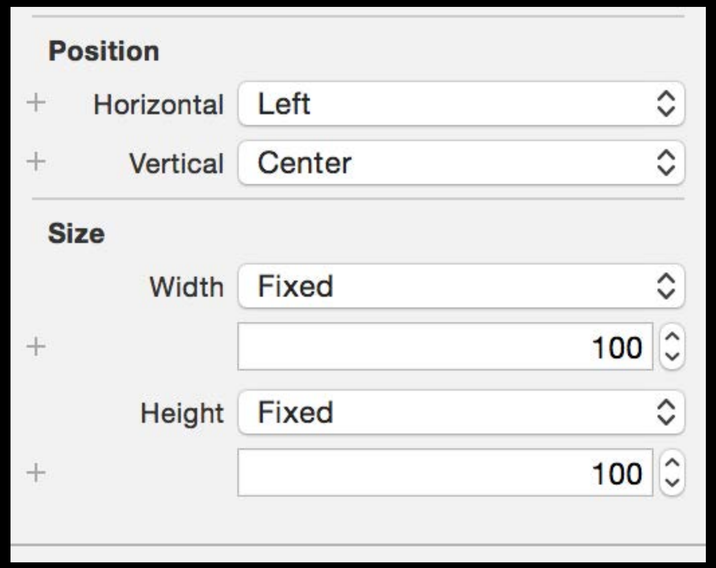
Properties on WKInterfaceObject

Alignment in containing object

- Horizontal and vertical
- Left, center, or right

Size

- Width and height heuristic



Alignment and Size

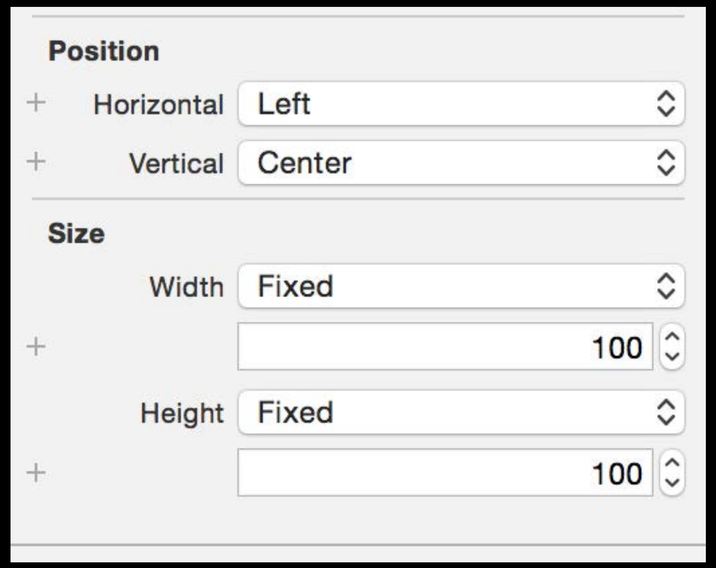
Properties on WKInterfaceObject

Alignment in containing object

- Horizontal and vertical
- Left, center, or right

Size

- Width and height heuristic
- Fixed, relative, or sized to fit



Alignment API

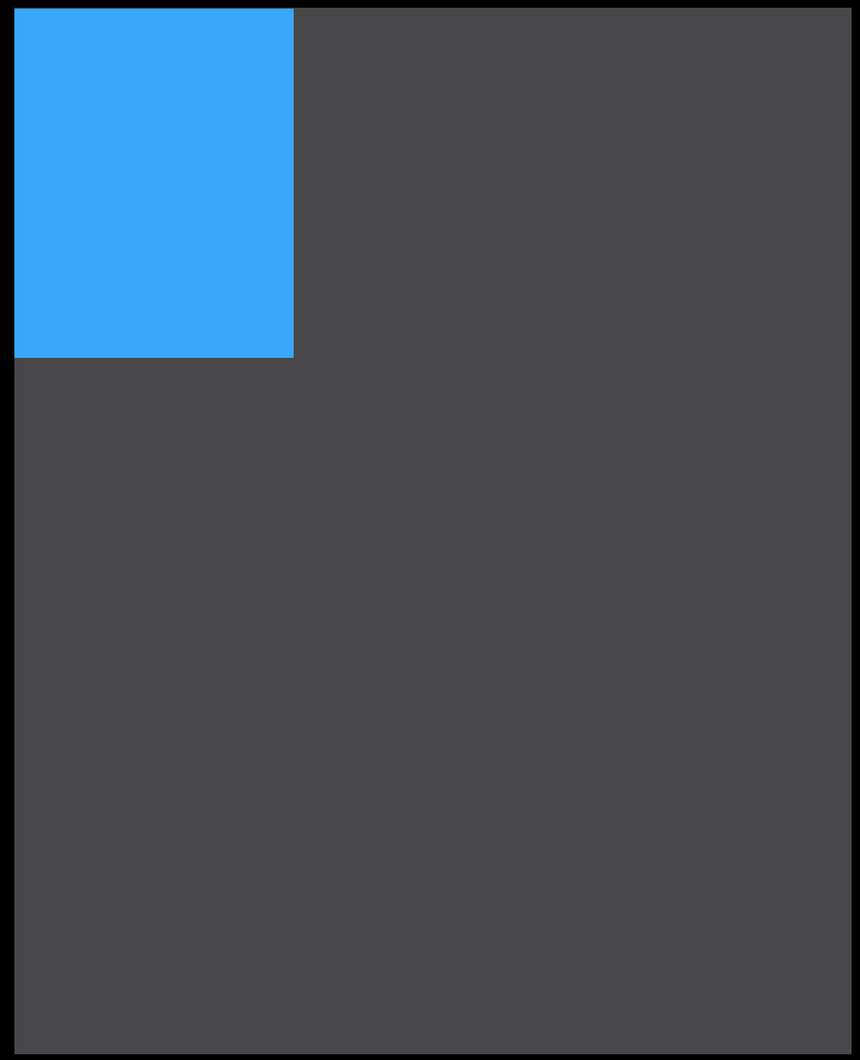
WKInterfaceObject

NEW

```
func setHorizontalAlignment(WKInterfaceObjectHorizontalAlignment)
func setVerticalAlignment(WKInterfaceObjectVerticalAlignment)
```

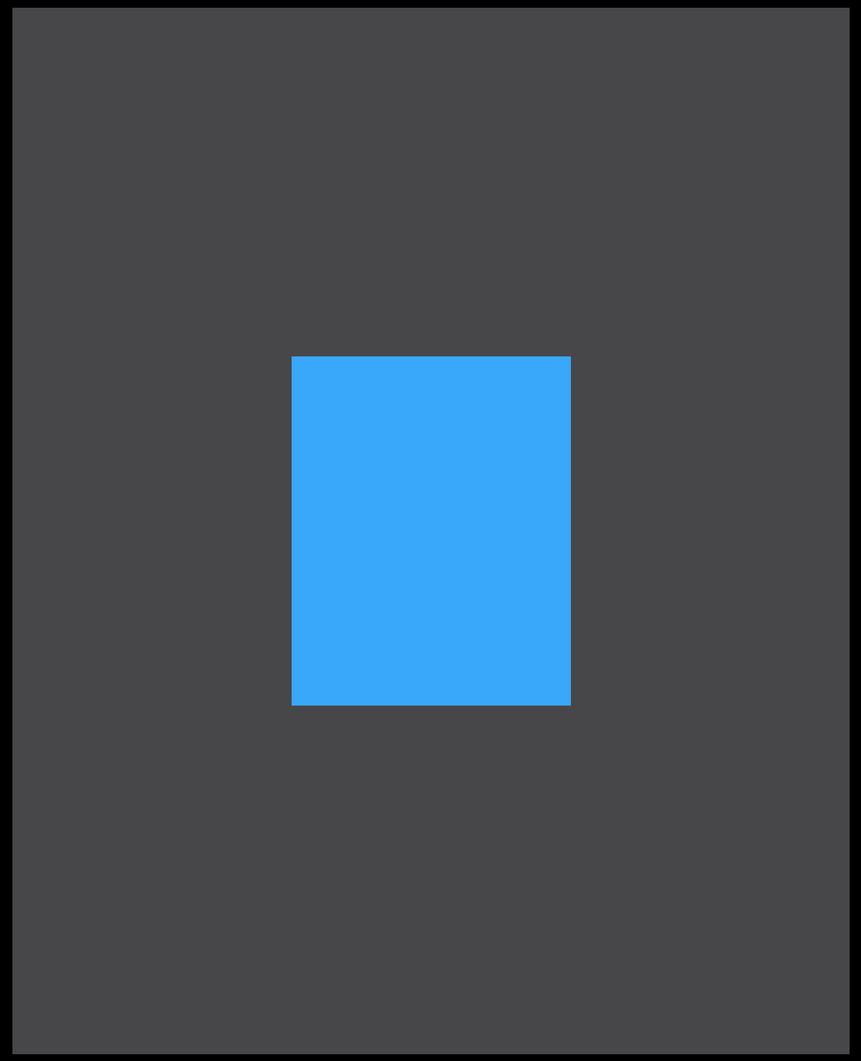
```
enum WKInterfaceObjectHorizontalAlignment  {
    case Left
    case Center
    case Right
}

enum WKInterfaceObjectVerticalAlignment {
    case Top
    case Center
    case Bottom
}
```

Alignment in Containing Element



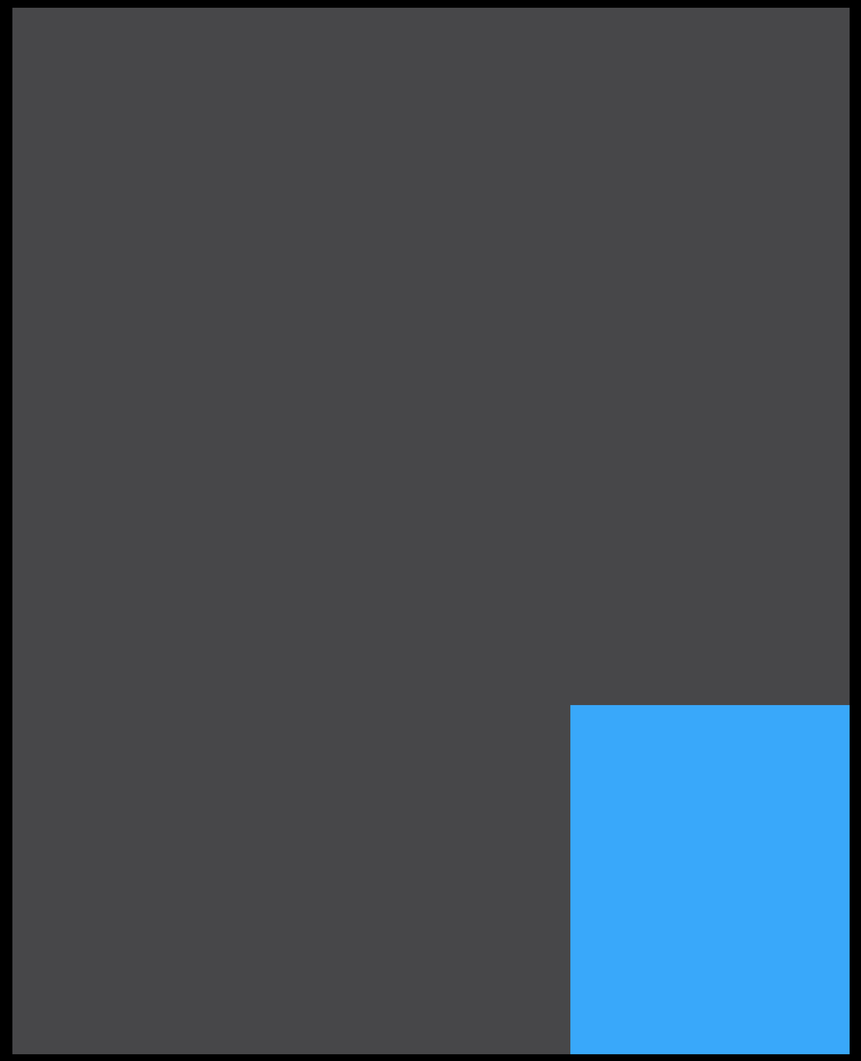
```
myObj.setHorizontalAlignment(.Left)
myObj.setVerticalAlignment(.Top)
```

Alignment in Containing Element



```
myObj.setHorizontalAlignment(.Center)
myObj.setVerticalAlignment(.Center)
```

Alignment in Containing Element



```
myObj.setHorizontalAlignment(.Right)
myObj.setVerticalAlignment(.Bottom)
```

Sizing API

WKInterfaceObject

NEW

```
func setWidth(CGFloat)
func setHeight(CGFloat)

func setRelativeWidth(CGFloat, adjustment : CGFloat)
func setRelativeHeight(CGFloat, adjustment : CGFloat)

func sizeToFitWidth()
func sizeToFitHeight()
```

Sizing API

WKInterfaceObject

NEW

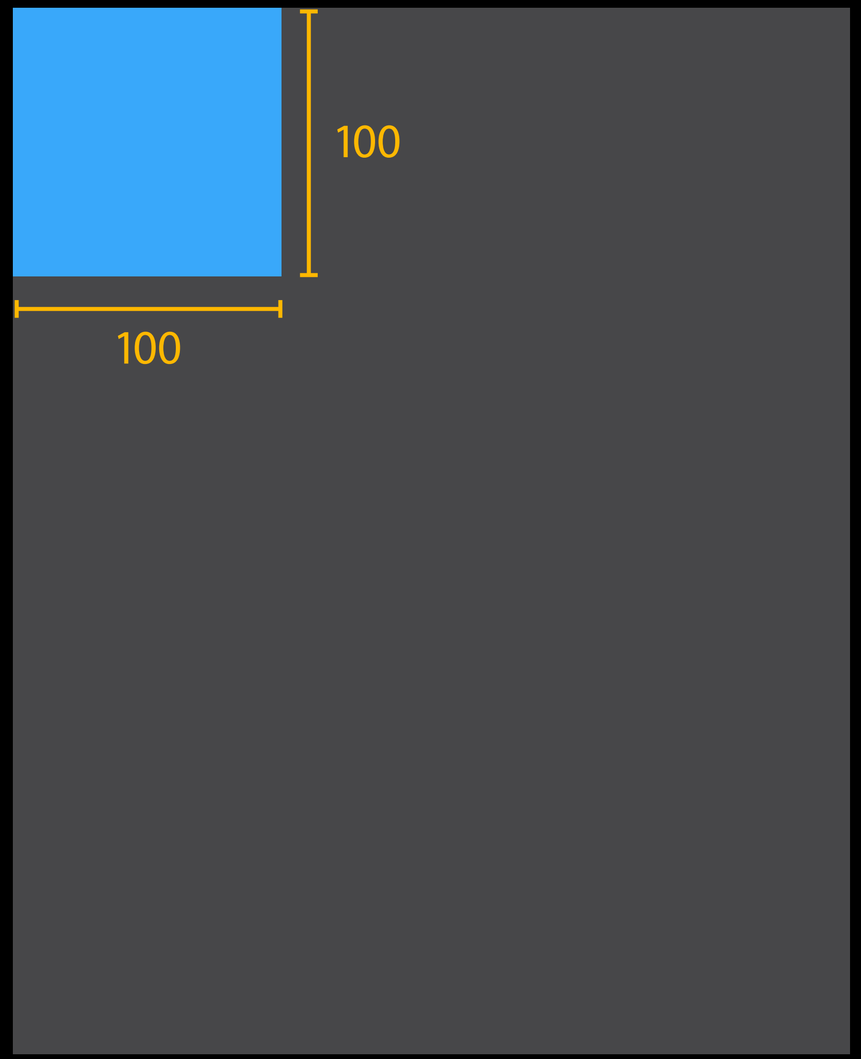
```
func setWidth(CGFloat)
func setHeight(CGFloat)
```

```
func setRelativeWidth(CGFloat, adjustment : CGFloat)
func setRelativeHeight(CGFloat, adjustment : CGFloat)
```

```
func sizeToFitWidth()
func sizeToFitHeight()
```

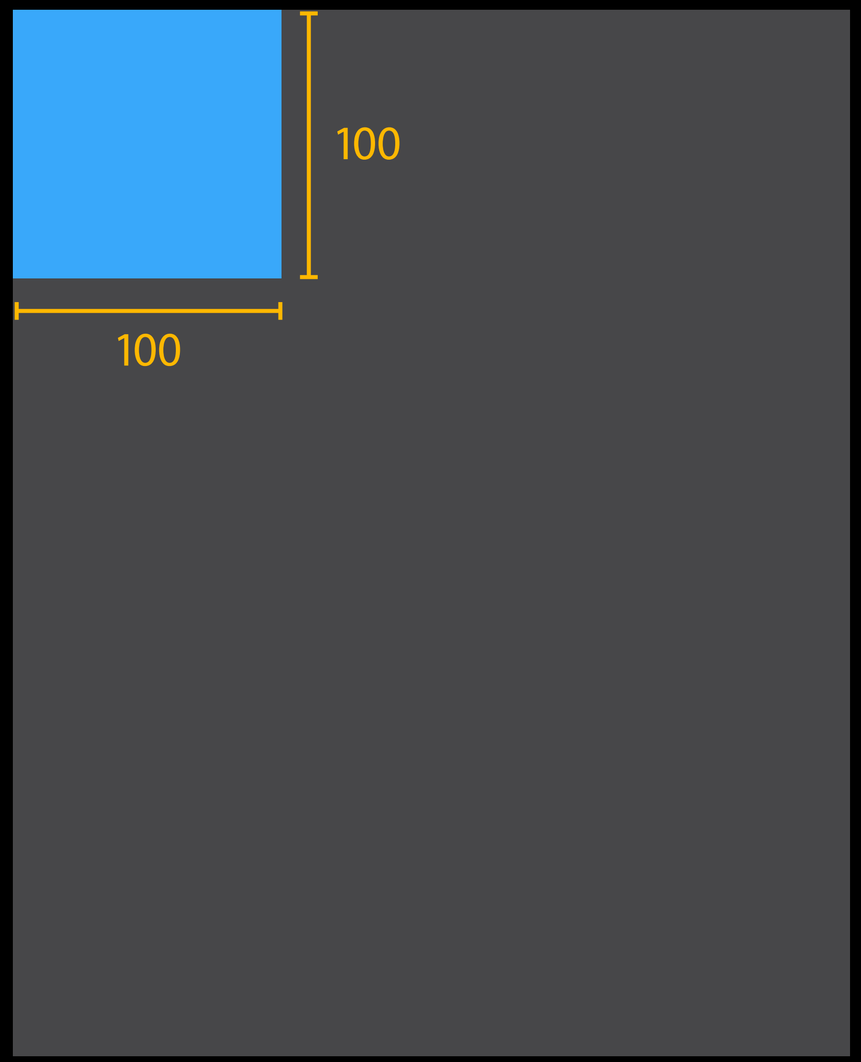
Fixed Width and Height

```
myObj.setWidth(100)
myObj.setHeight(100)
```



Fixed Width and Height

```
myObj.setWidth(100)
myObj.setHeight(100)
```

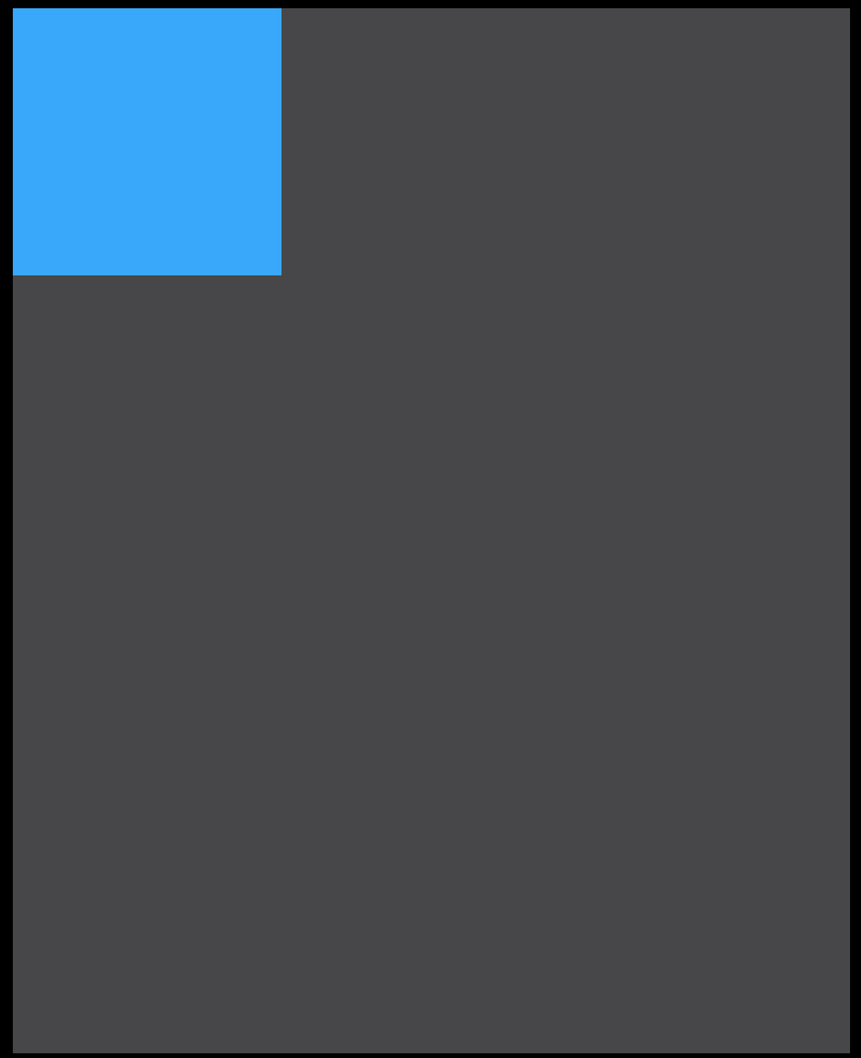


Interpreting values of zero

watchOS 1 - default to storyboard

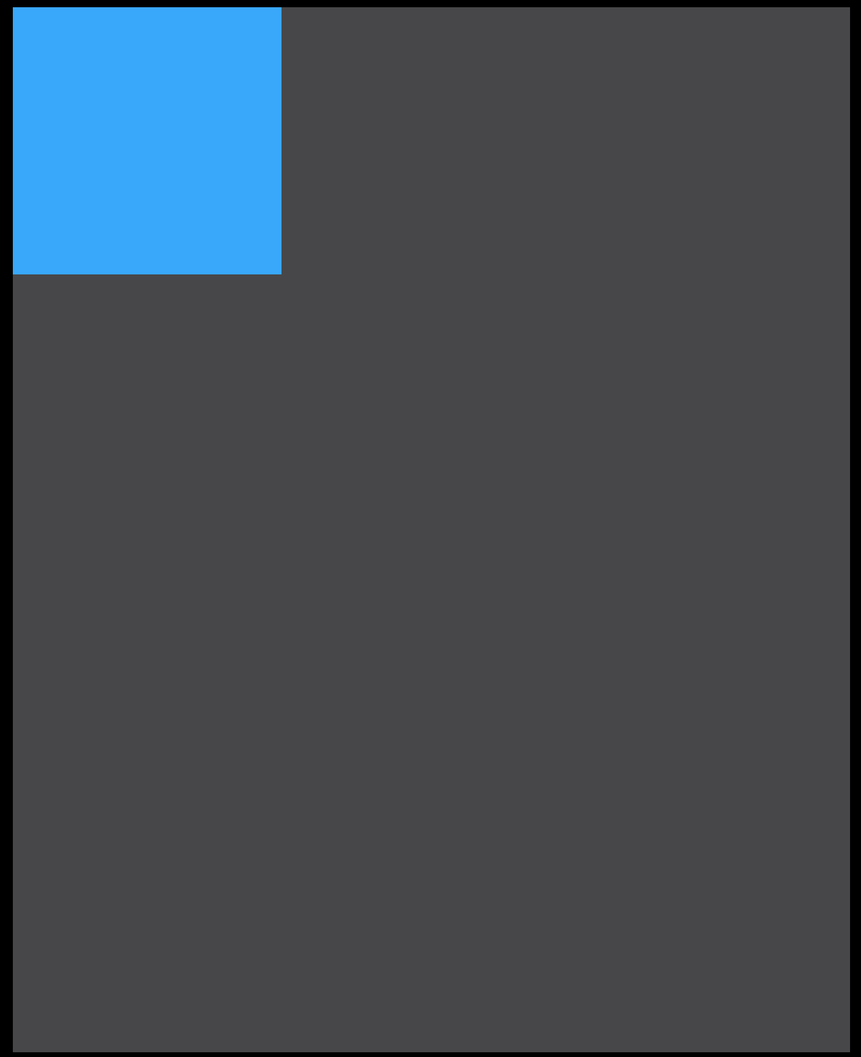
watchOS 2 - zero value NEW

Relative to Container



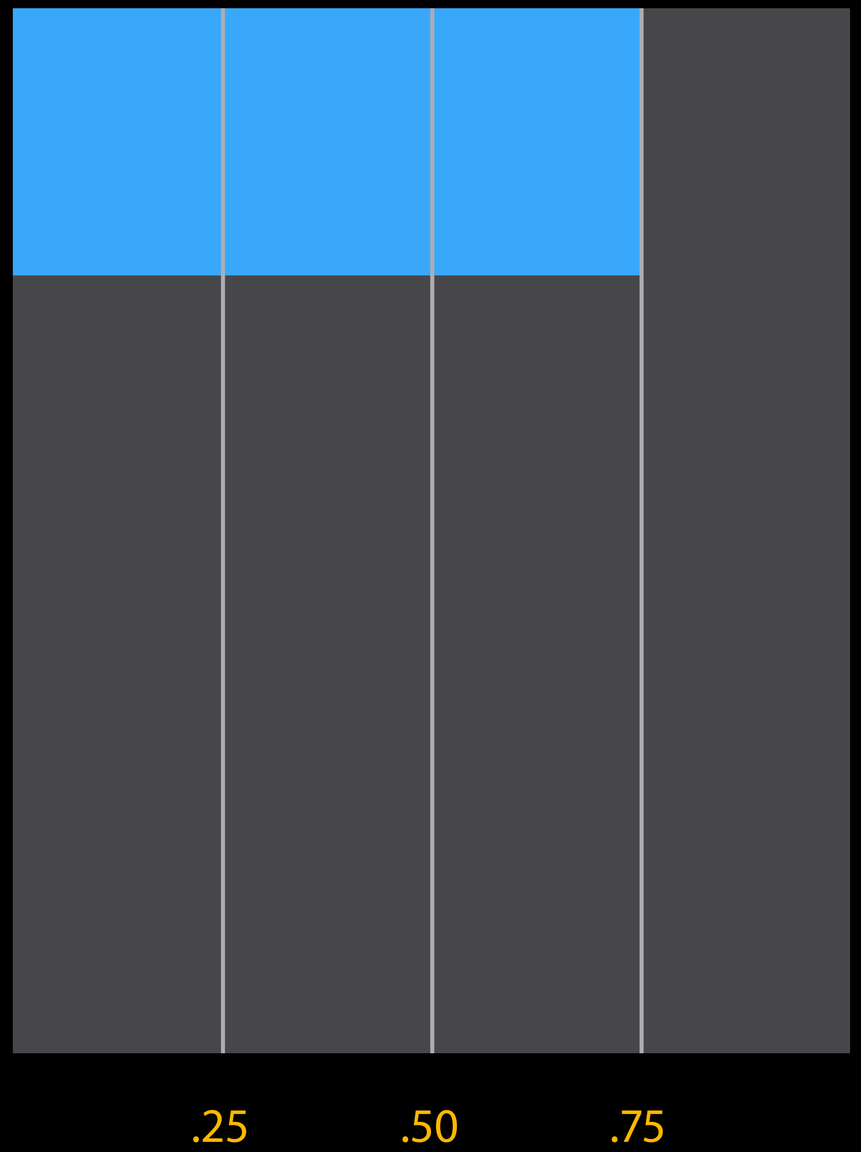
Relative to Container

```
myObj.setRelativeWidth(0.75, 0)
```



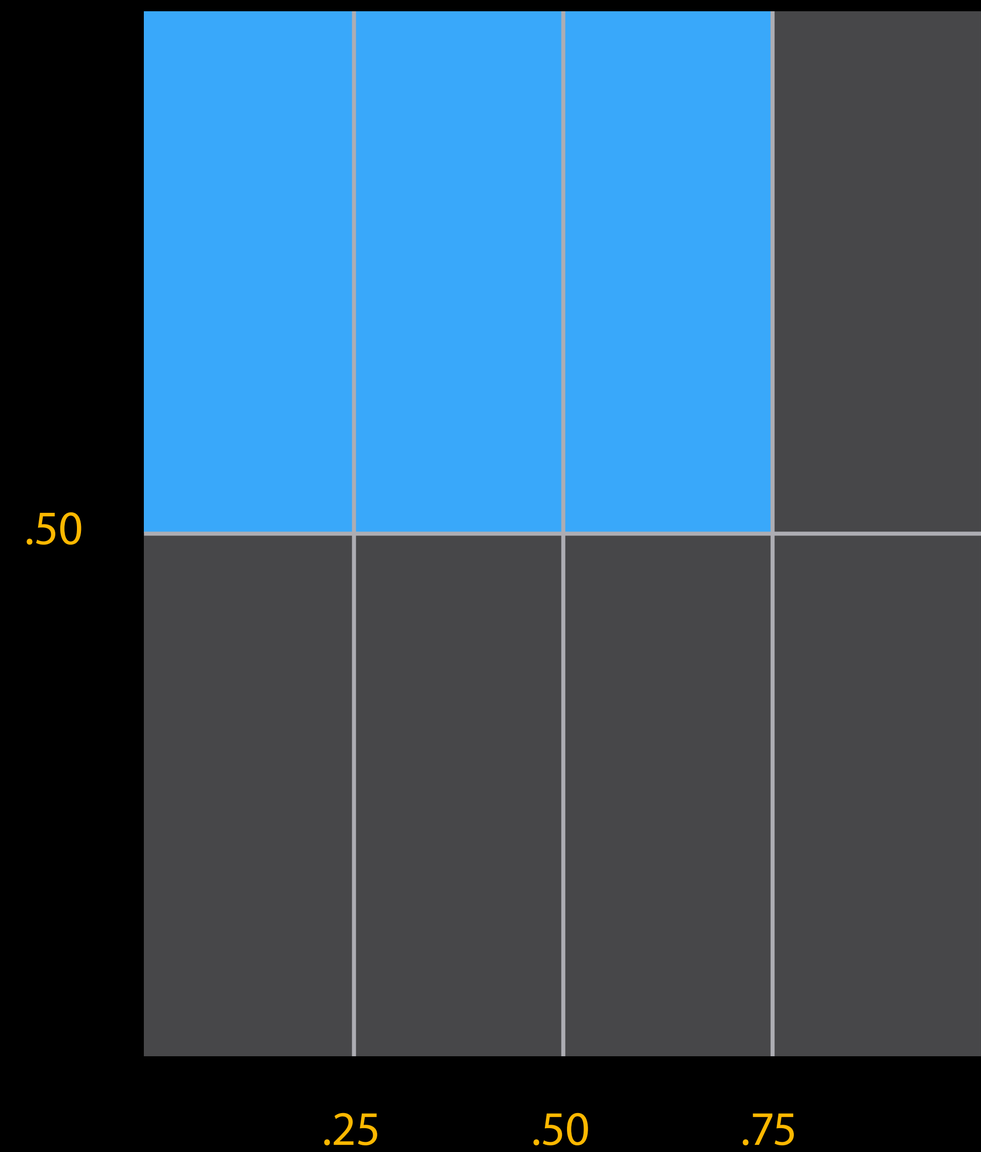
Relative to Container

```
myObj.setRelativeWidth(0.75, 0)
```



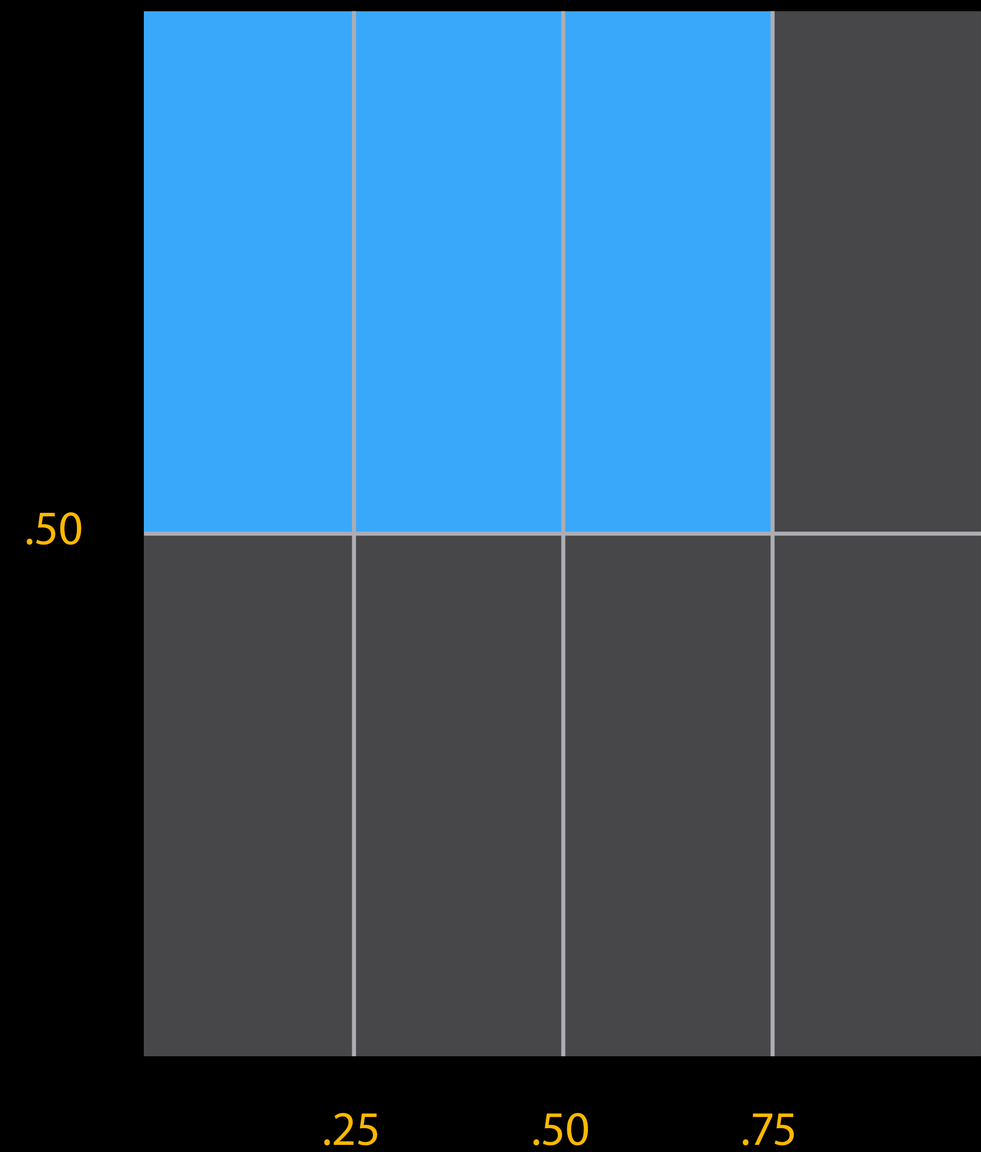
Relative to Container

```
myObj.setRelativeWidth(0.75, 0)

myObj.setRelativeHeight(0.50, 0)
```



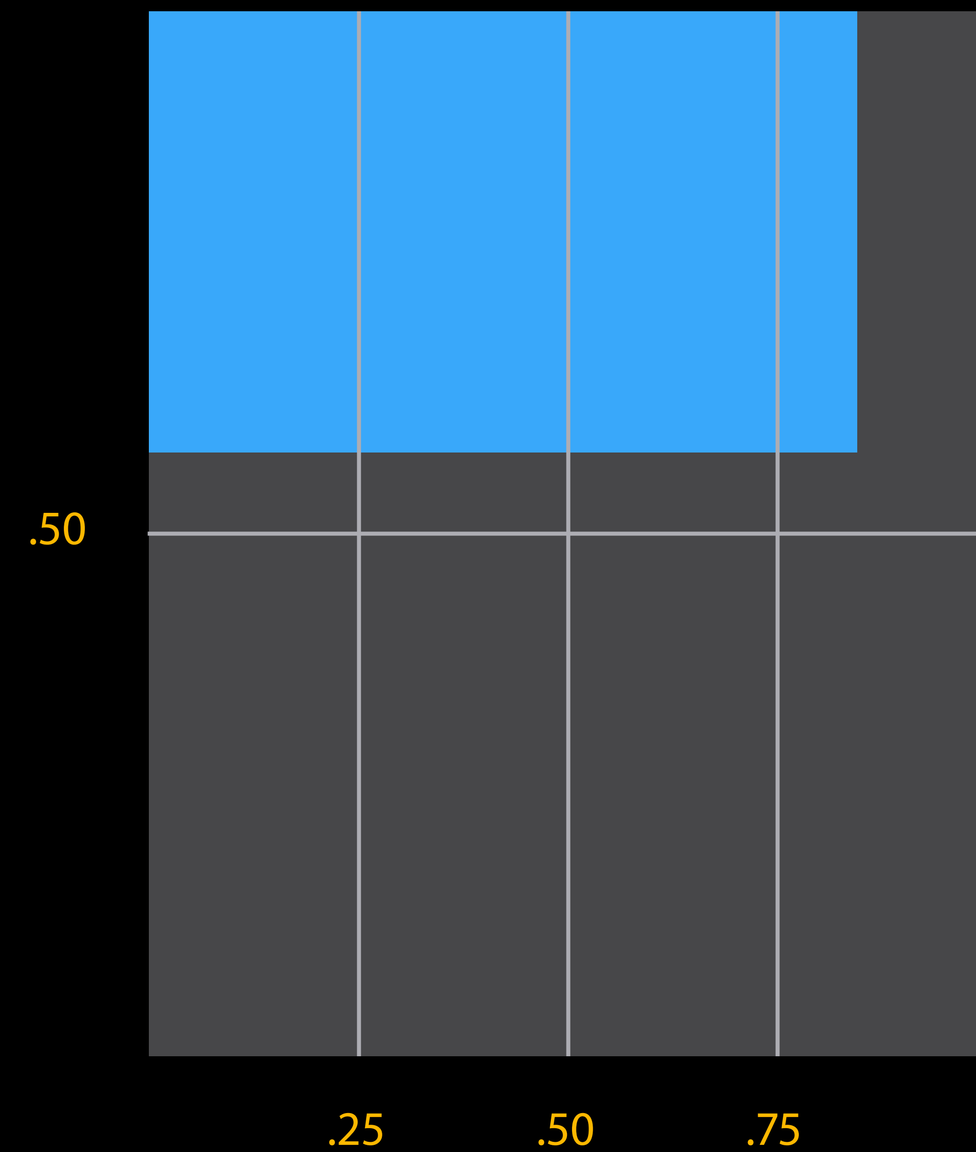
Adjustment Adds/Subtracts from Size

```
myObj.setRelativeWidth(0.75, 30)

myObj.setRelativeHeight(0.50, -30)
```



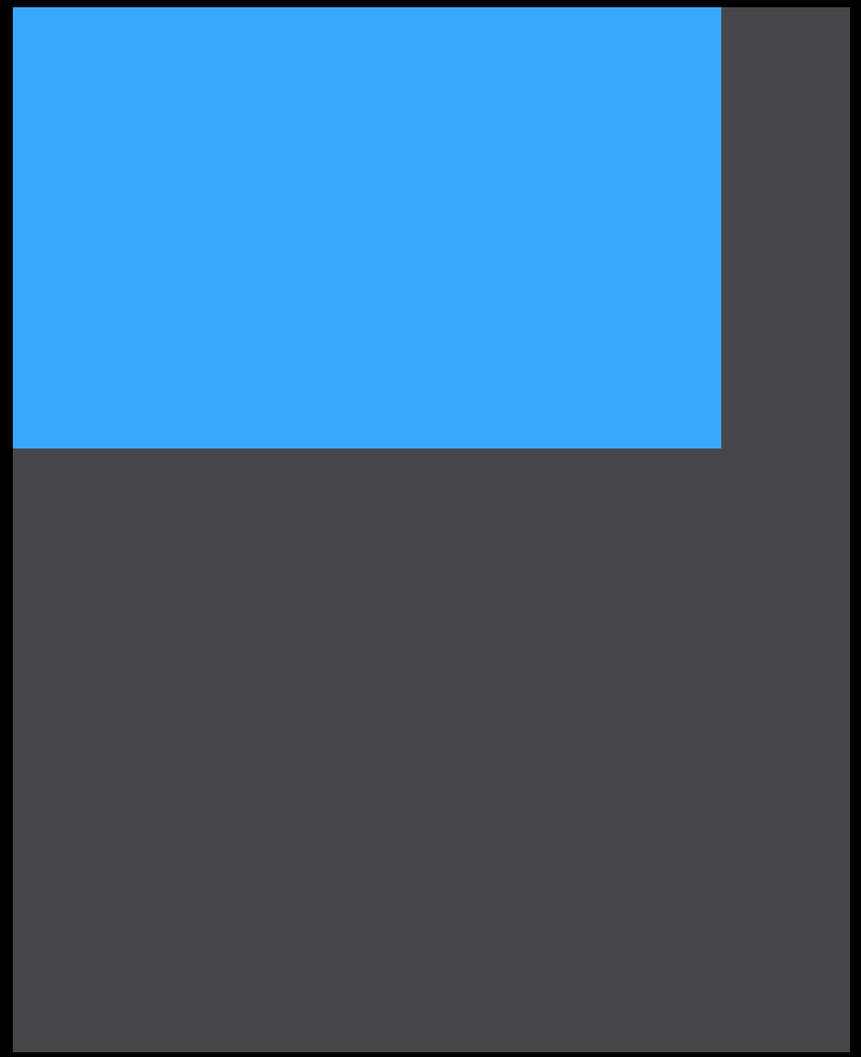
Adjustment Adds/Subtracts from Size

```
myObj.setRelativeWidth(0.75, 30)

myObj.setRelativeHeight(0.50, -30)
```



Sized-to-Fit Content

```
myObj.sizeToFitWidth()
myObj.sizeToFitHeight()
```



Sized-to-Fit Content

```
myObj.sizeToFitWidth()
myObj.sizeToFitHeight()
```

Sized-to-Fit Content

```
myObj.sizeToFitWidth()
myObj.sizeToFitHeight()
```

Group Elements

Fine tuning your layouts

WKInterfaceGroup

WKInterfaceGroup

Container without default content

WKInterfaceGroup

Container without default content

Tool for arranging elements

WKInterfaceGroup

Container without default content

Tool for arranging elements

- Vertical or horizontal flow

WKInterfaceGroup

Container without default content

Tool for arranging elements

- Vertical or horizontal flow
- Nesting (including other groups)

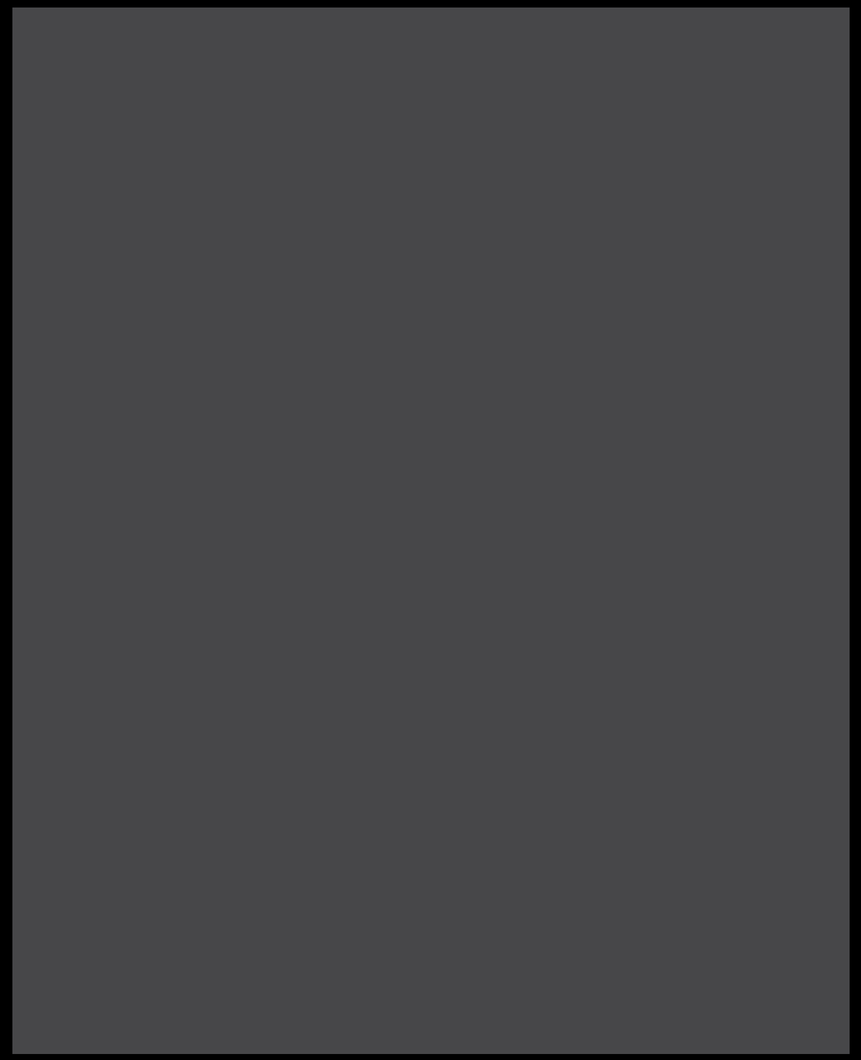
WKInterfaceGroup

Container without default content

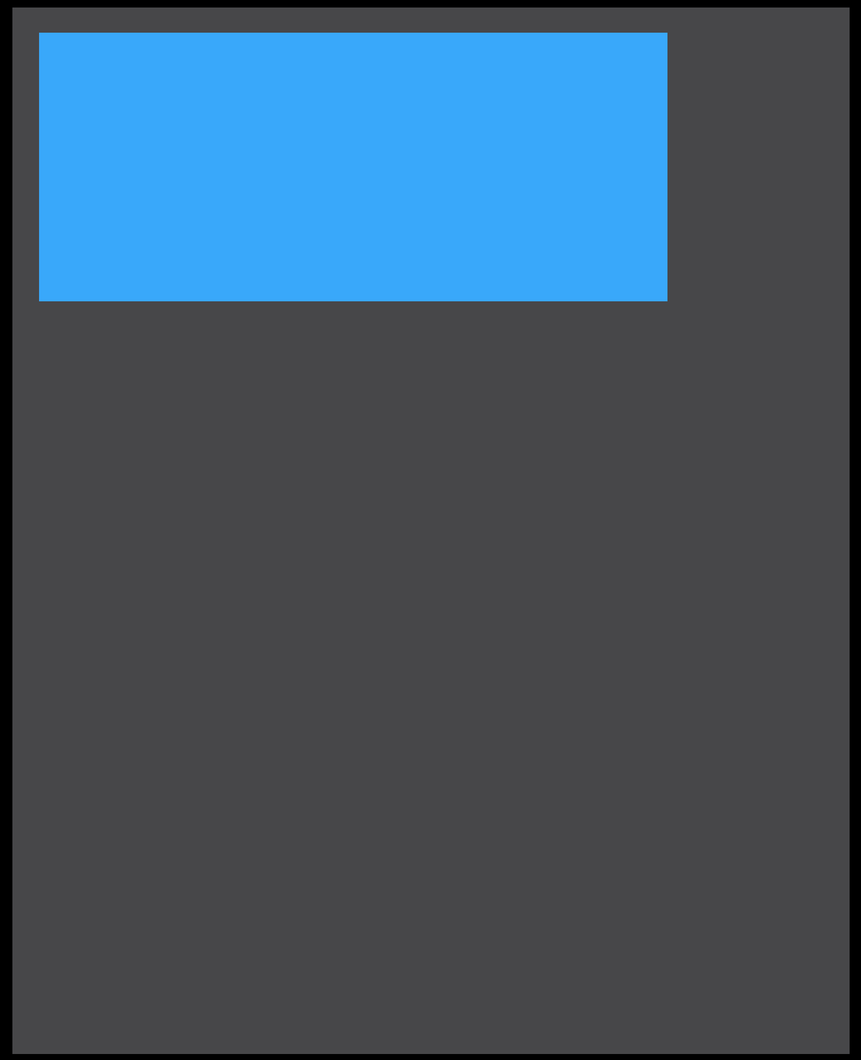
Tool for arranging elements

- Vertical or horizontal flow
- Nesting (including other groups)
- Fine control of alignment and sizing

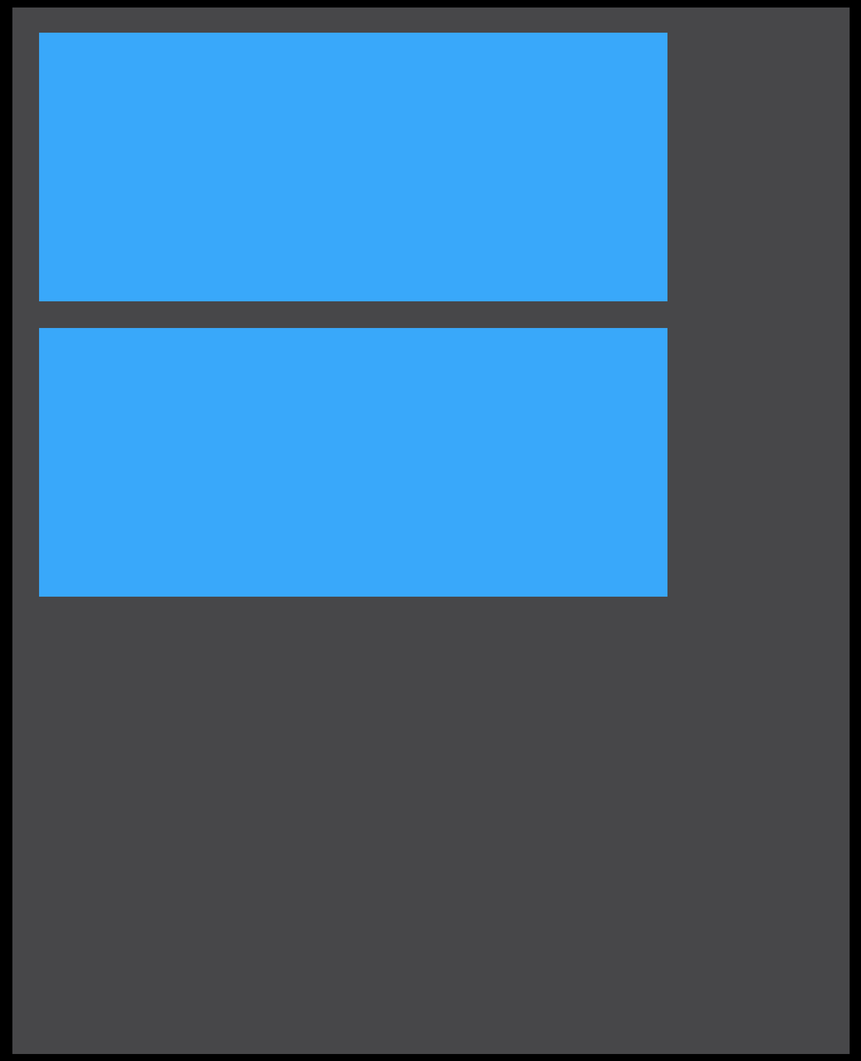
Adding Horizontal Flow



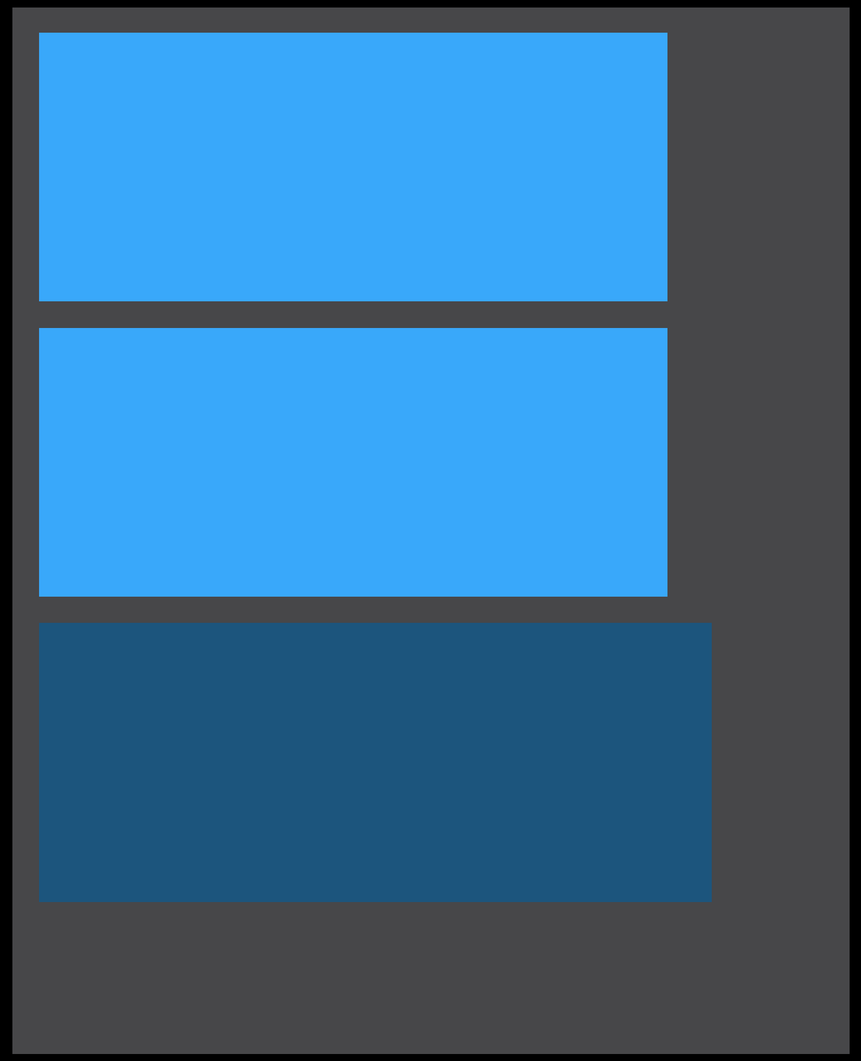
Adding Horizontal Flow



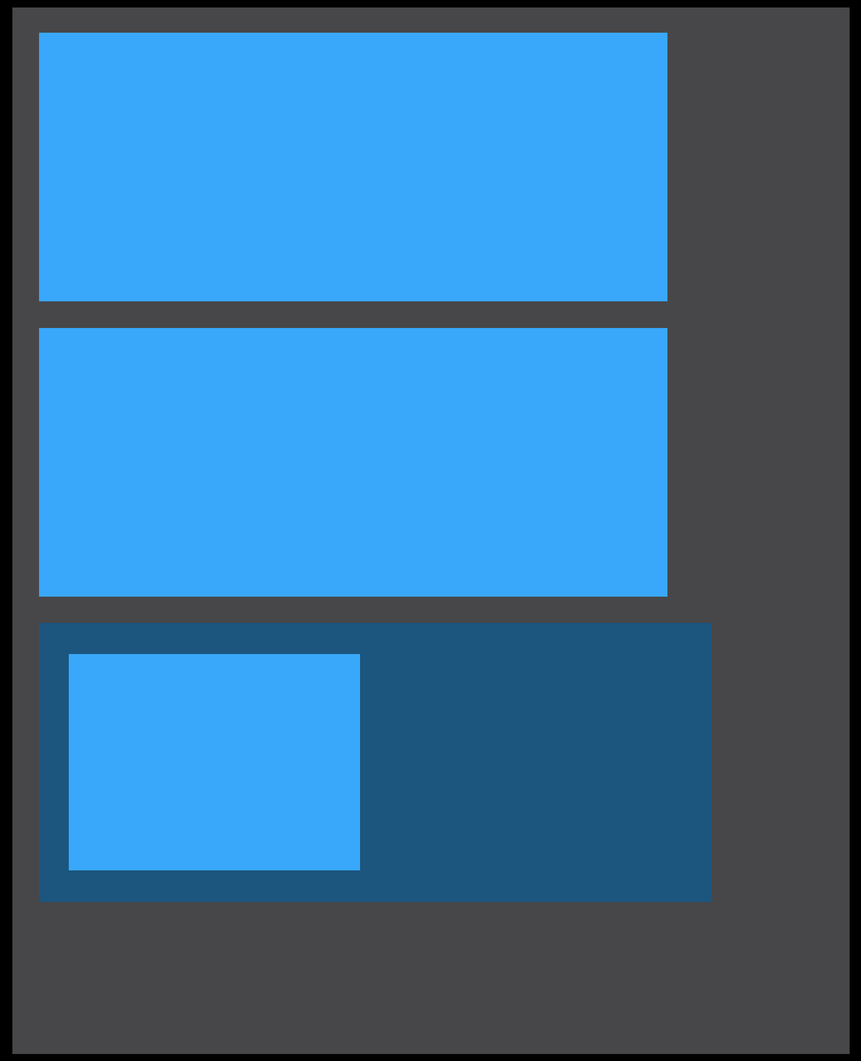
Adding Horizontal Flow



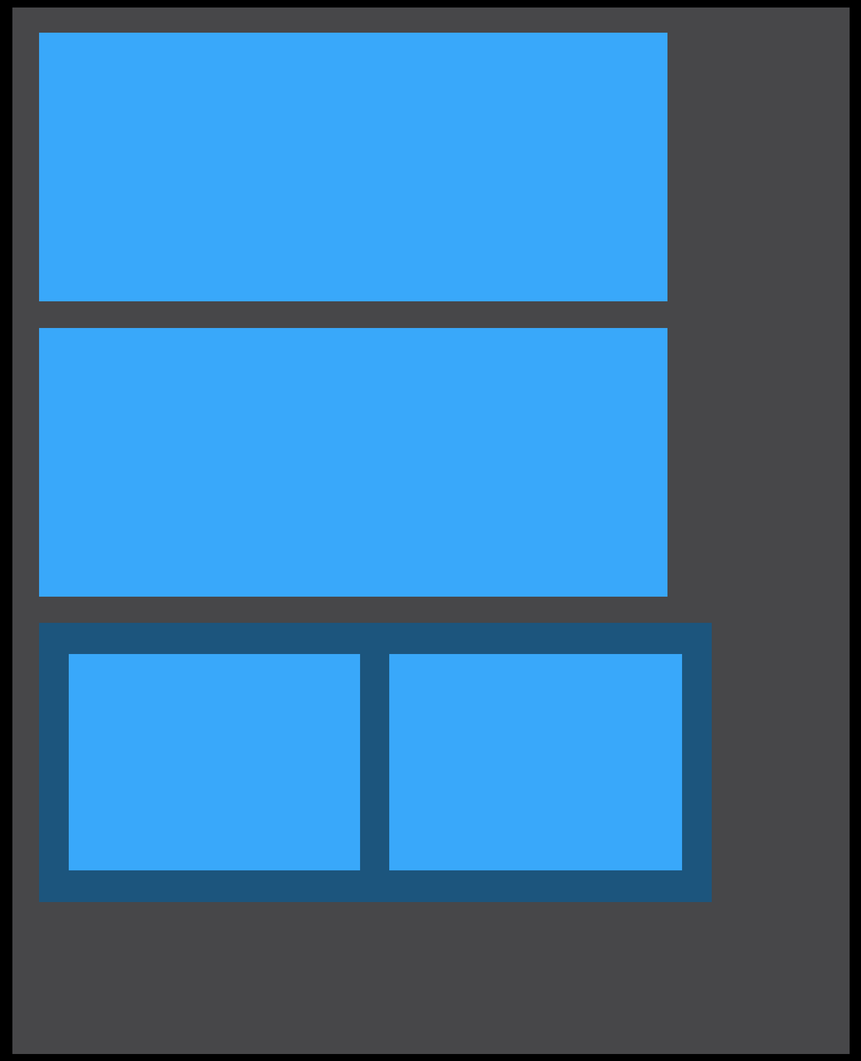
Adding Horizontal Flow



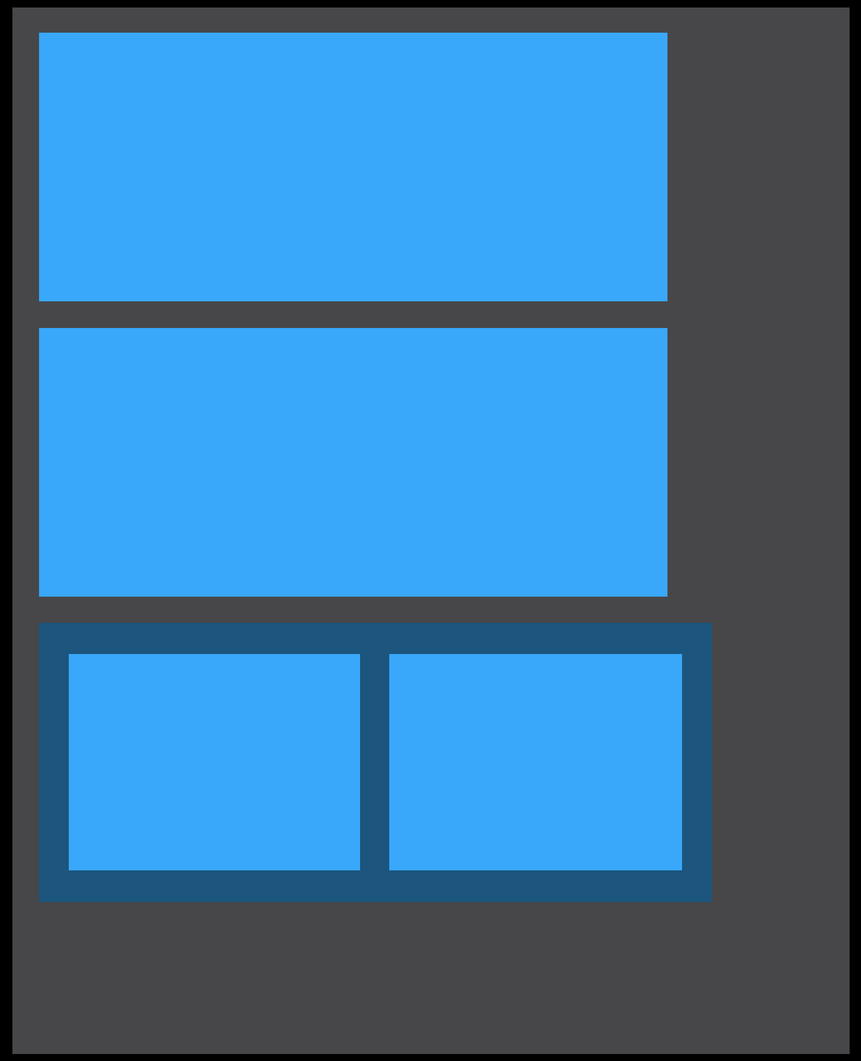
Adding Horizontal Flow



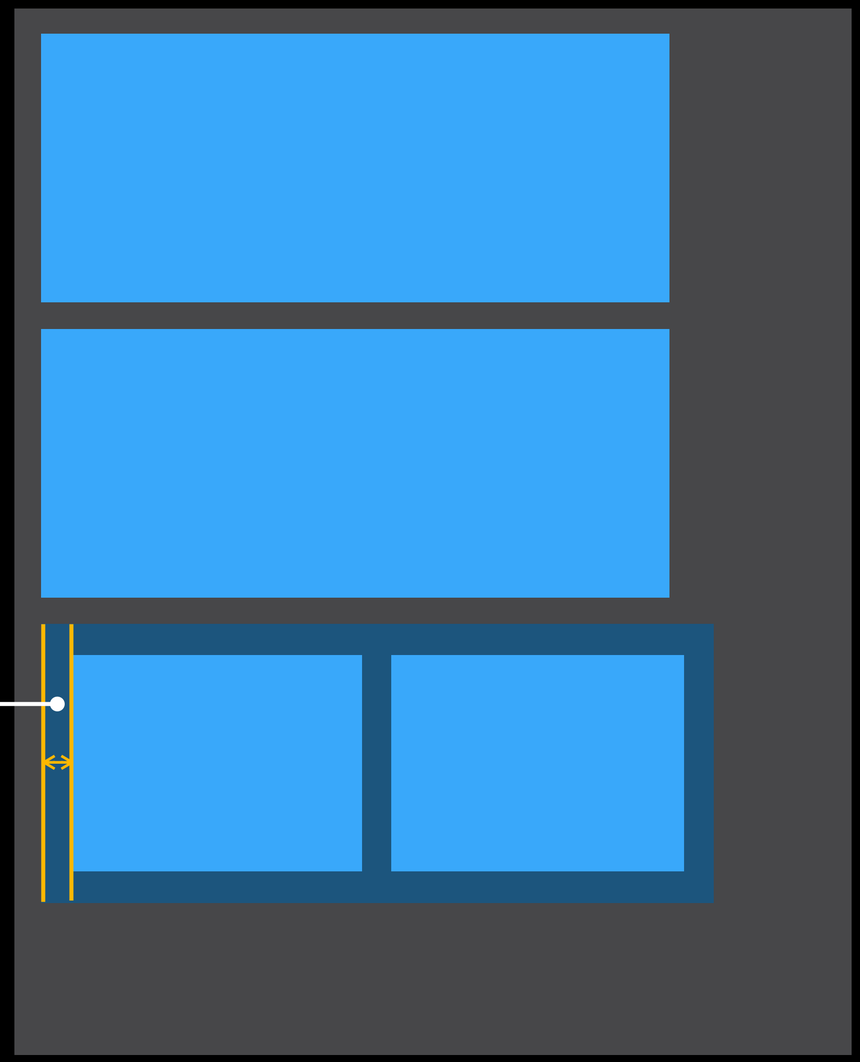
Adding Horizontal Flow



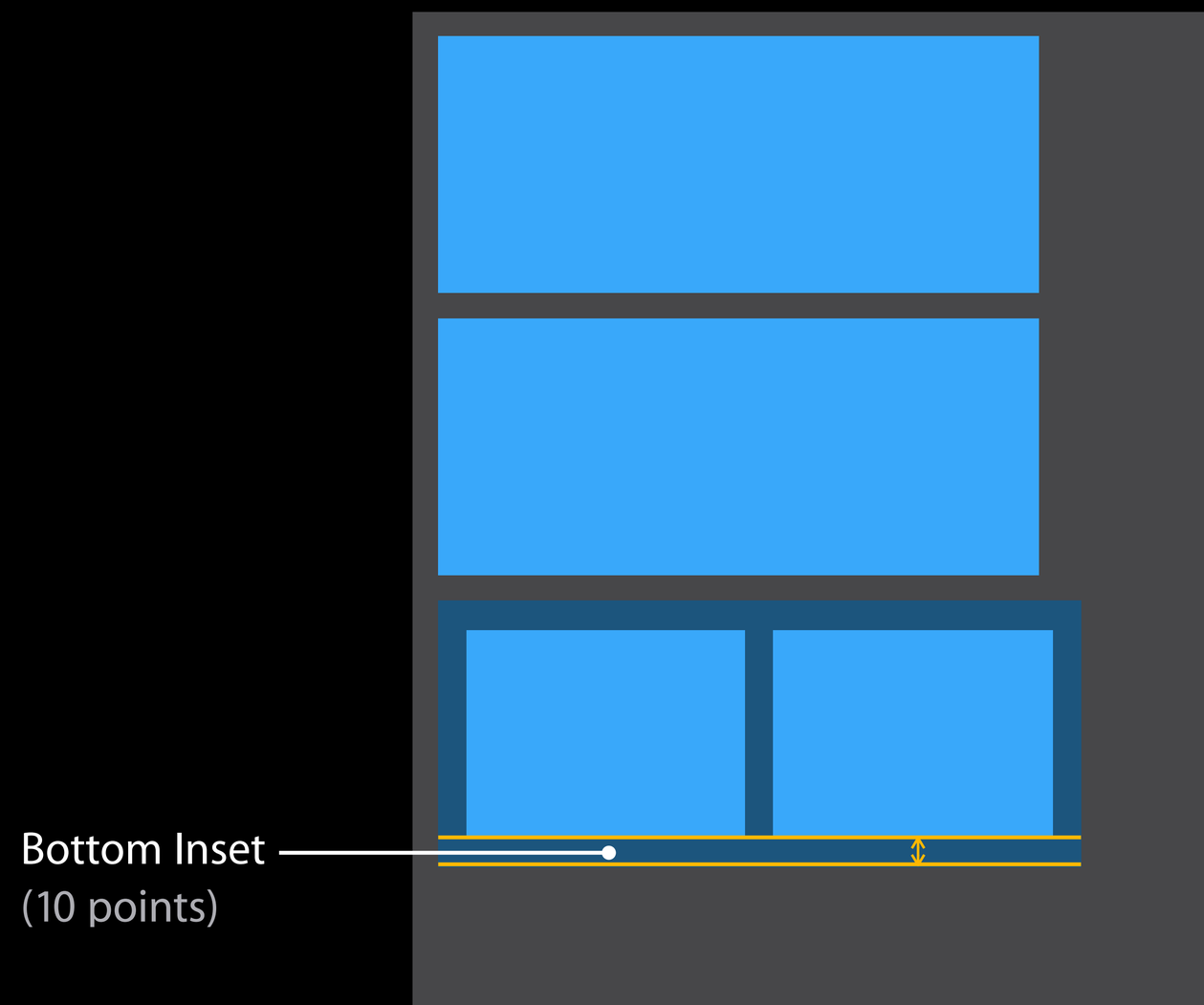
Insets and Spacing



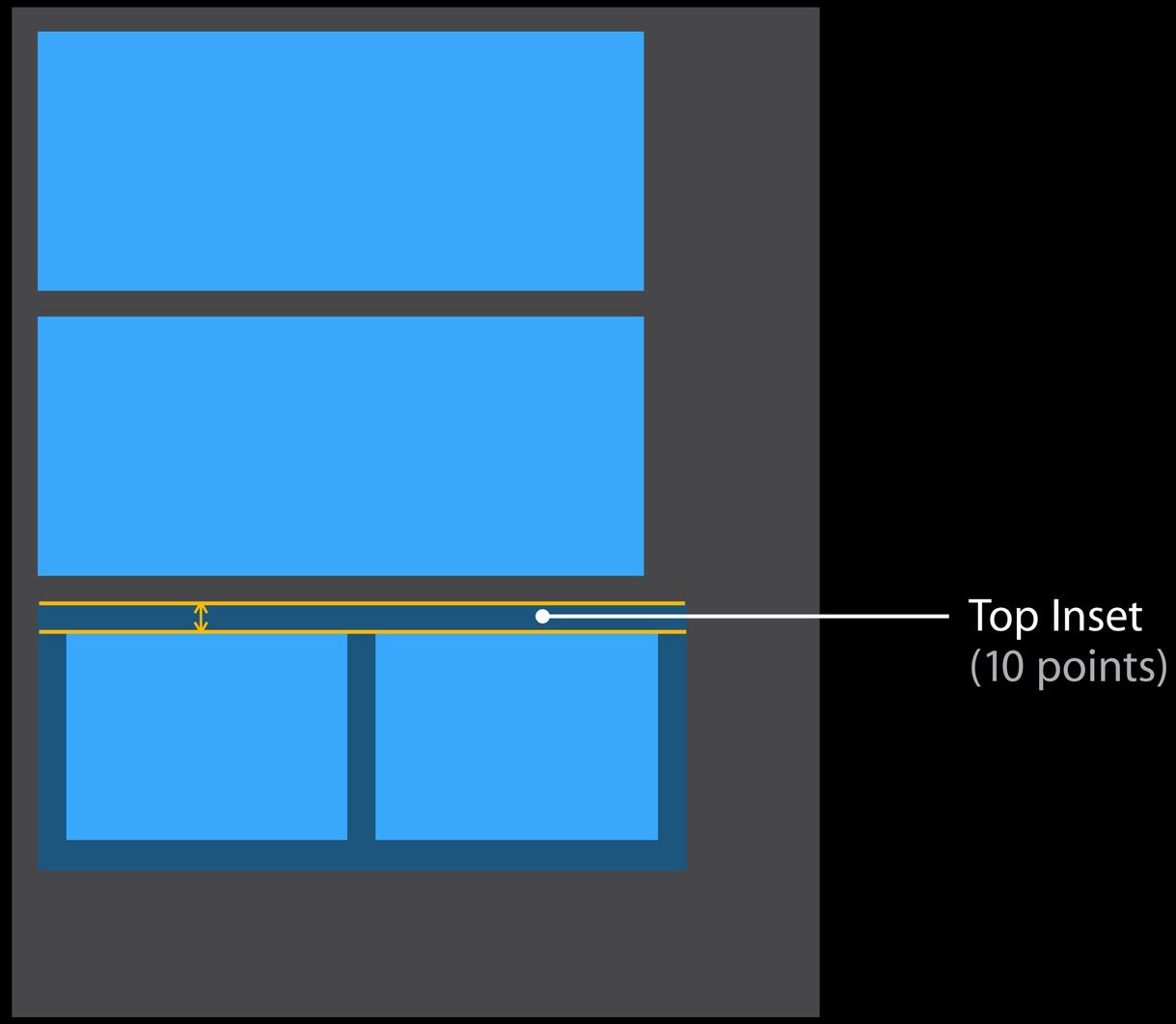
Insets and Spacing



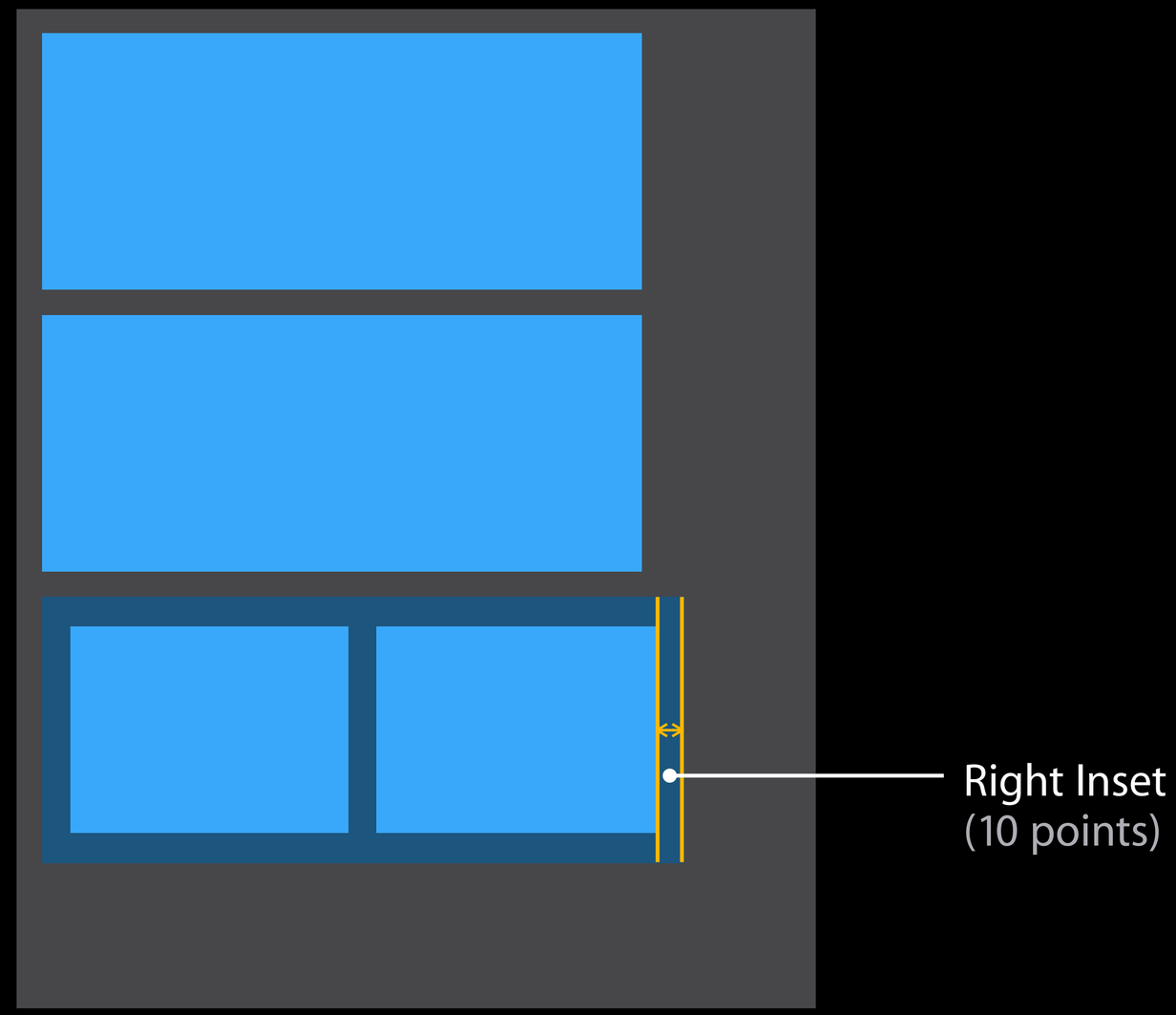
Insets and Spacing



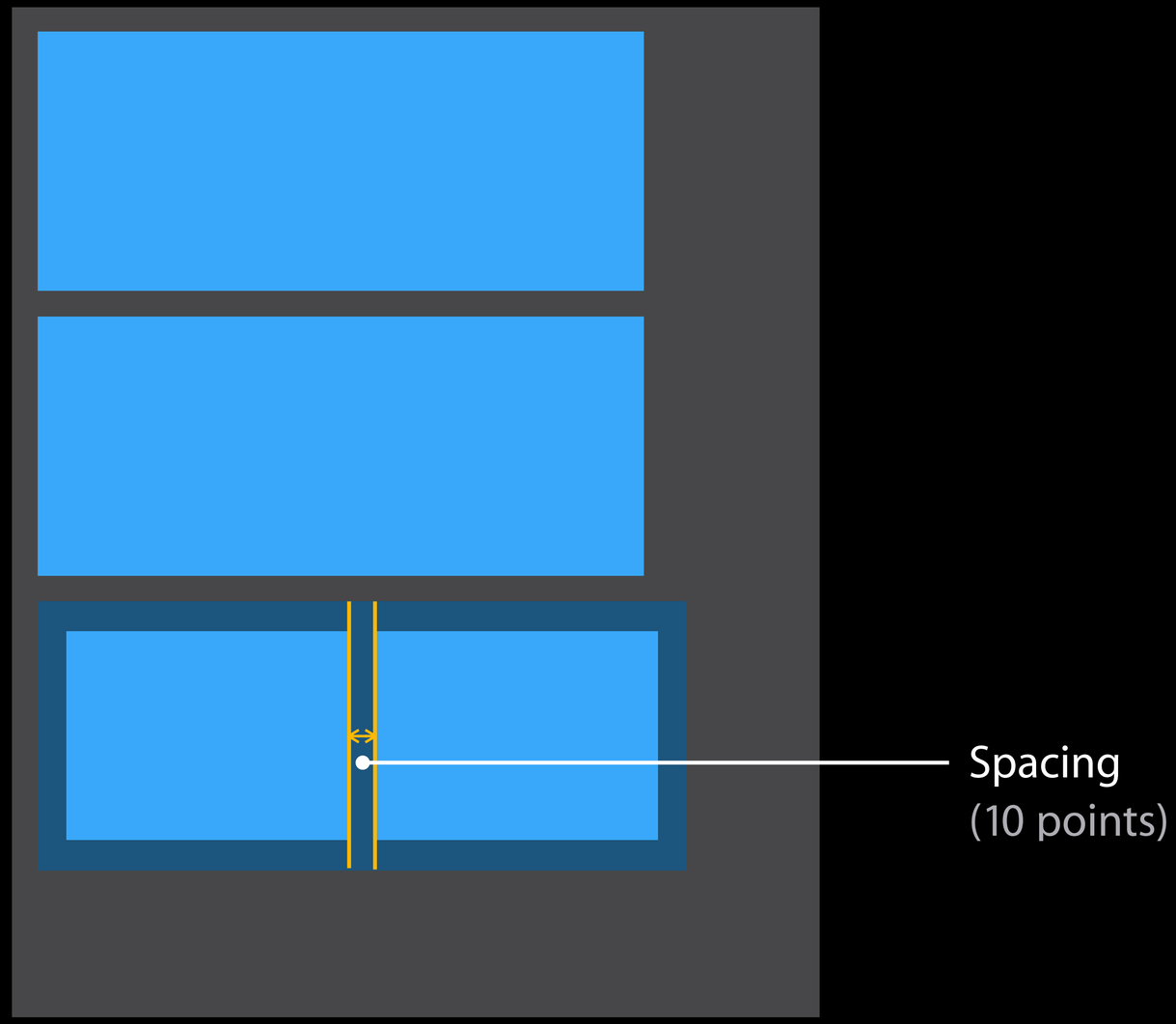
Insets and Spacing



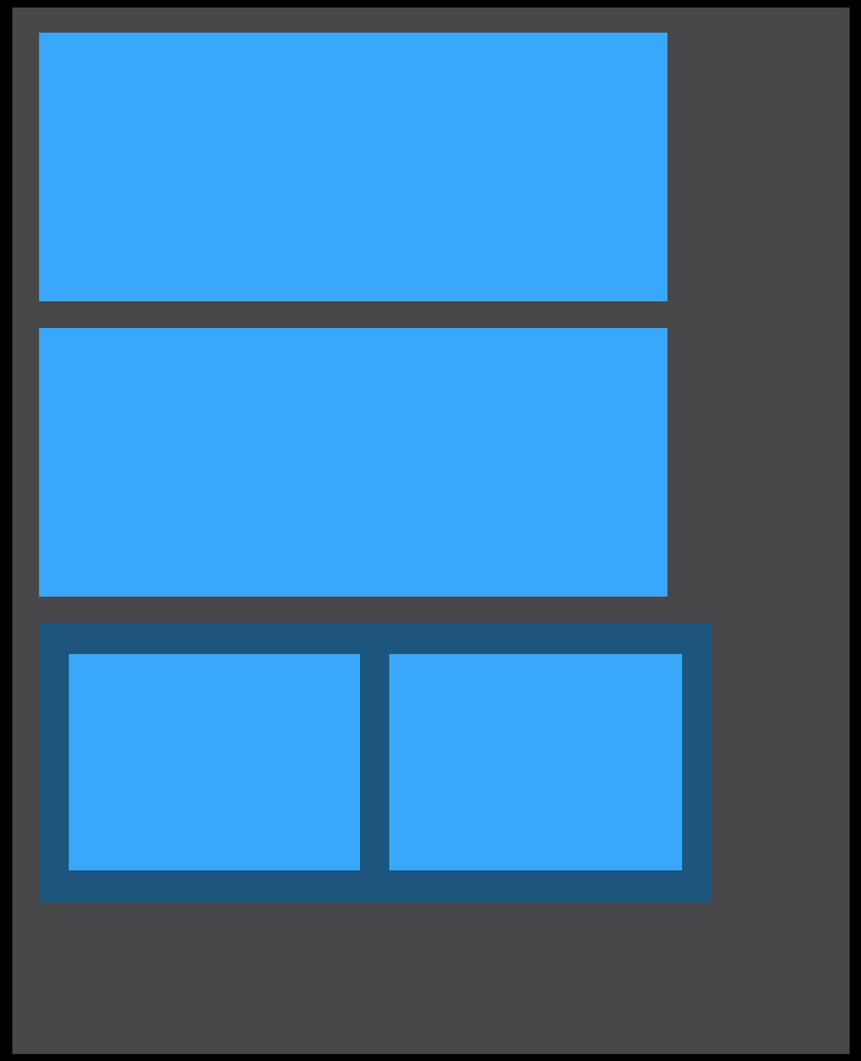
Insets and Spacing



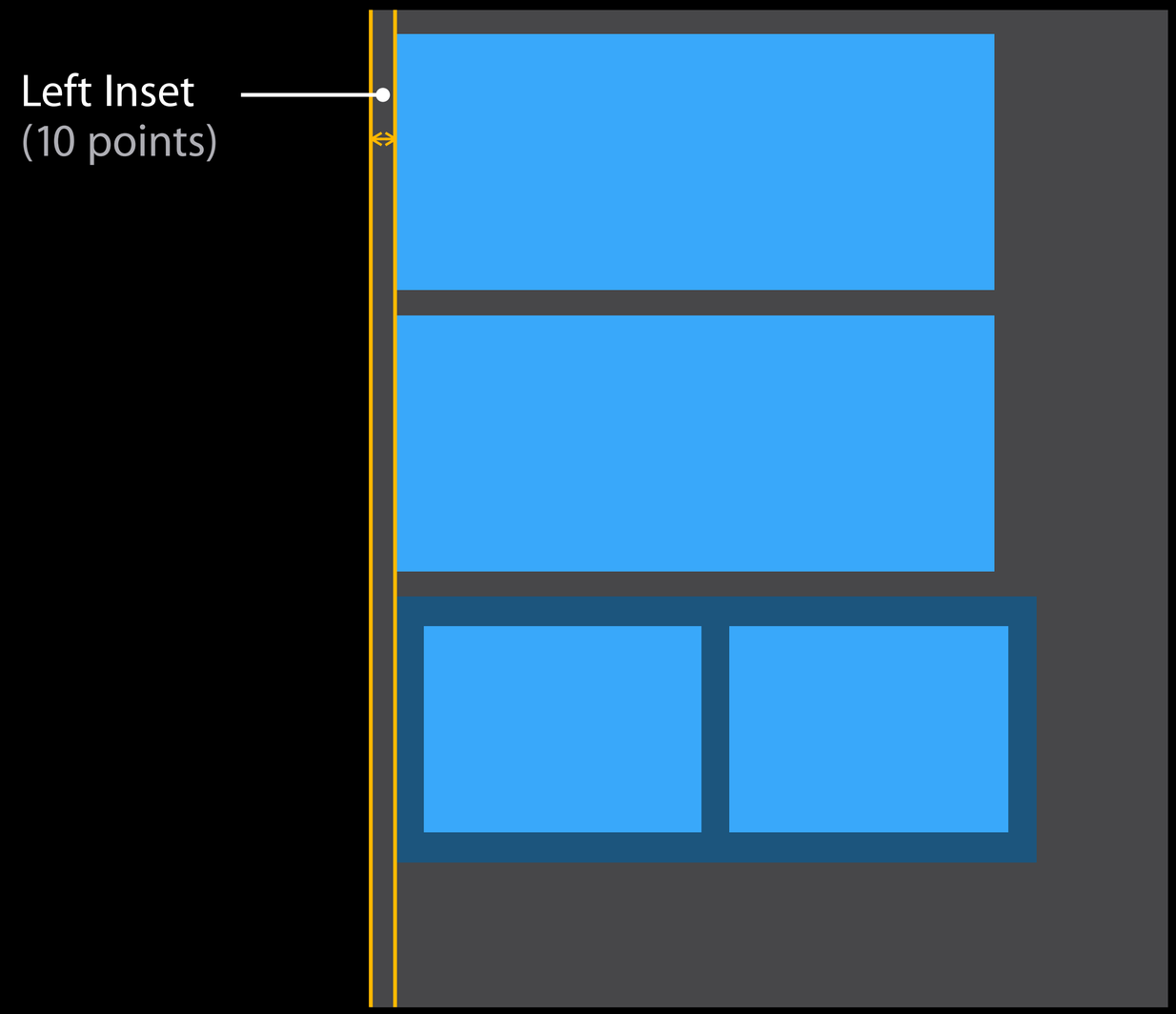
Insets and Spacing



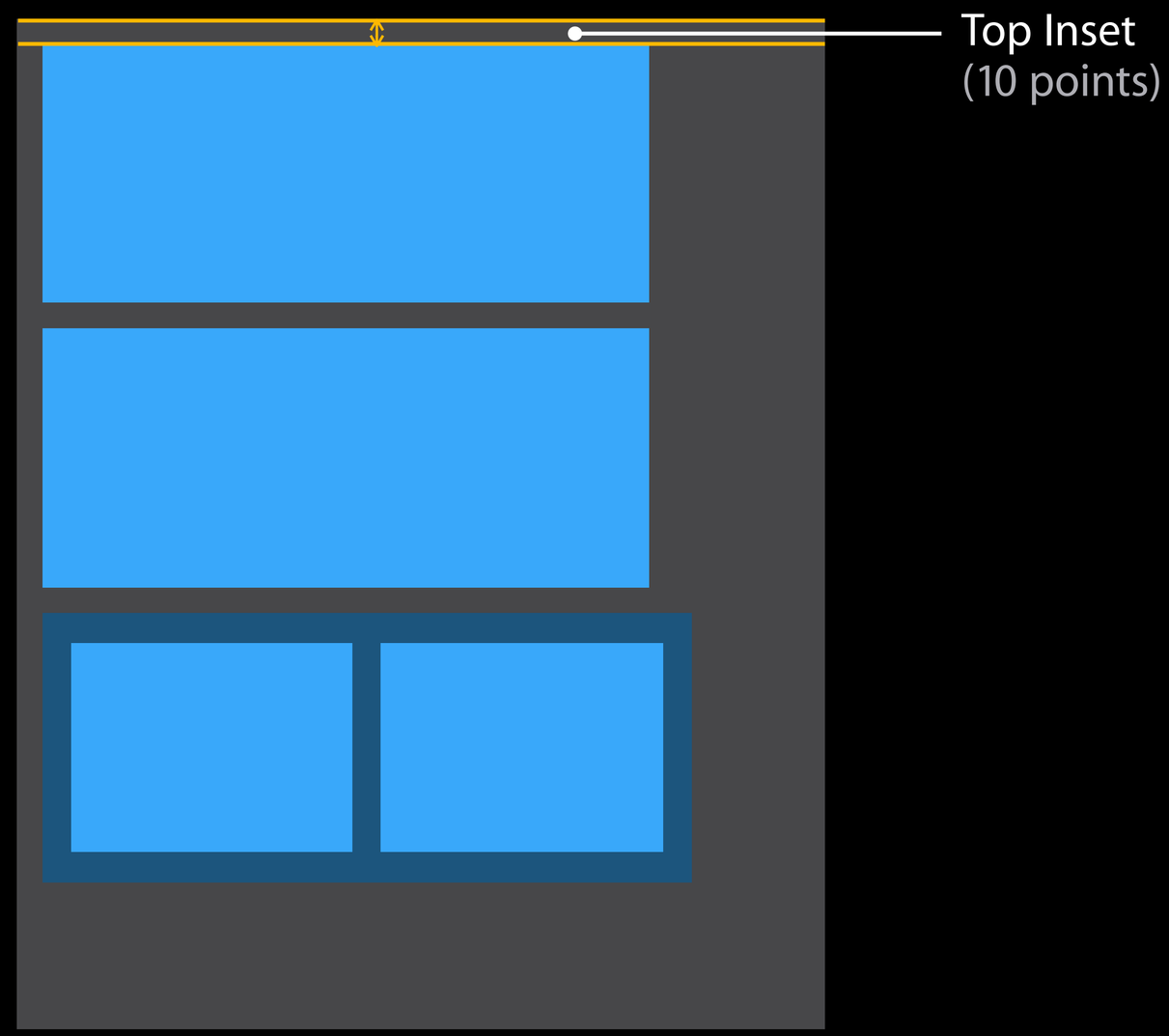
Can Also Be Set at Top Level



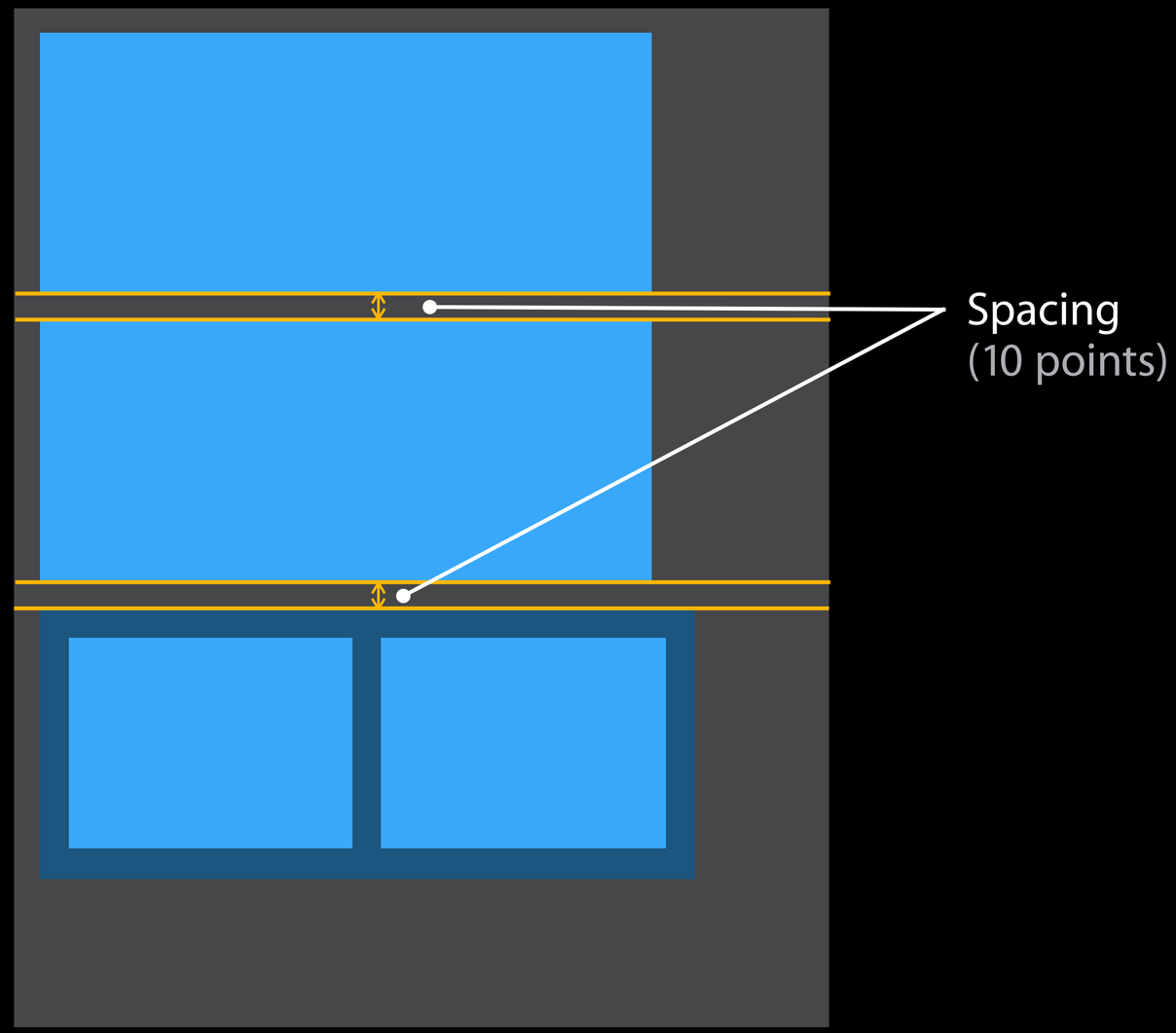
Can Also Be Set at Top Level



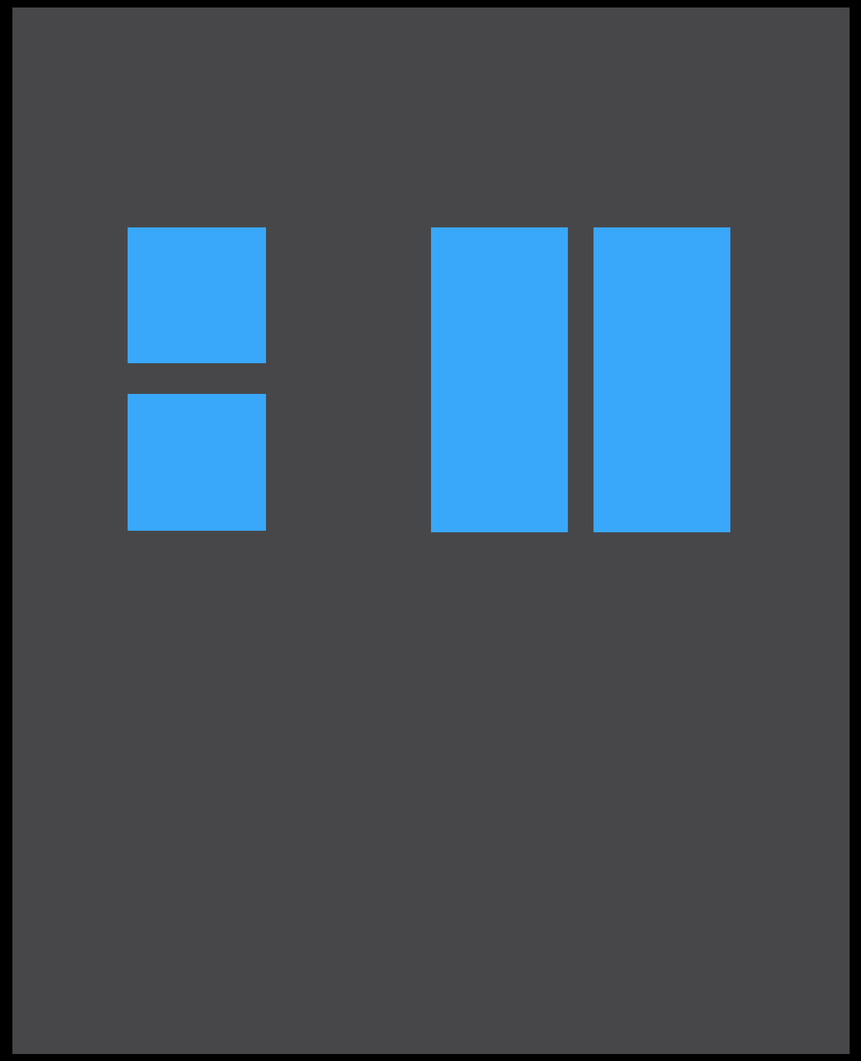
Can Also Be Set at Top Level



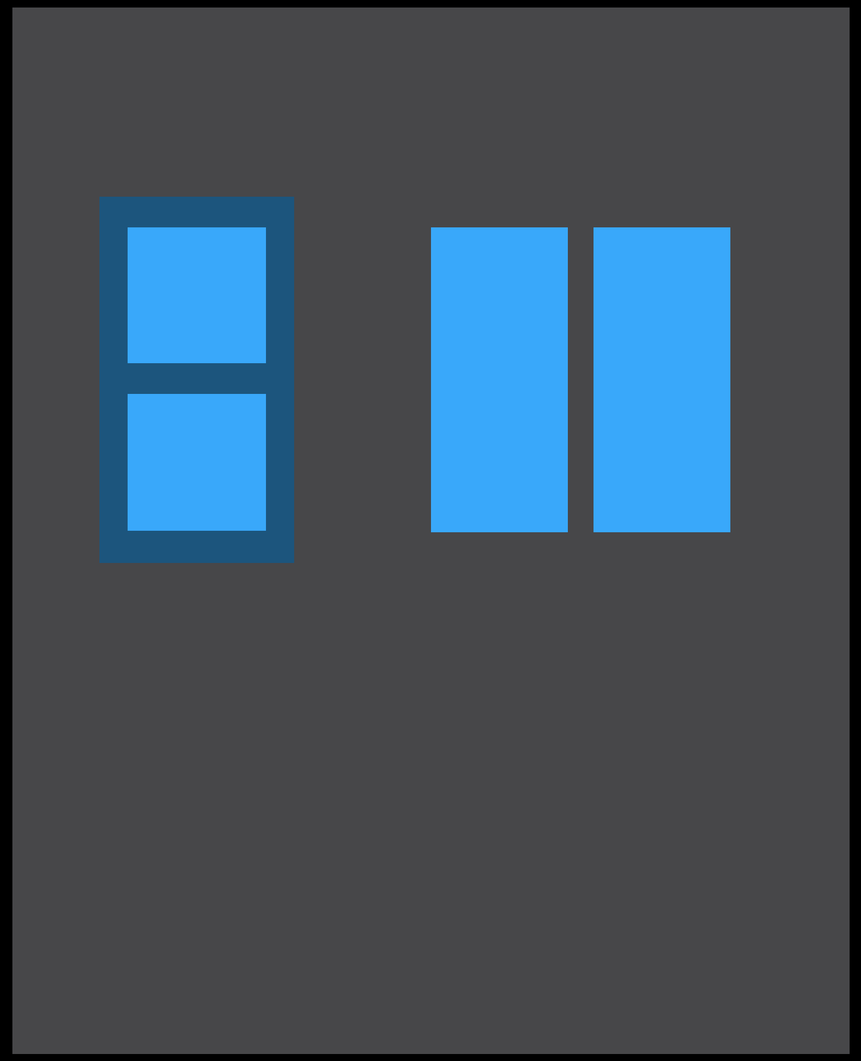
Can Also Be Set at Top Level



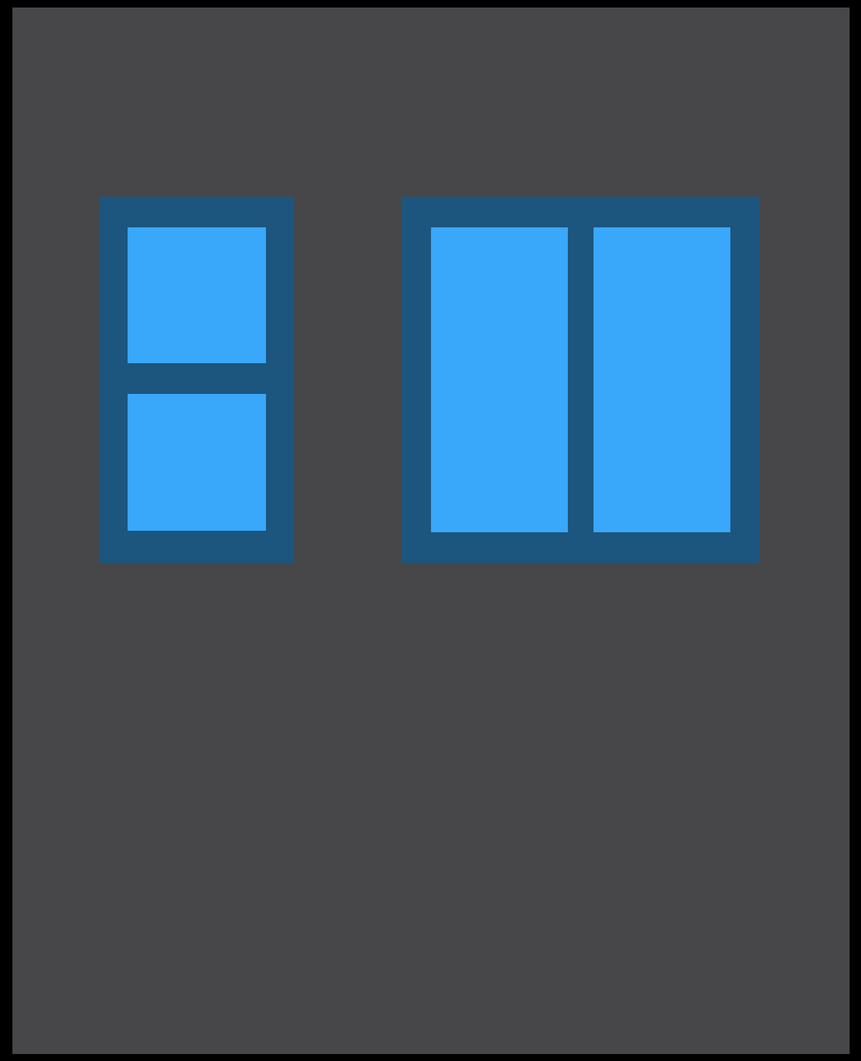
Group Nesting for Complex Layouts



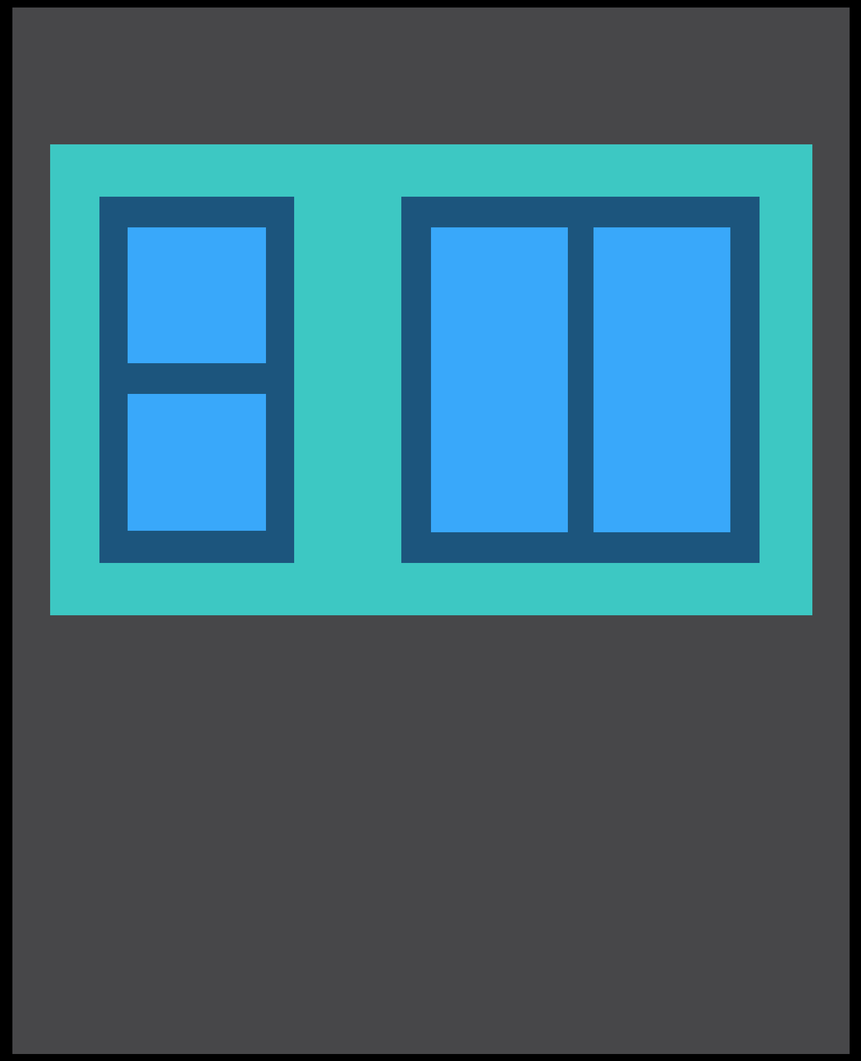
Group Nesting for Complex Layouts



Group Nesting for Complex Layouts



Group Nesting for Complex Layouts



Layouts in WKRecipes

WKRecipes

Recipe Viewer

WKRecipes

Ingredients

WKRecipes

Servings

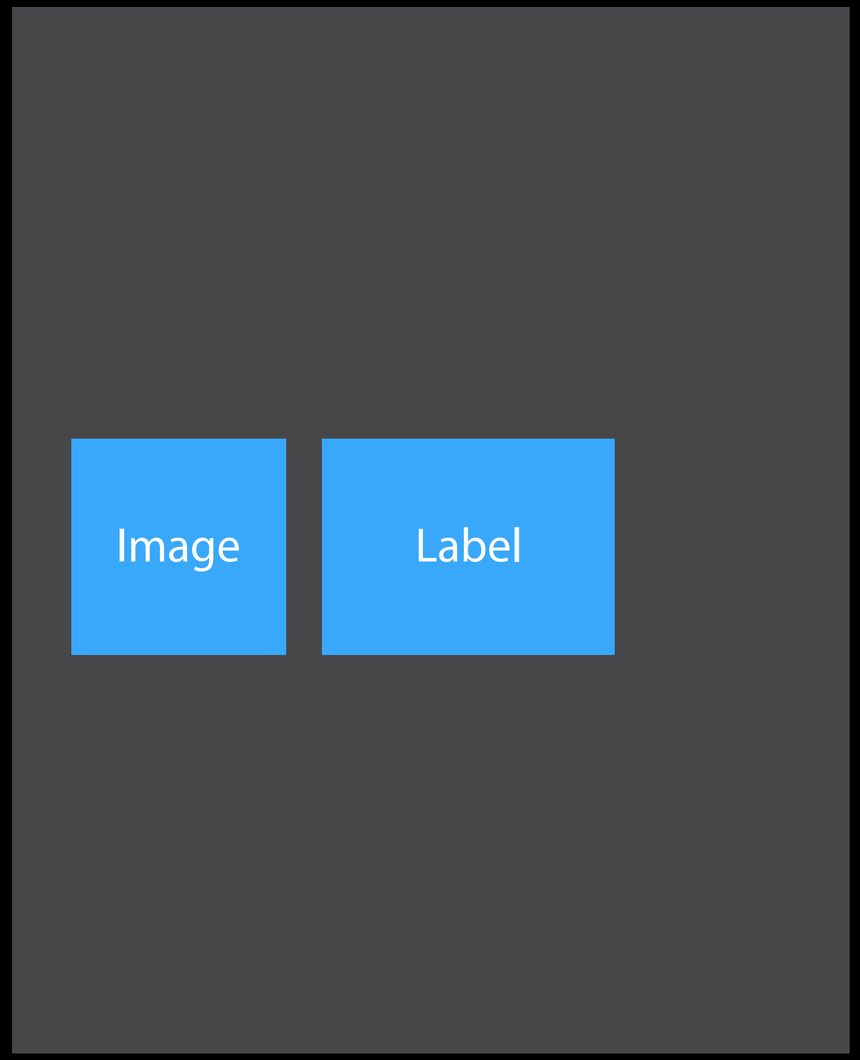
Let’s Explore Three Layouts

Table Rows

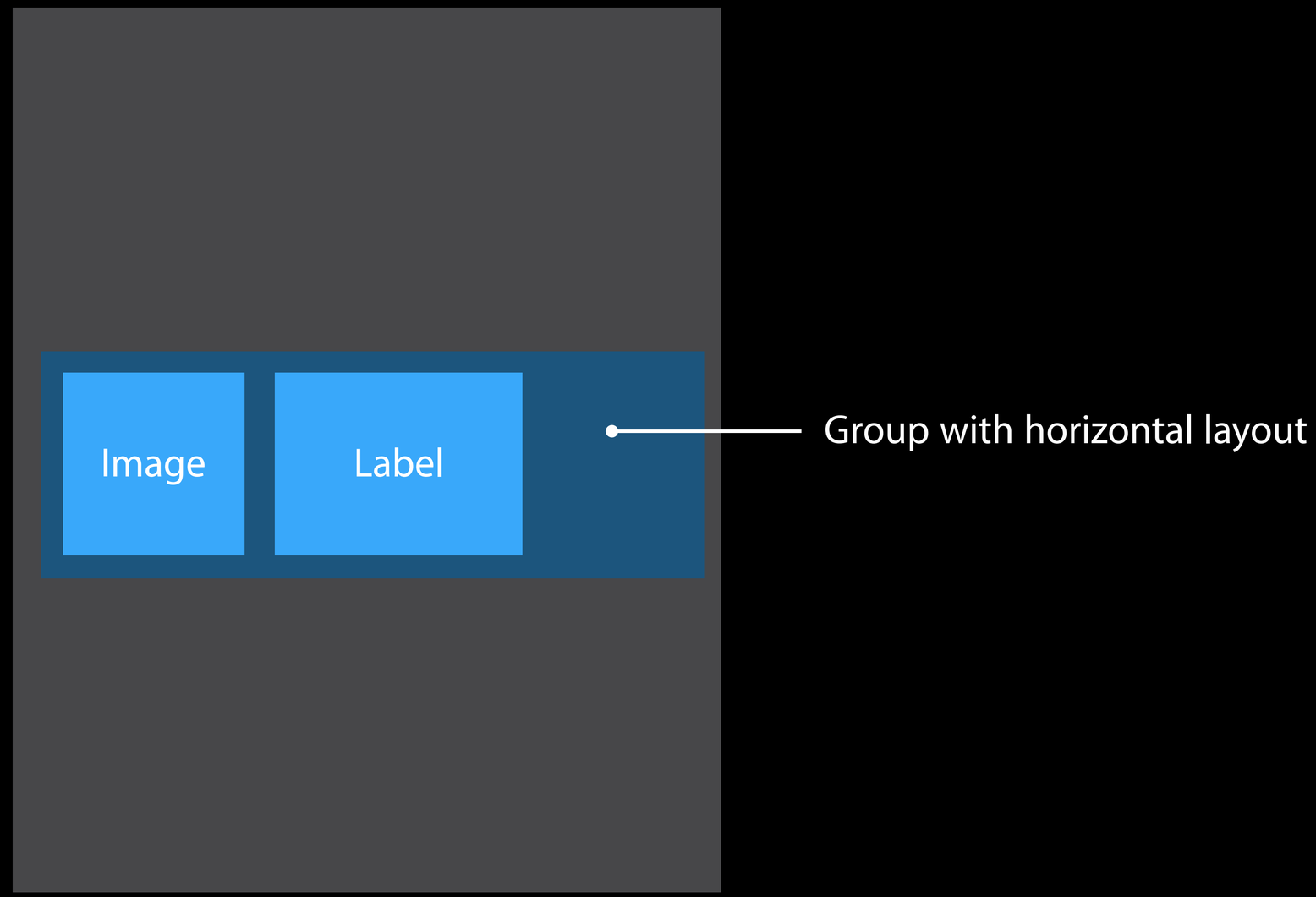
Table Rows

Table Rows

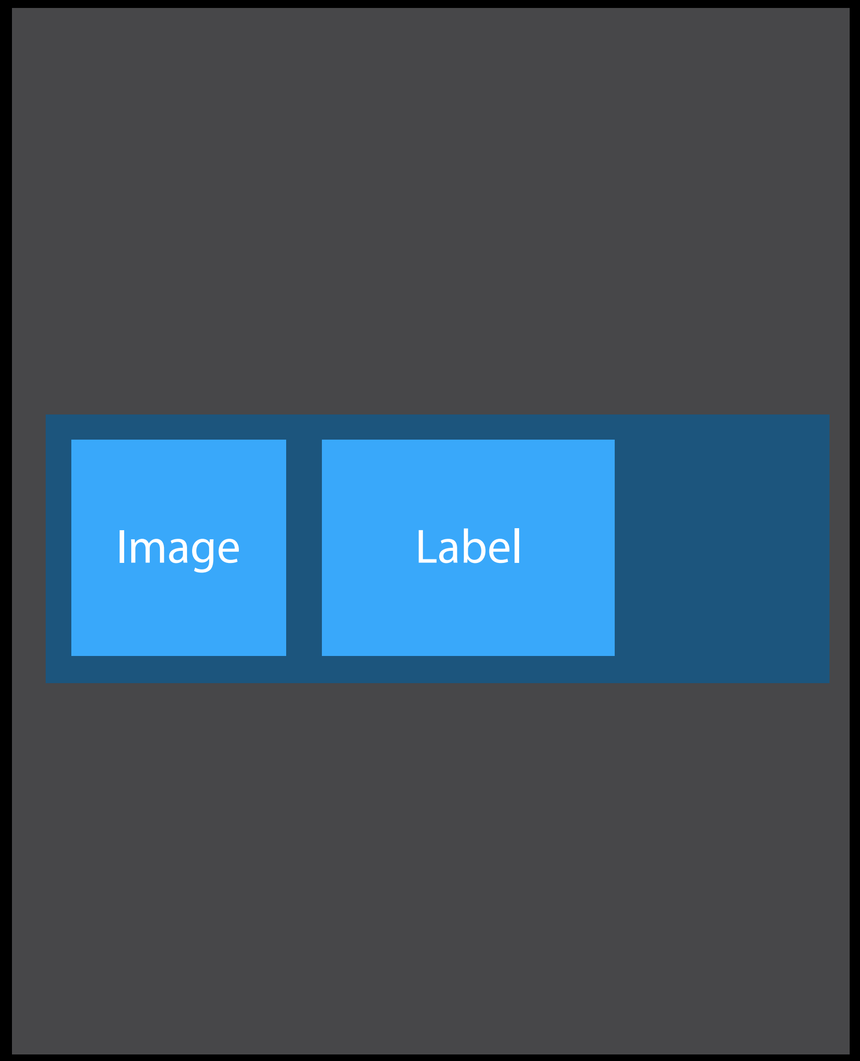
Layout for Table Row Controller



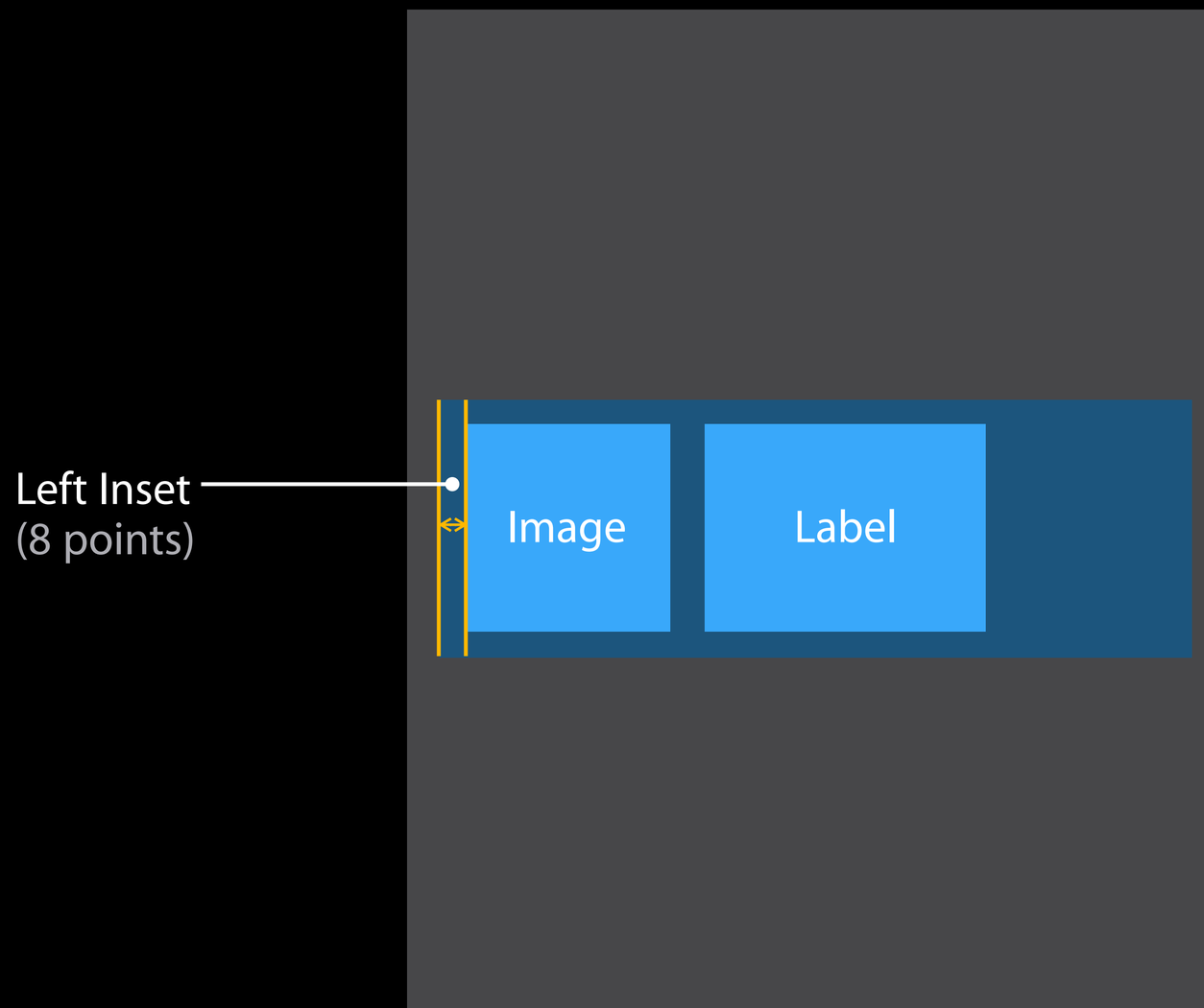
Layout for Table Row Controller



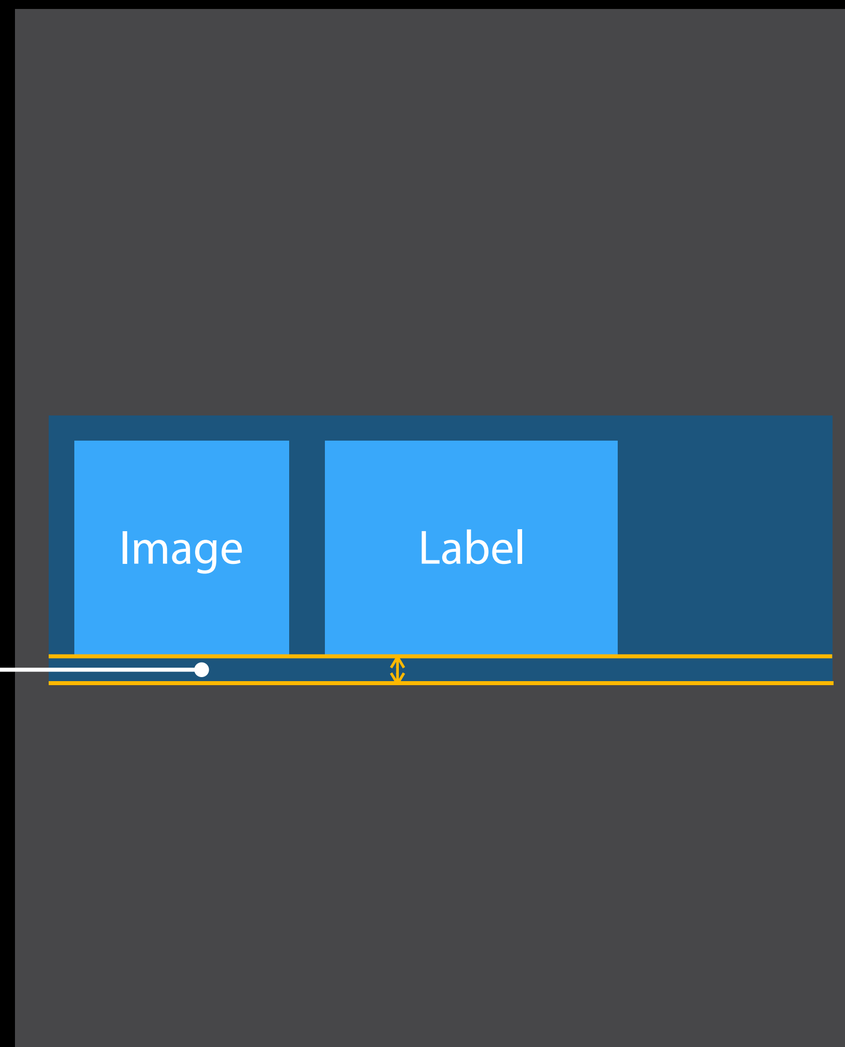
Insets and Spacing



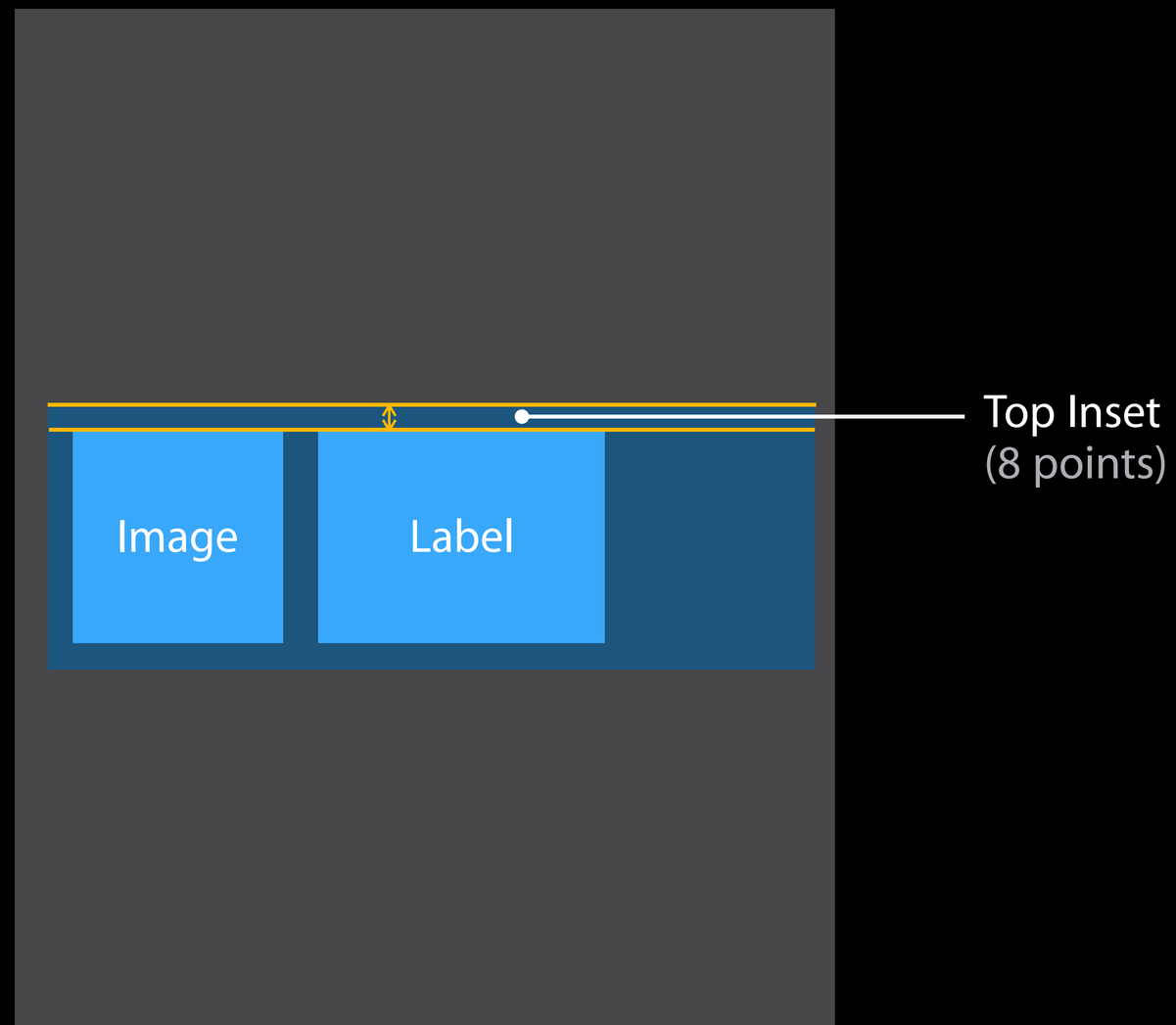
Insets and Spacing



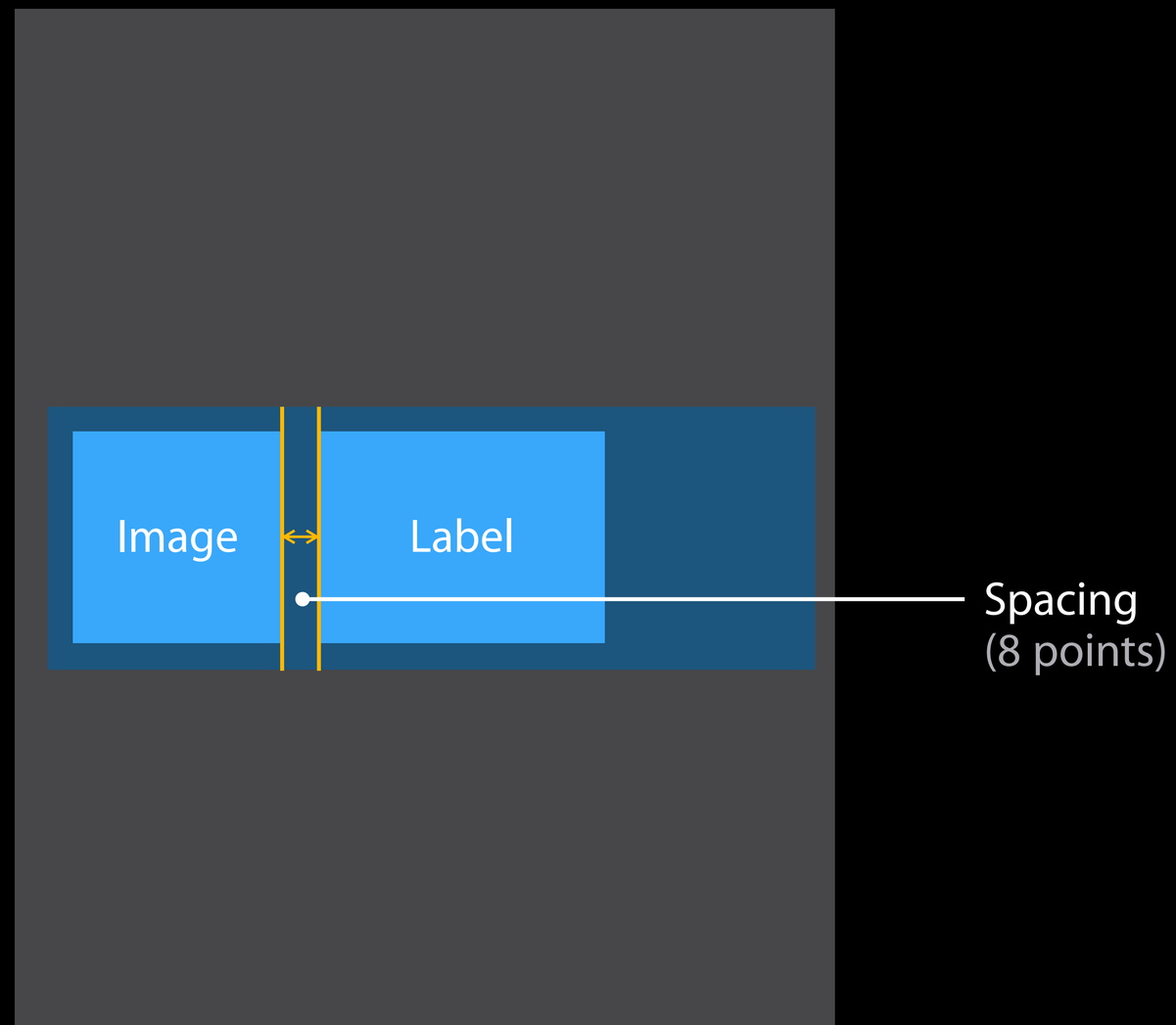
Insets and Spacing



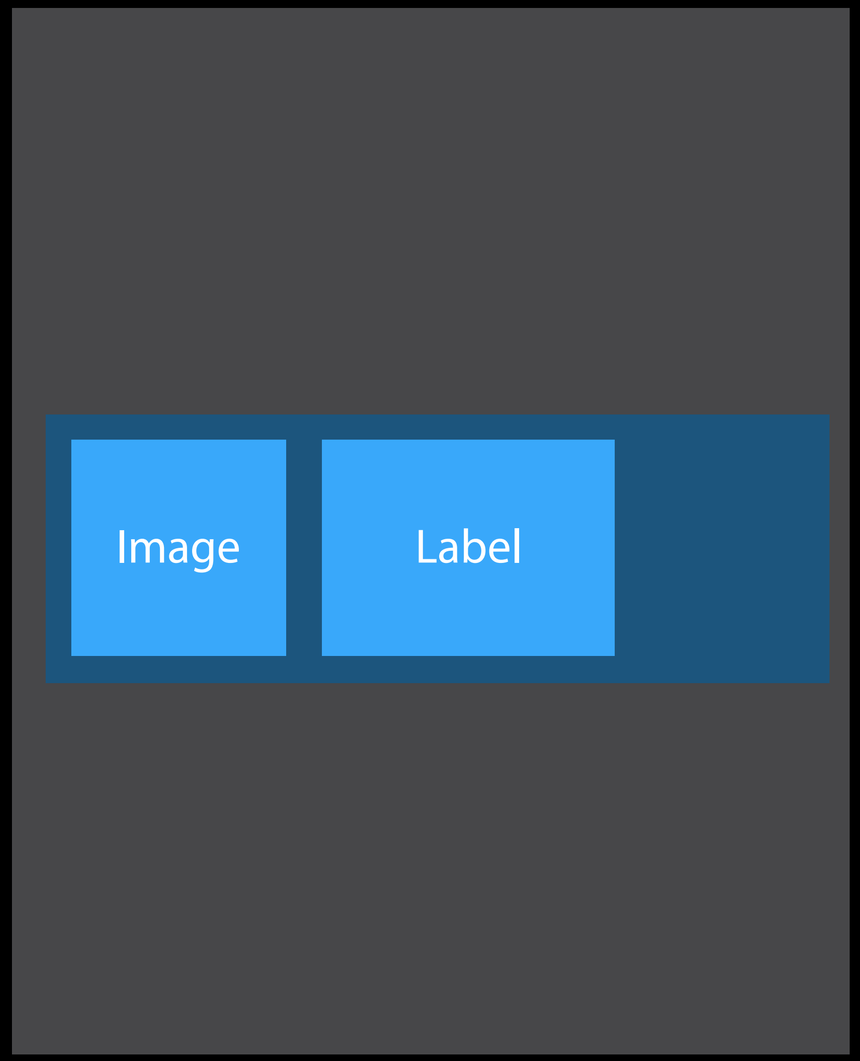
Insets and Spacing



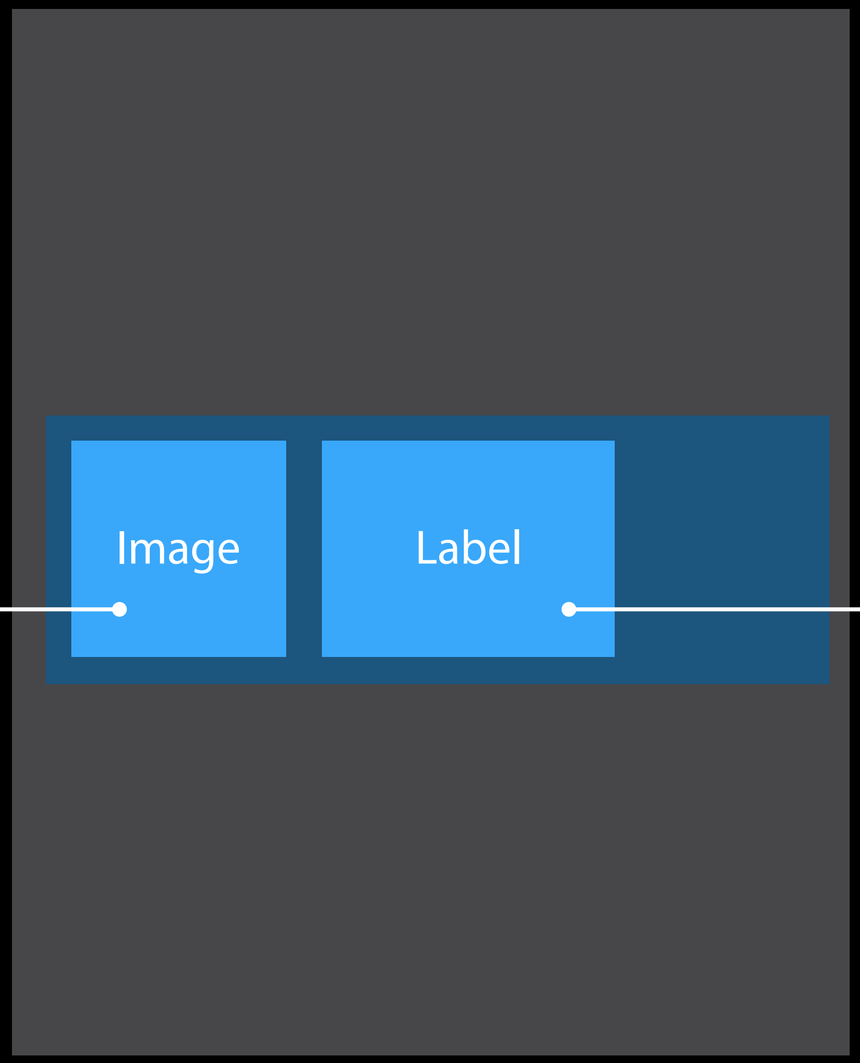
Insets and Spacing



Alignment



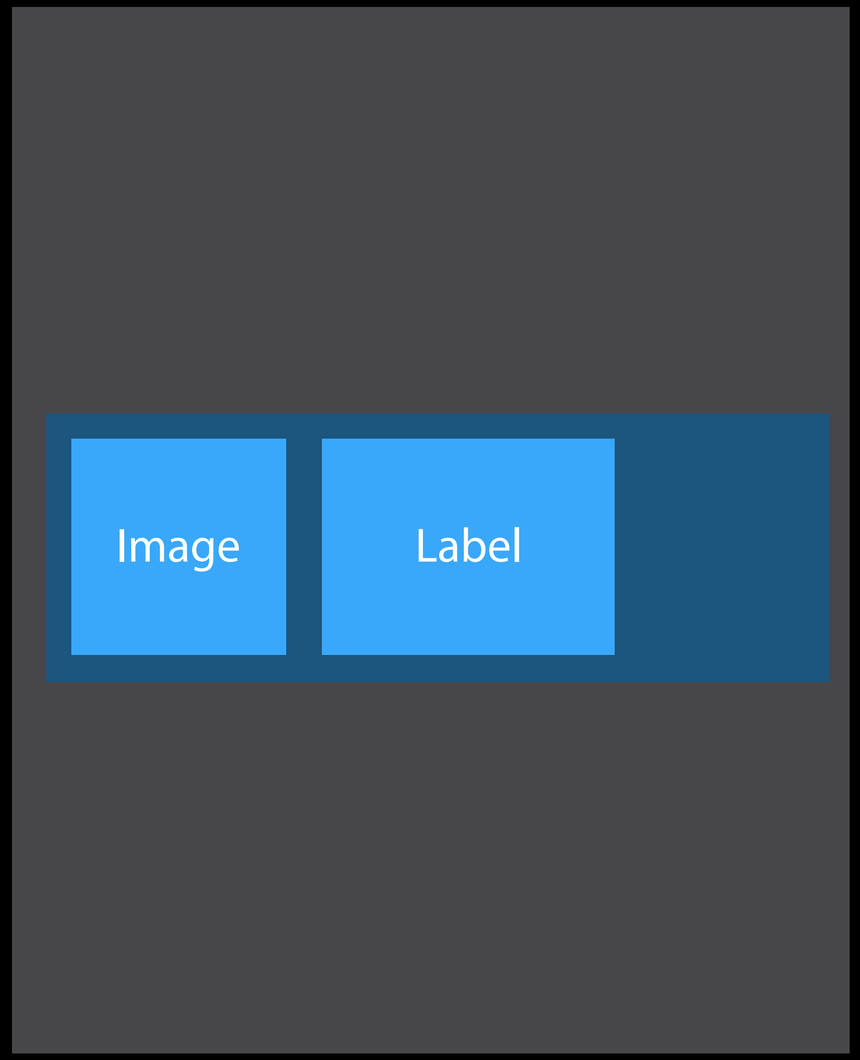
Alignment



Horizontal: Left
Vertical: Center

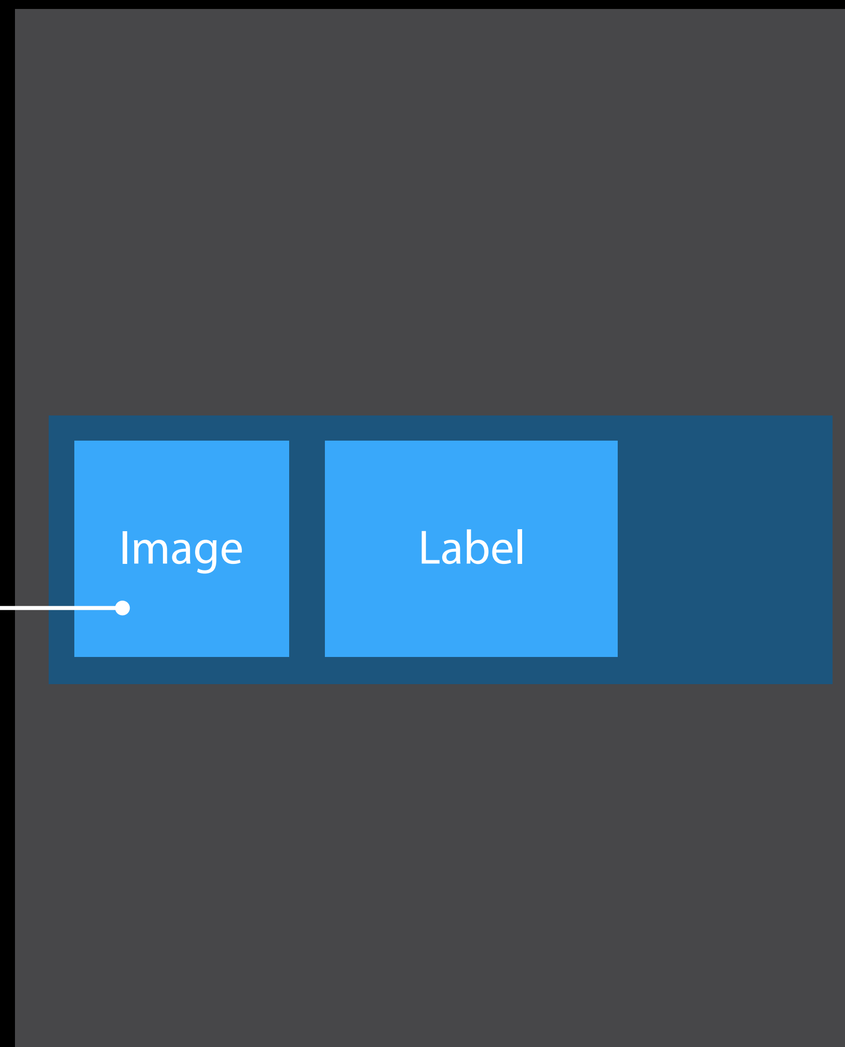
Horizontal: Left
Vertical: Center

Sizing

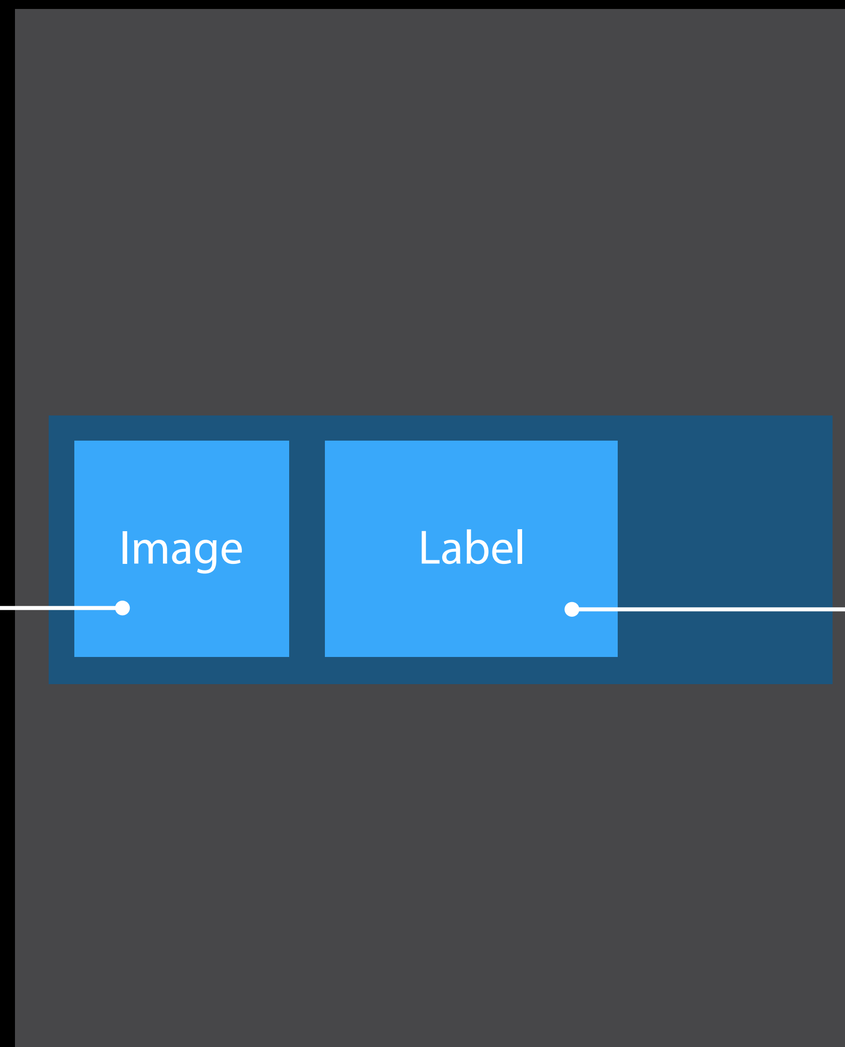


Sizing

Fixed Size
Width 30
Height 30



Sizing



Fixed Size
Width 30
Height 30

Size To Fit

Table Row Layout

Table Row Layout

Groups with horizontal layout

Table Row Layout

Groups with horizontal layout

Fine tune

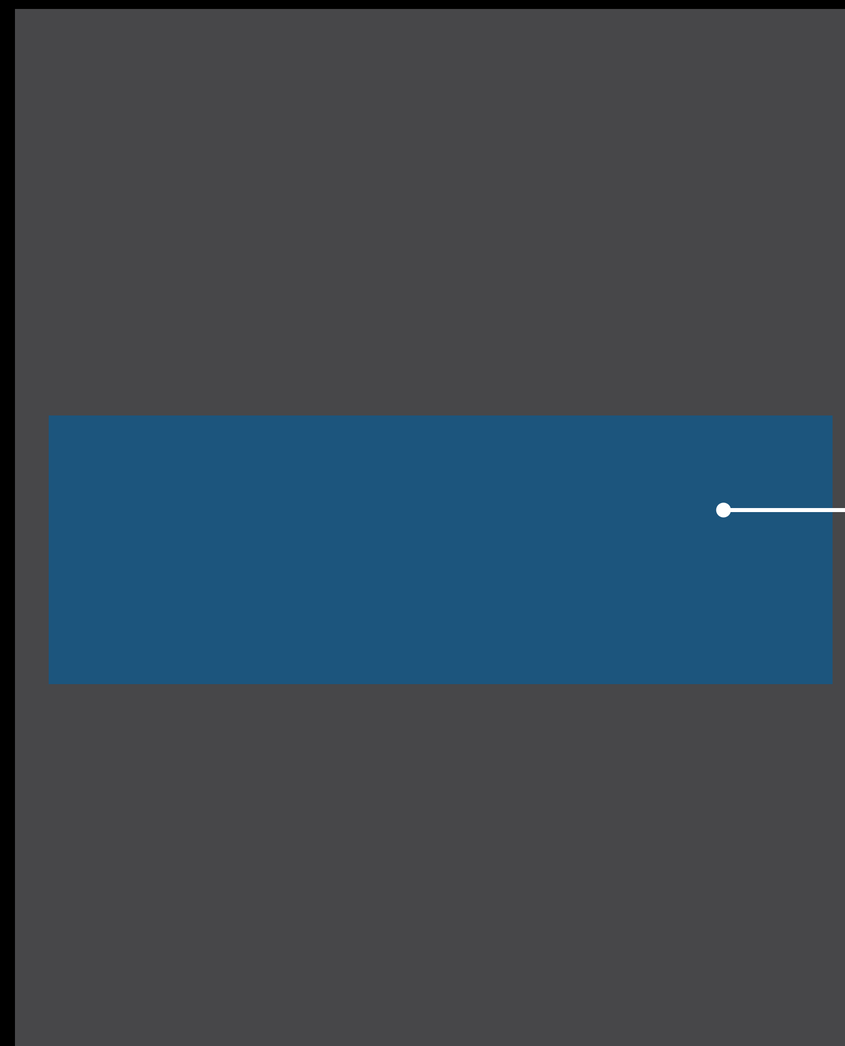
- Alignment
- Insets and spacing
- Sizing

Layout for Ingredients Controller

Layout for Ingredients Controller

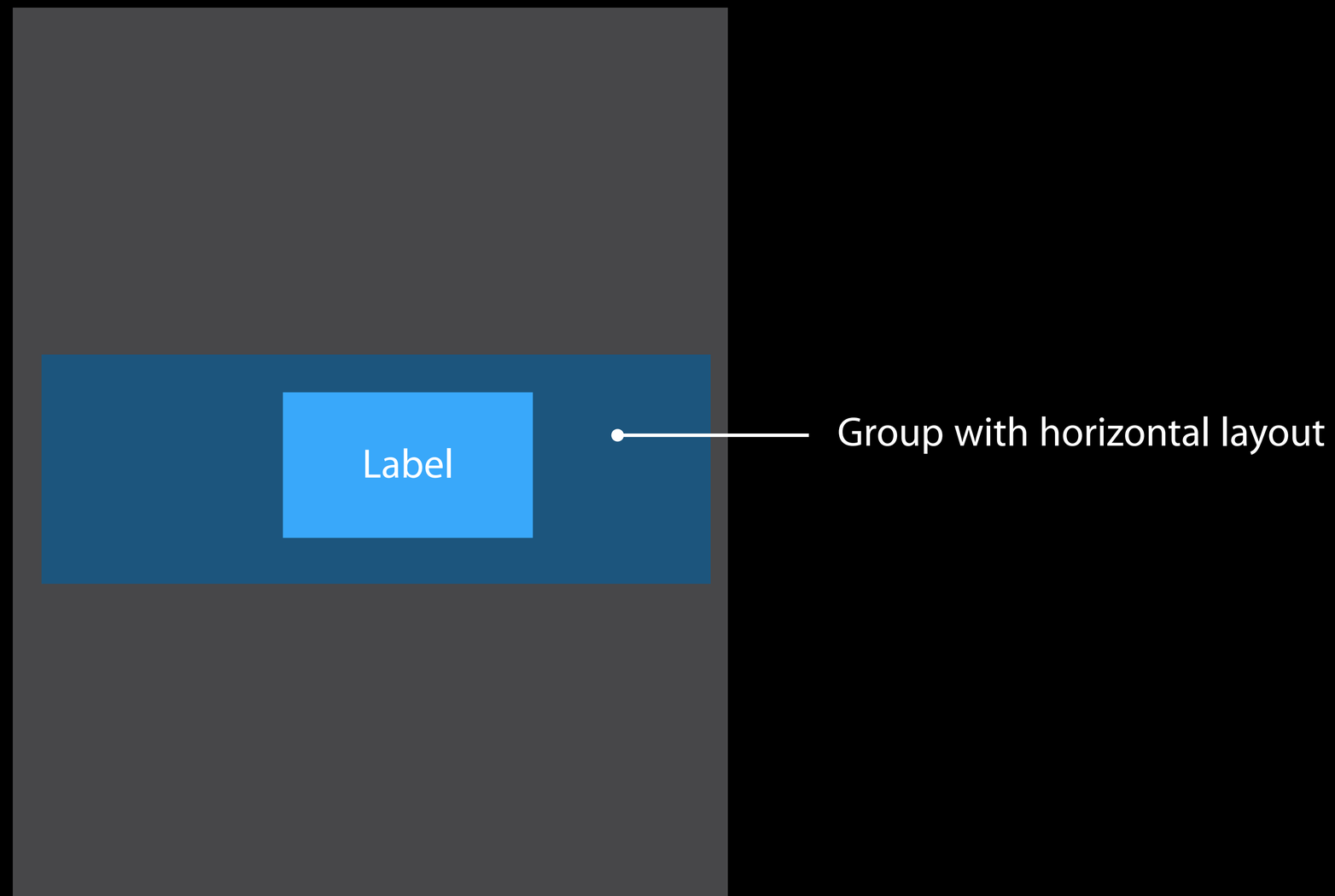
Layout for Ingredients Controller

Layout for Ingredients Controller

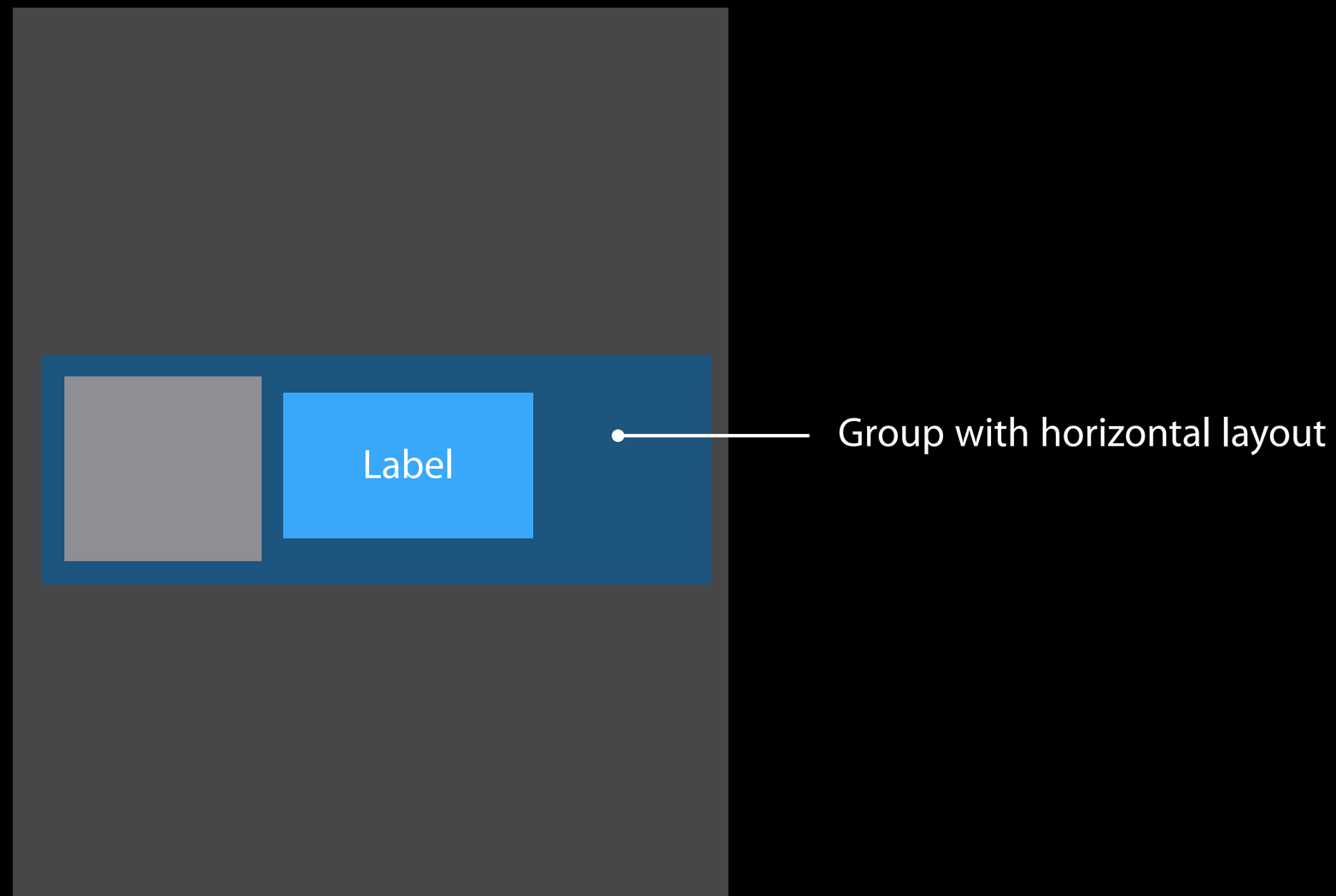


Group with horizontal layout

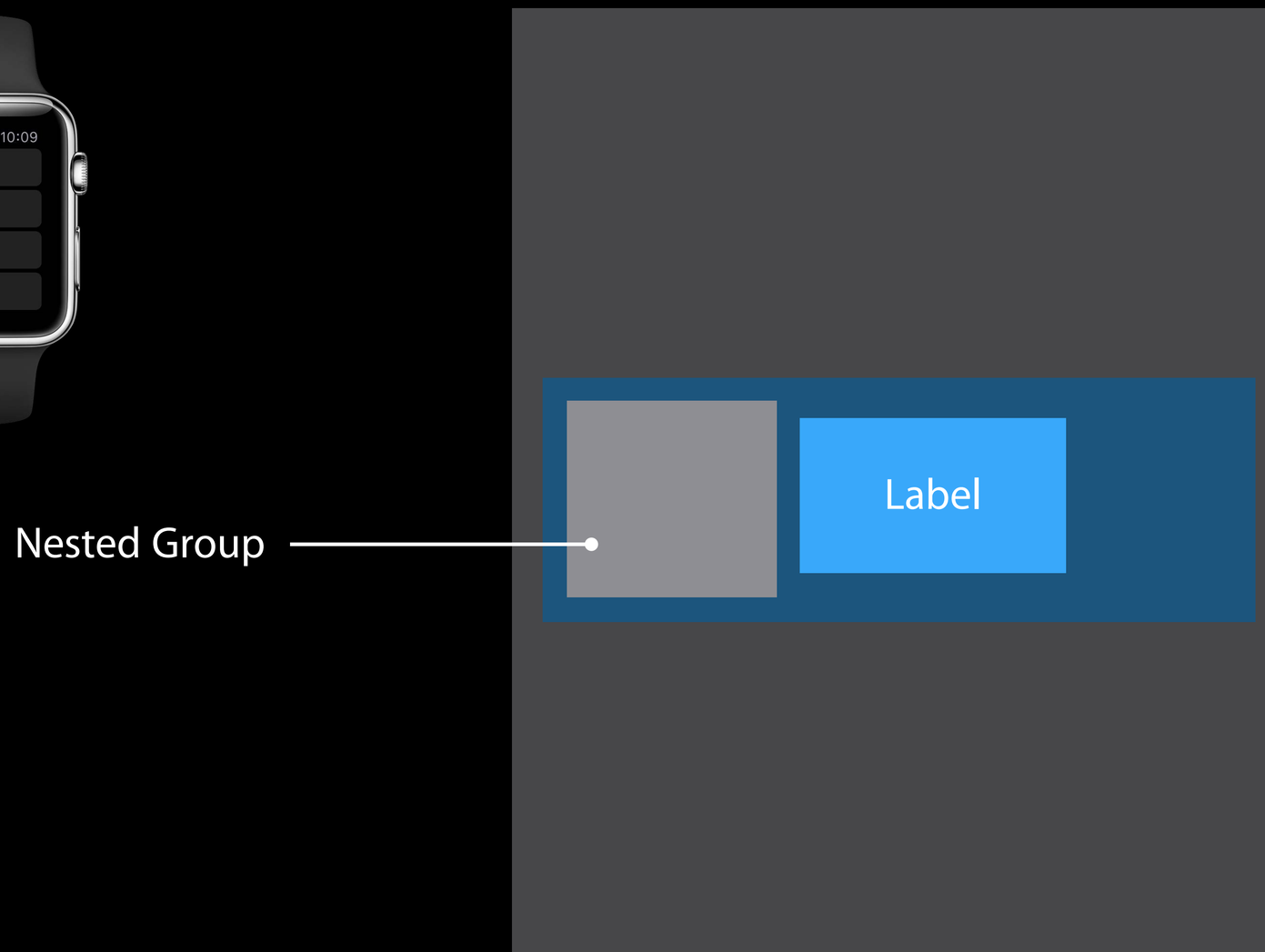
Layout for Ingredients Controller



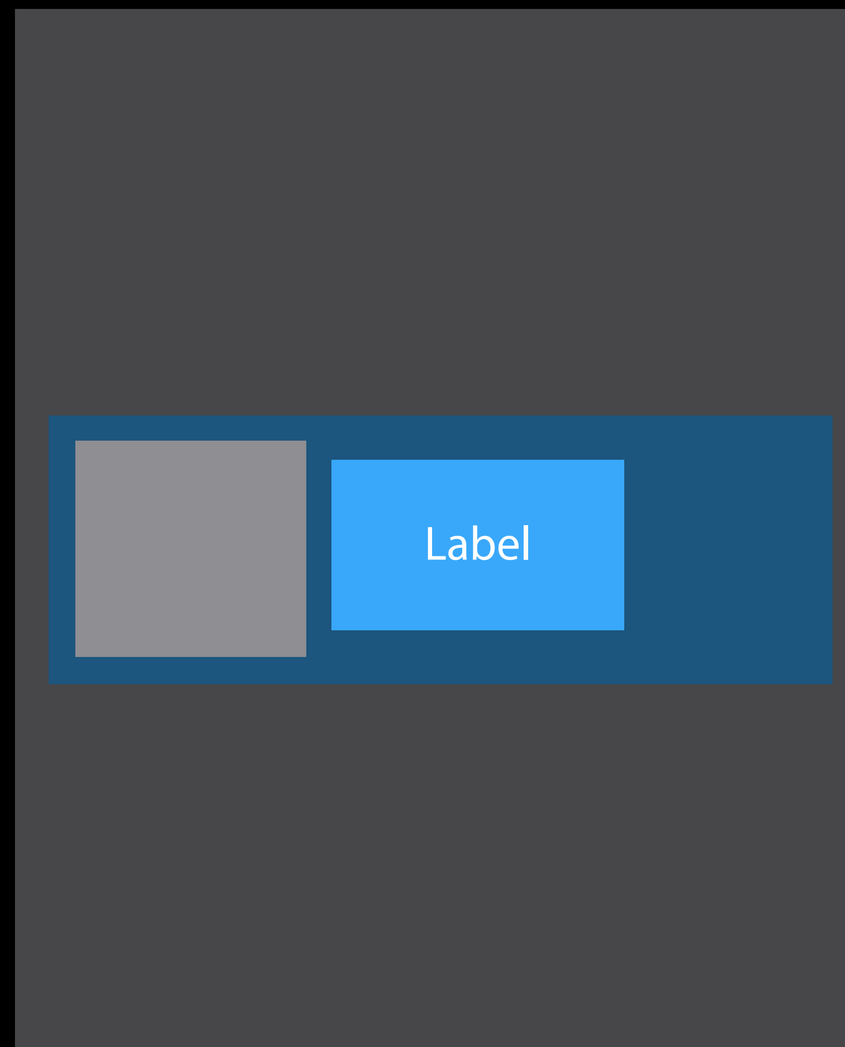
Layout for Ingredients Controller



Layout for Ingredients Controller

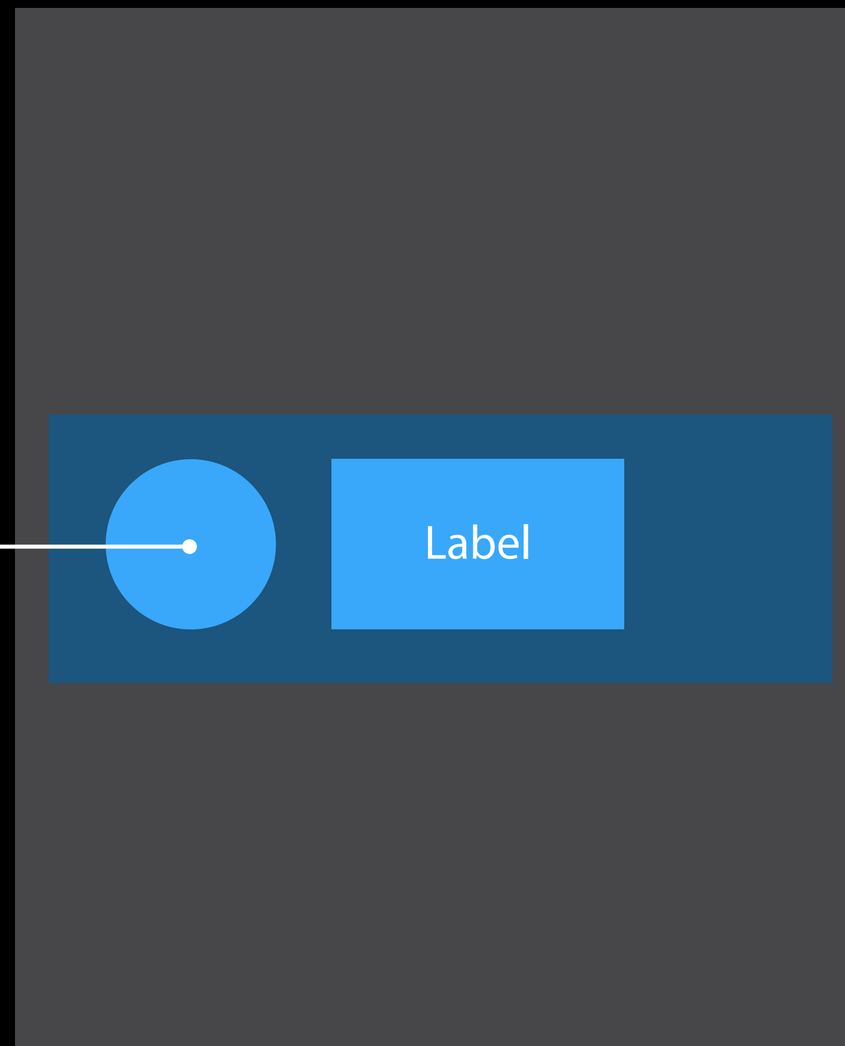


Groups Can Have Backgrounds



Groups Can Have Backgrounds

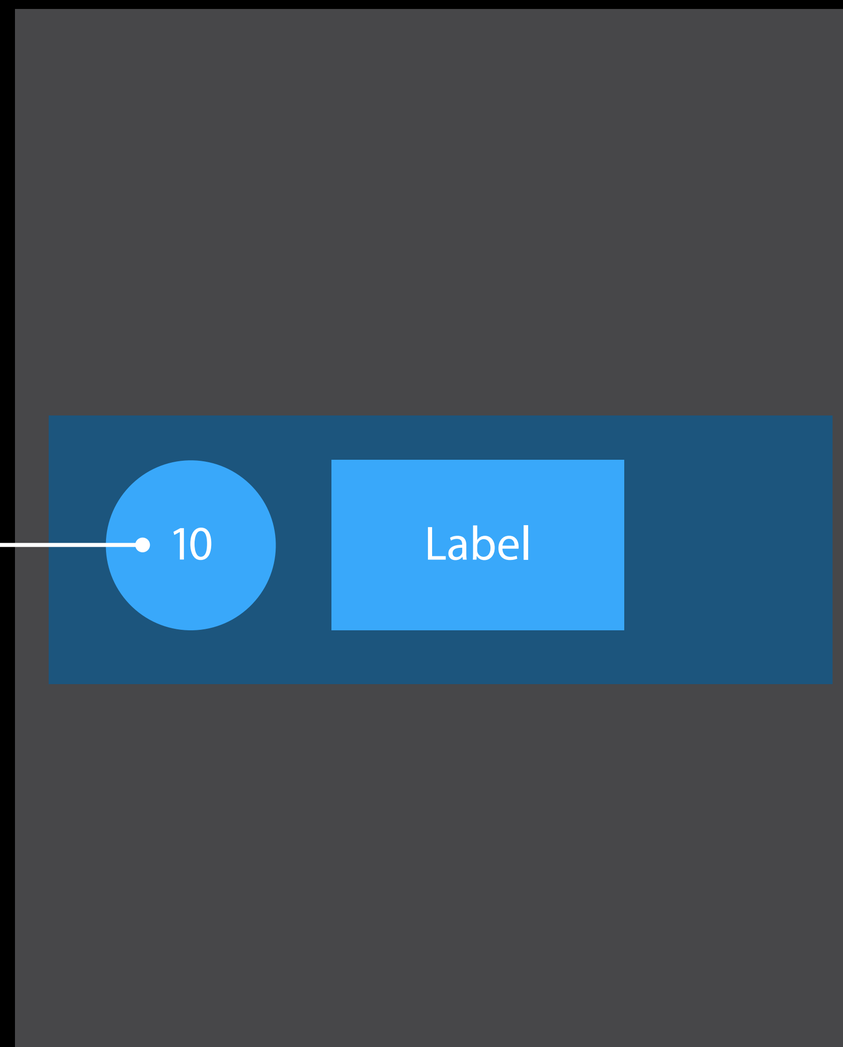
Group with
blue background
and radius of 8



Alignment of Number Label

Alignment of Number Label

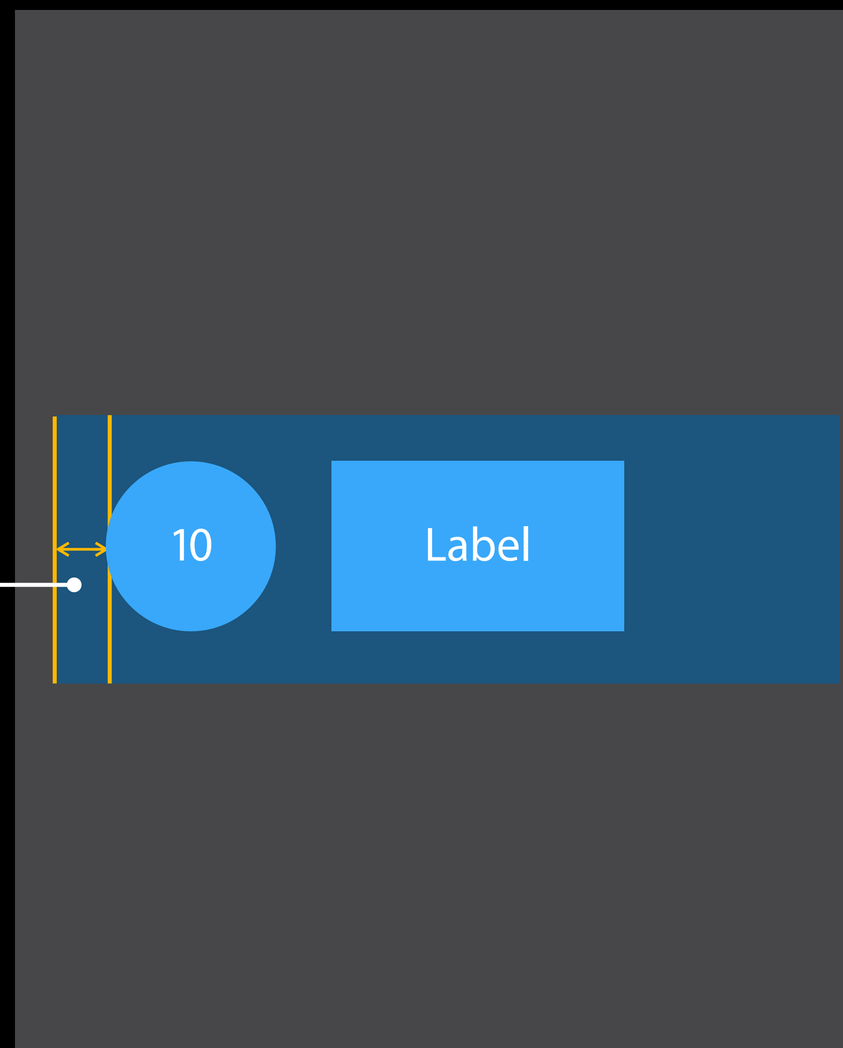
Centered in group



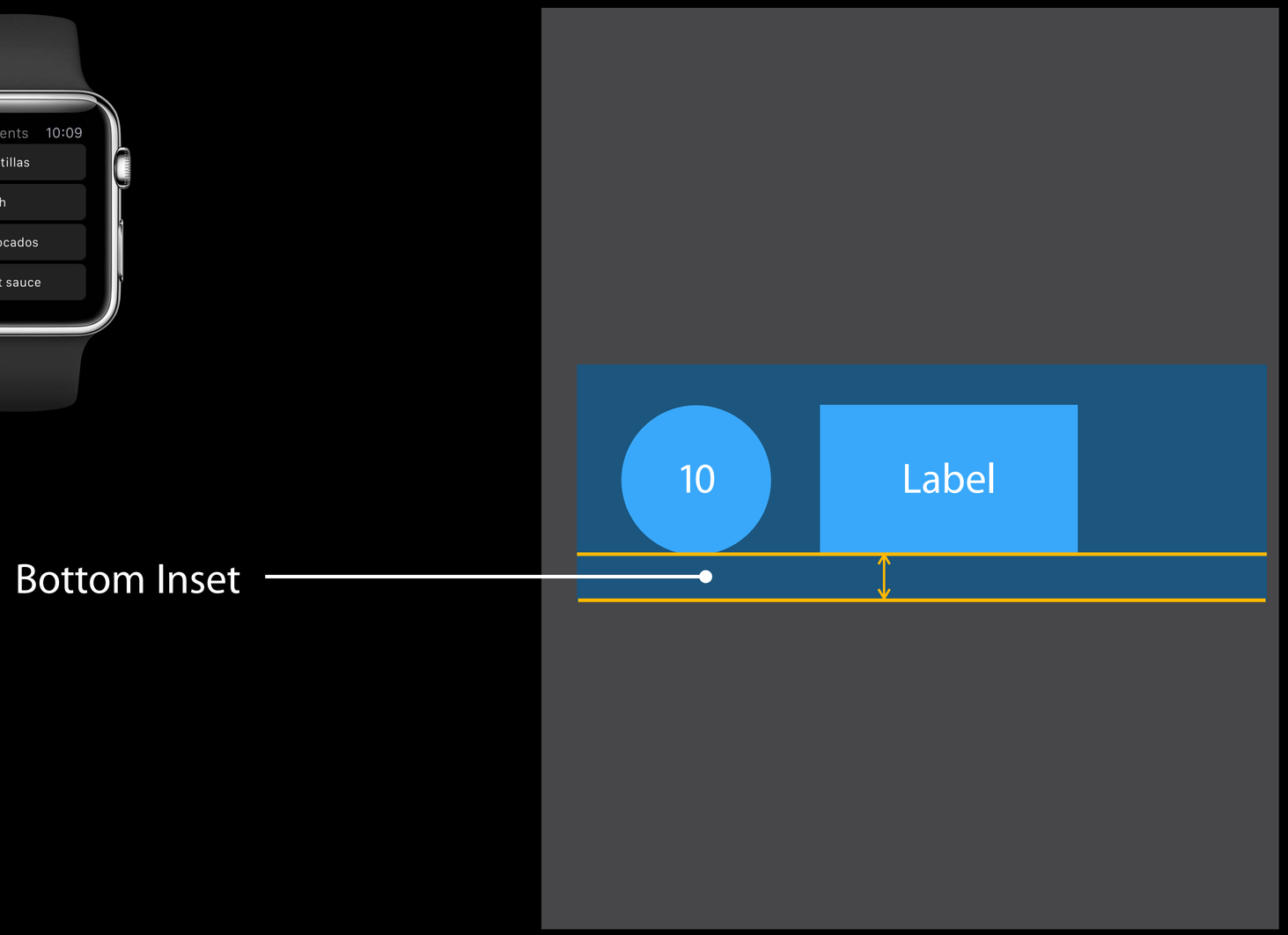
Insets and Spacing

Insets and Spacing

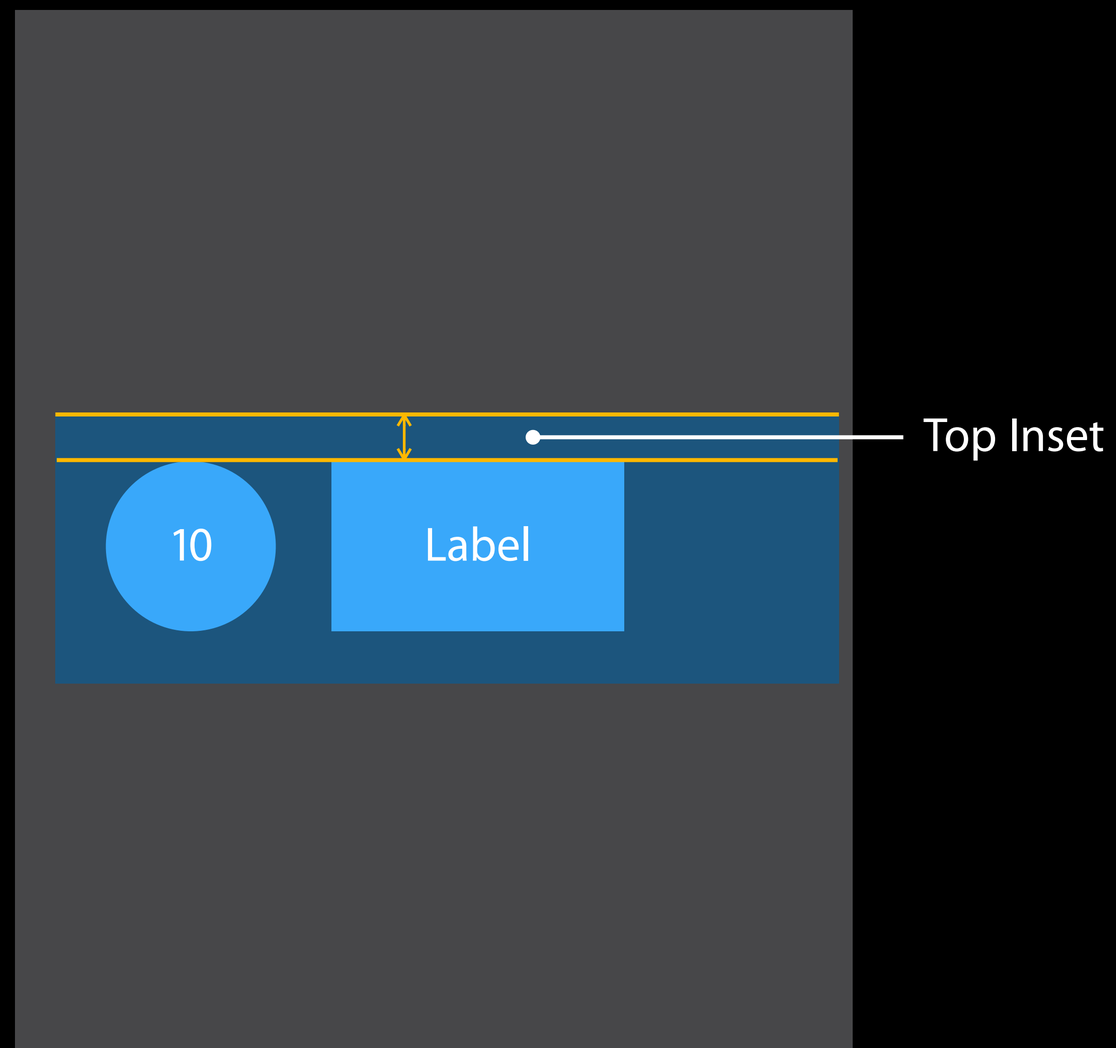
Left Inset



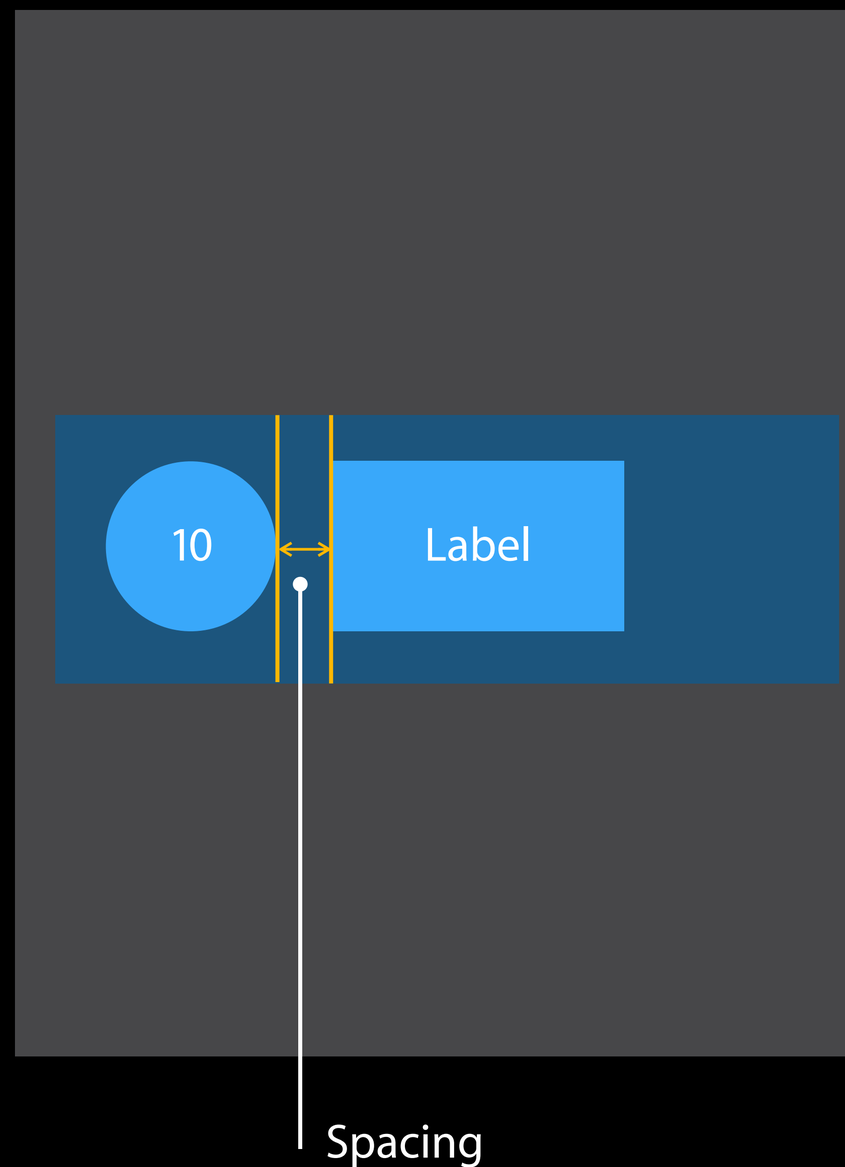
Insets and Spacing



Insets and Spacing



Insets and Spacing



Ingredients Controller Layout

Ingredients Controller Layout

Nested groups

Ingredients Controller Layout

Nested groups

Groups with background

Ingredients Controller Layout

Nested groups

Groups with background

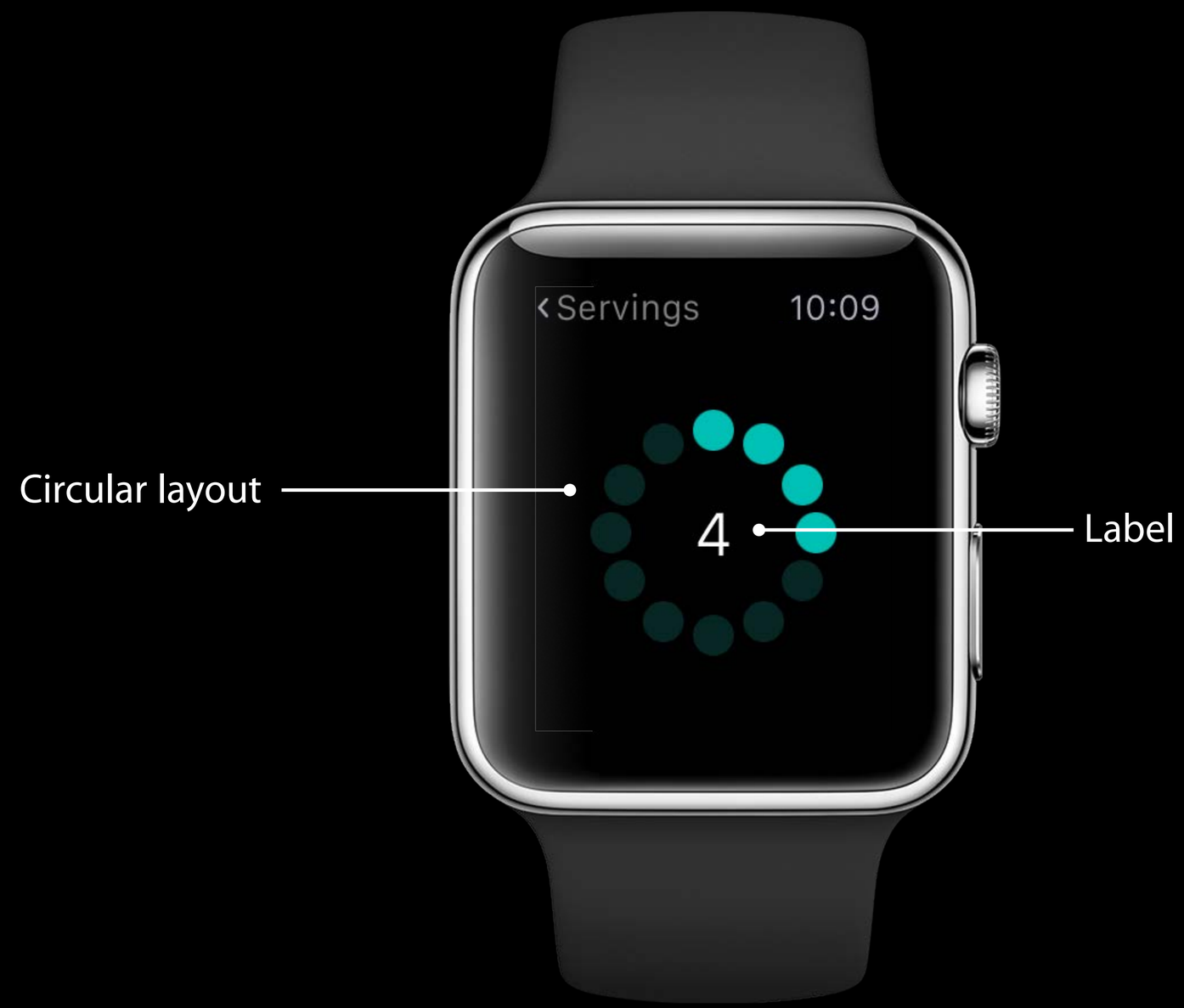
- Color
- Images

Layout for Servings Controller

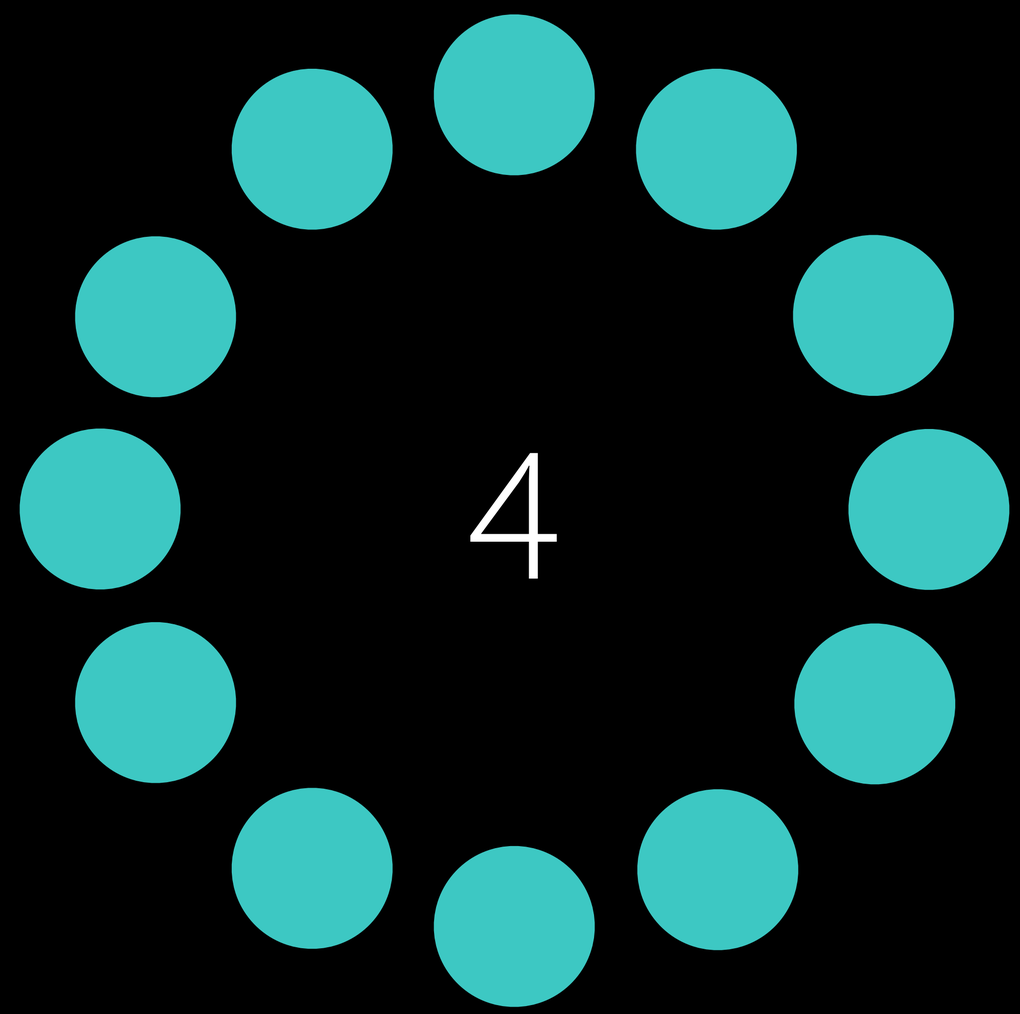
Layout for Servings Controller

Circular layout

Layout for Servings Controller

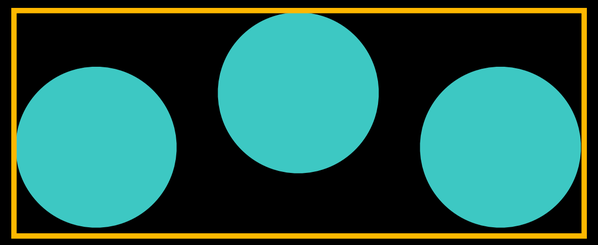


Layout for Servings Controller

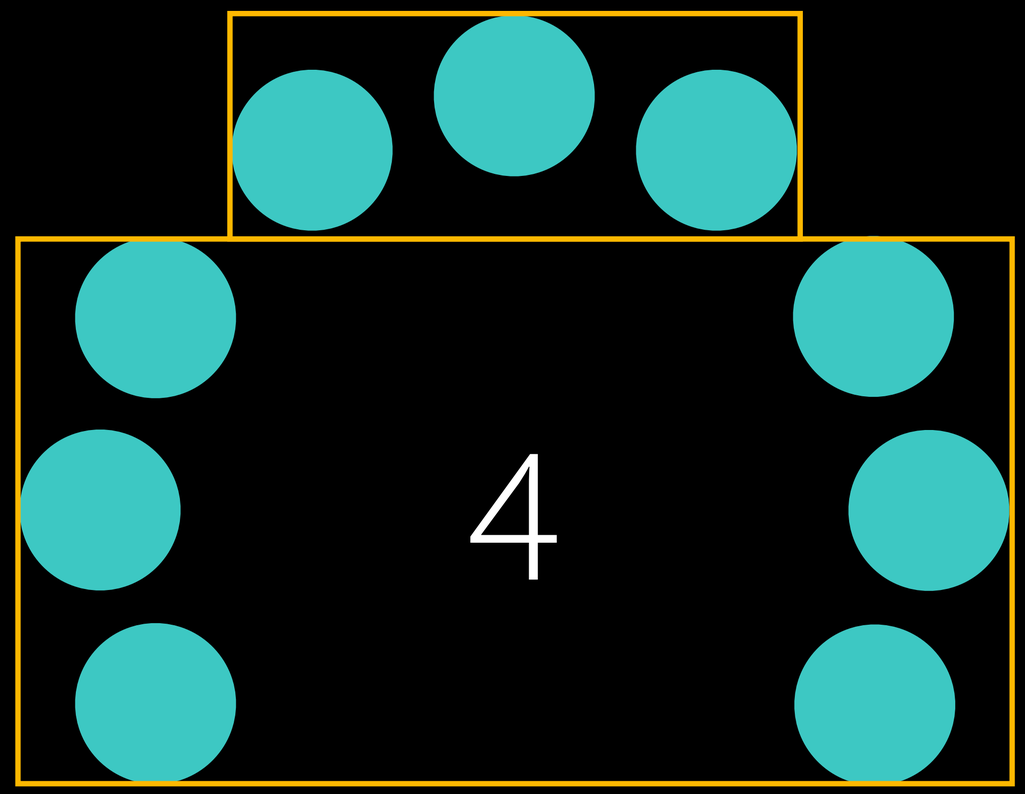


Three Top-Level Groups

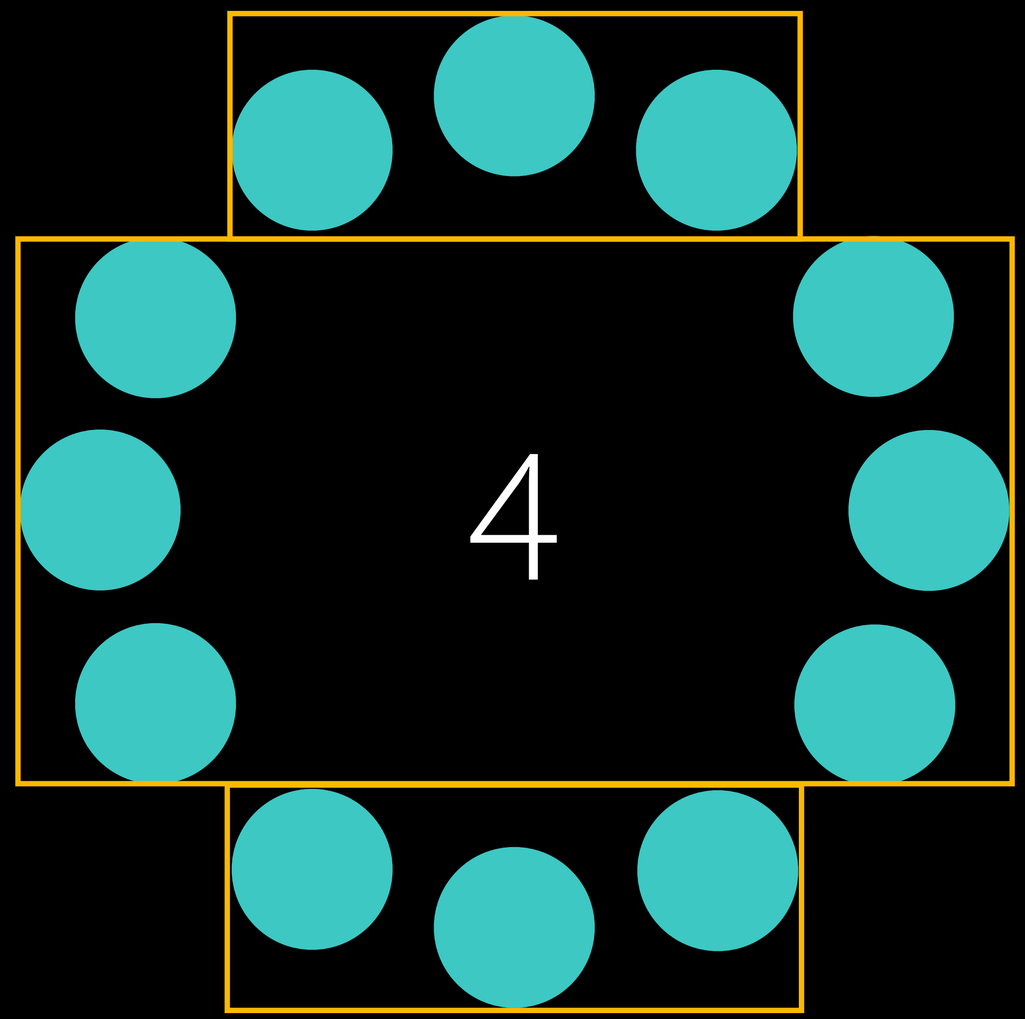
Three Top-Level Groups



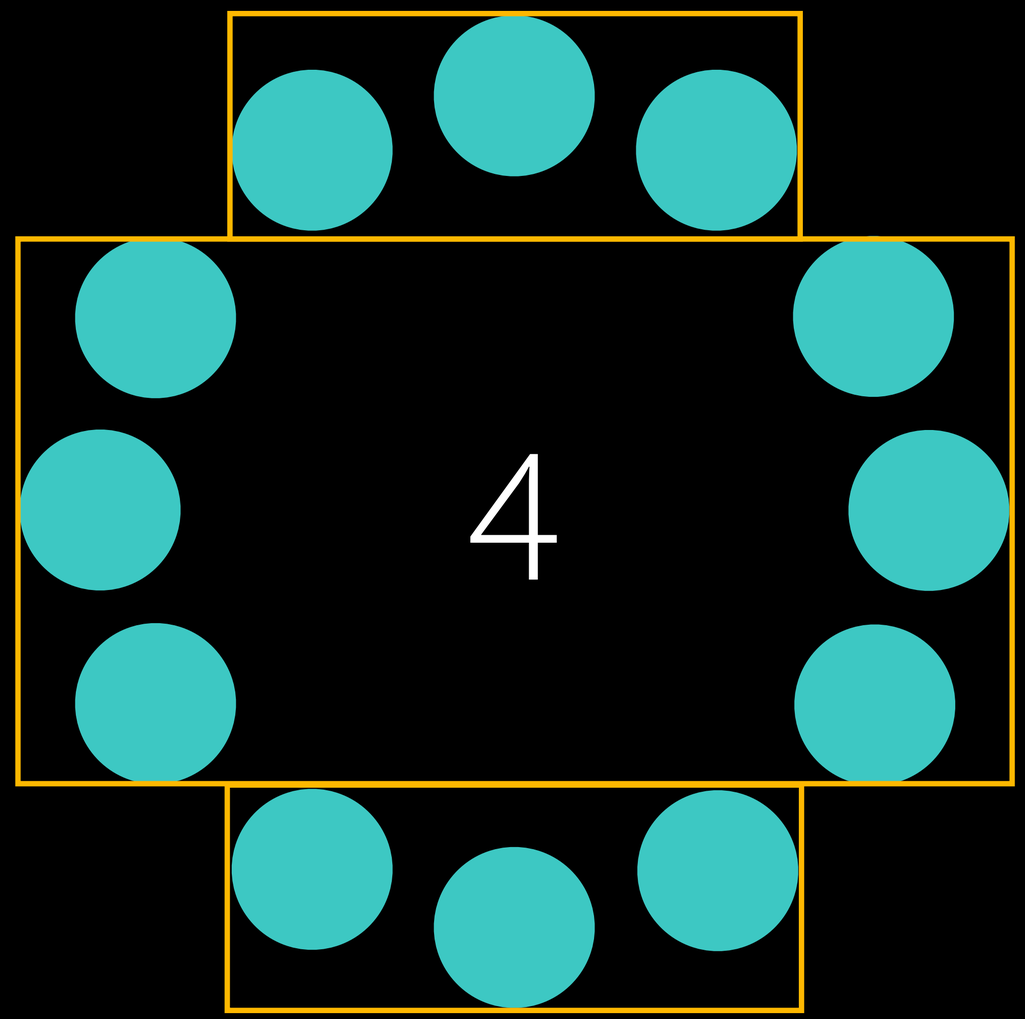
Three Top-Level Groups



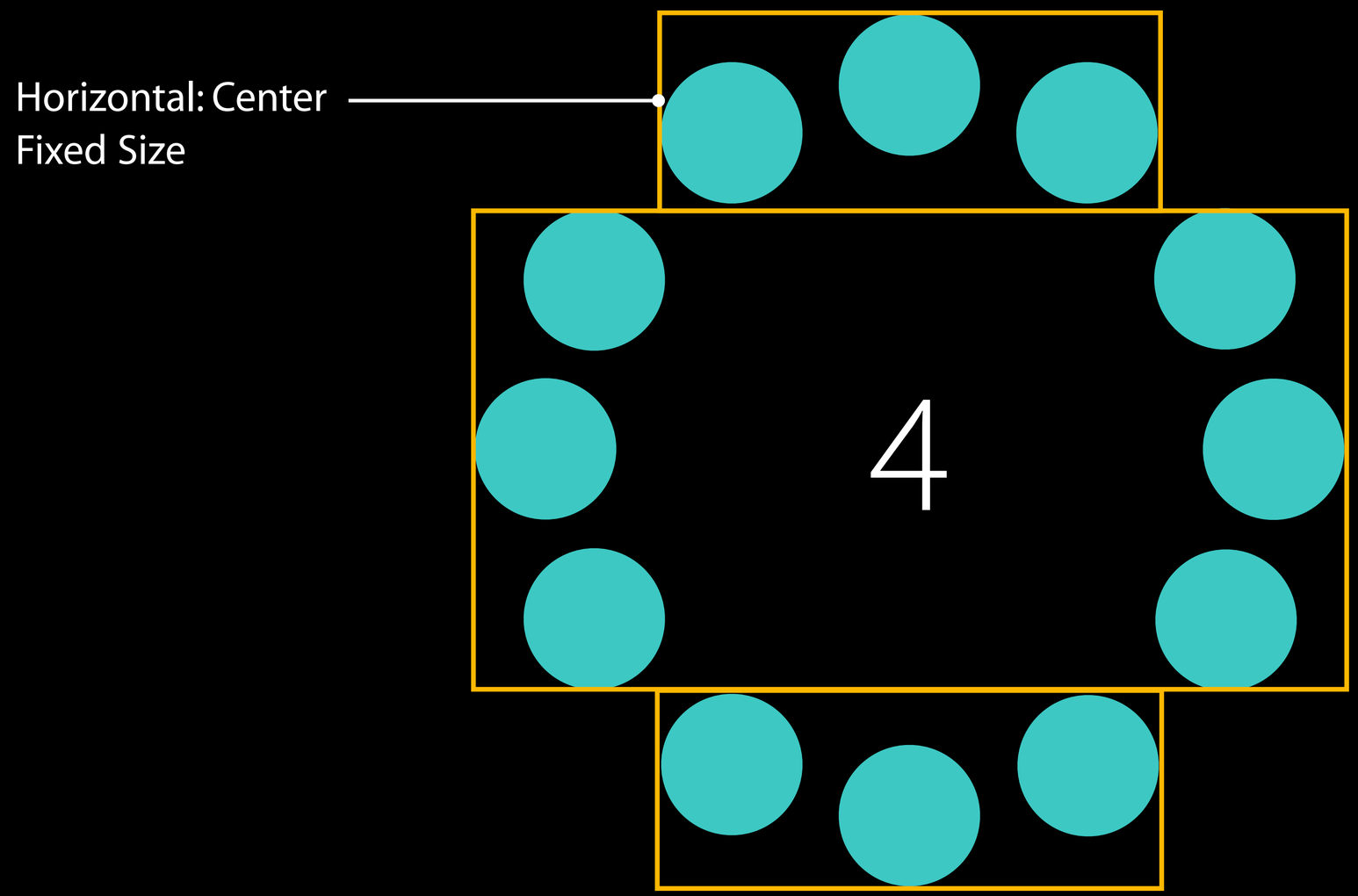
Three Top-Level Groups



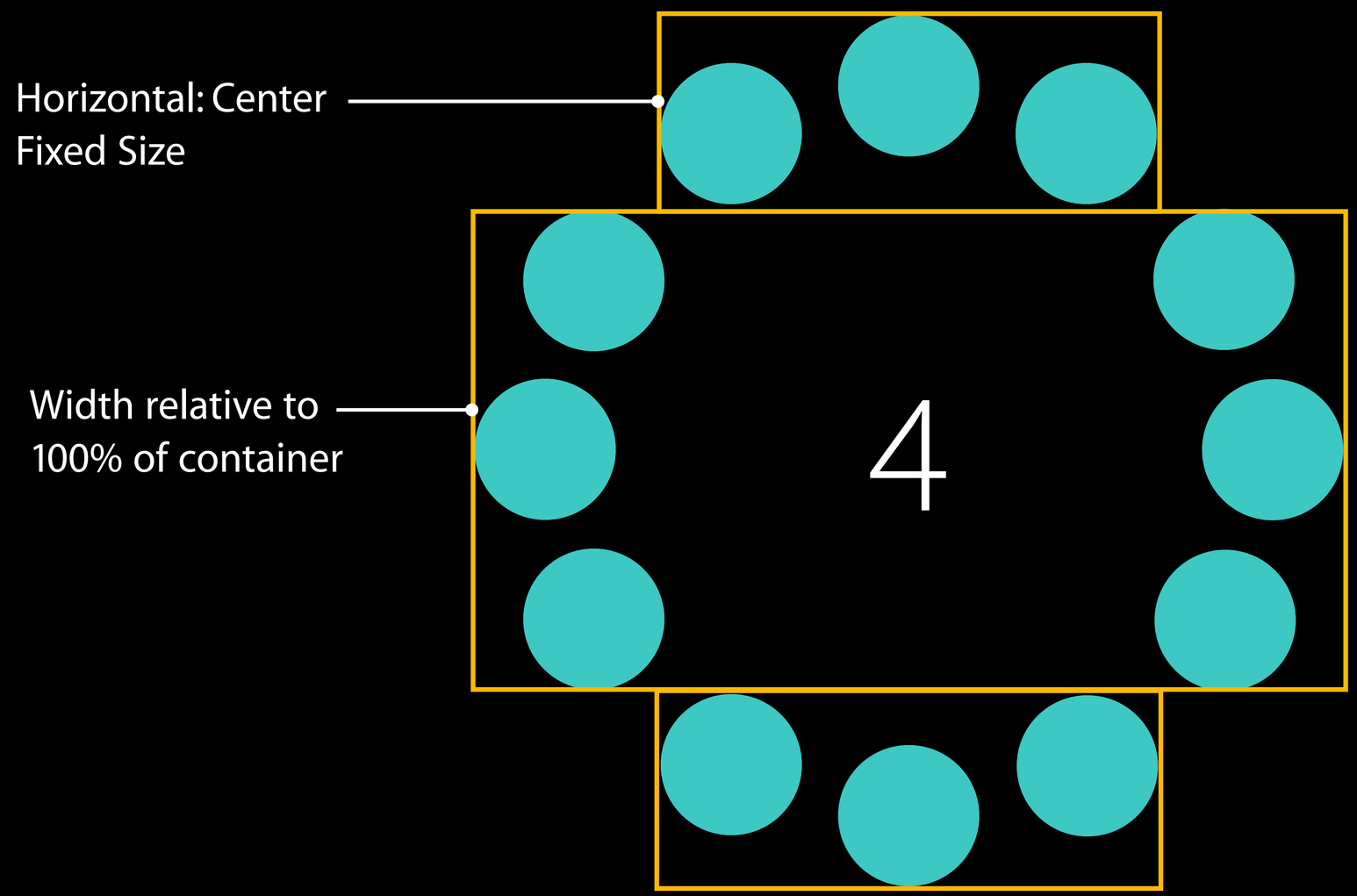
Group Alignment and Sizing



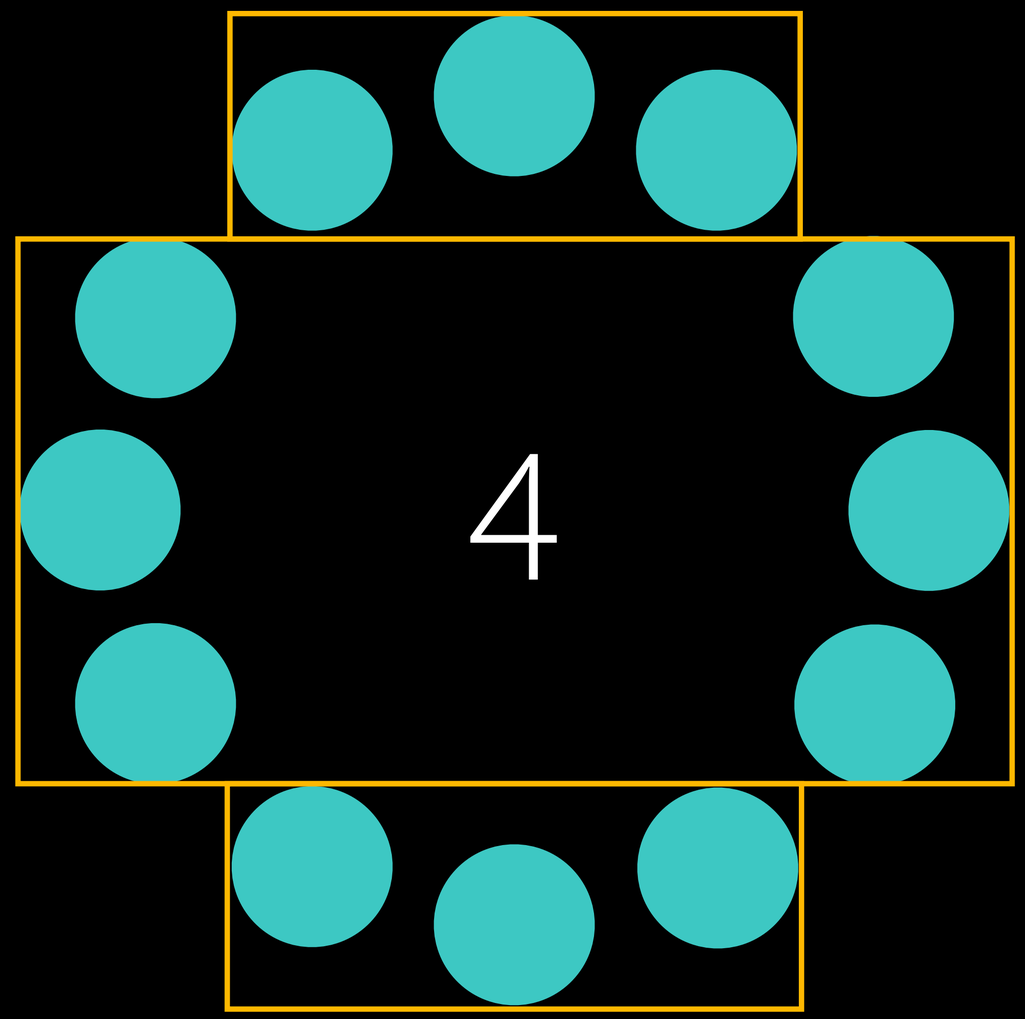
Group Alignment and Sizing



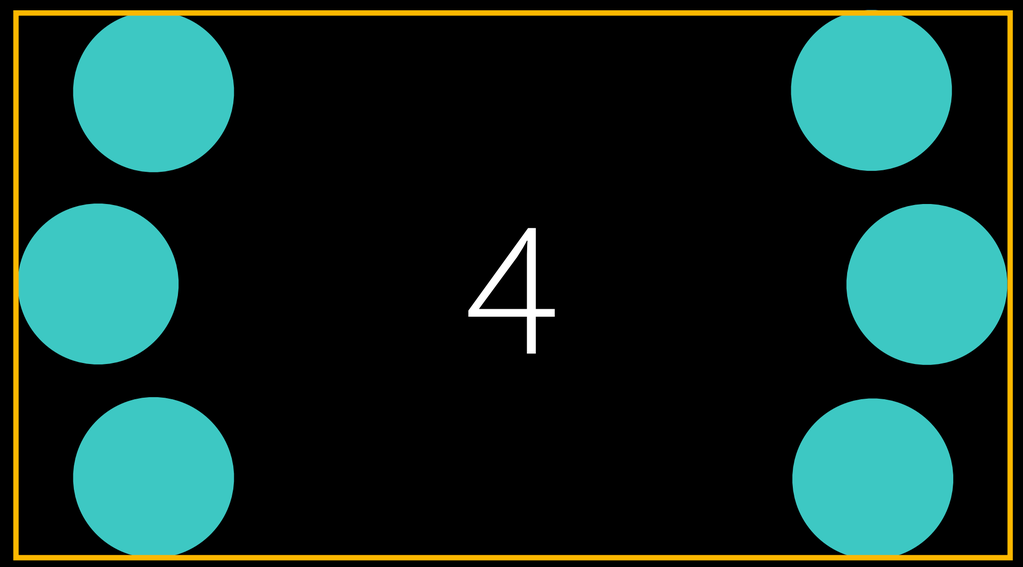
Group Alignment and Sizing



Second Group Has Nested Groups

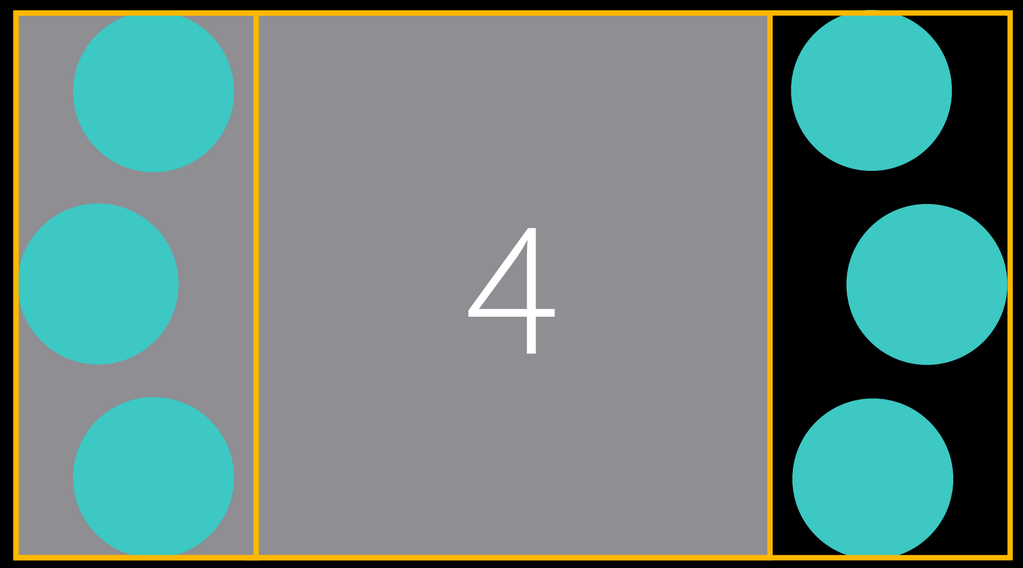


Second Group Has Nested Groups

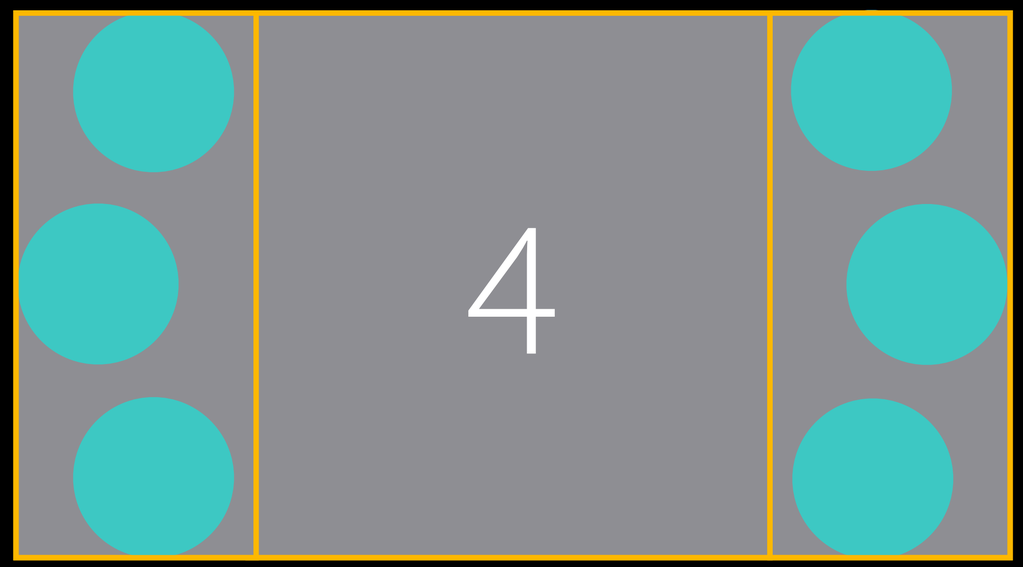


Second Group Has Nested Groups

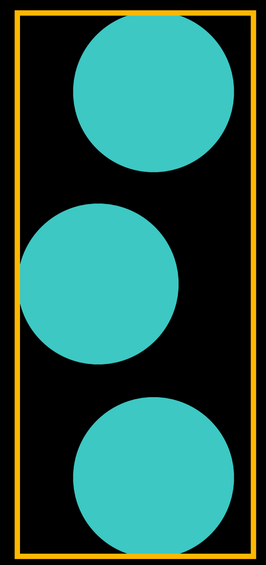
Second Group Has Nested Groups



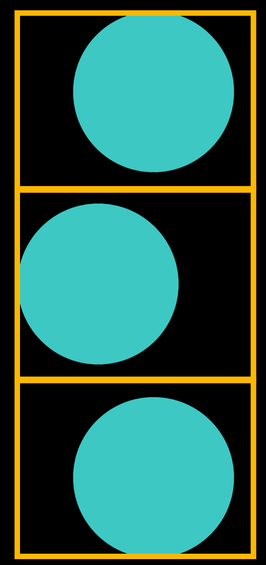
Second Group Has Nested Groups



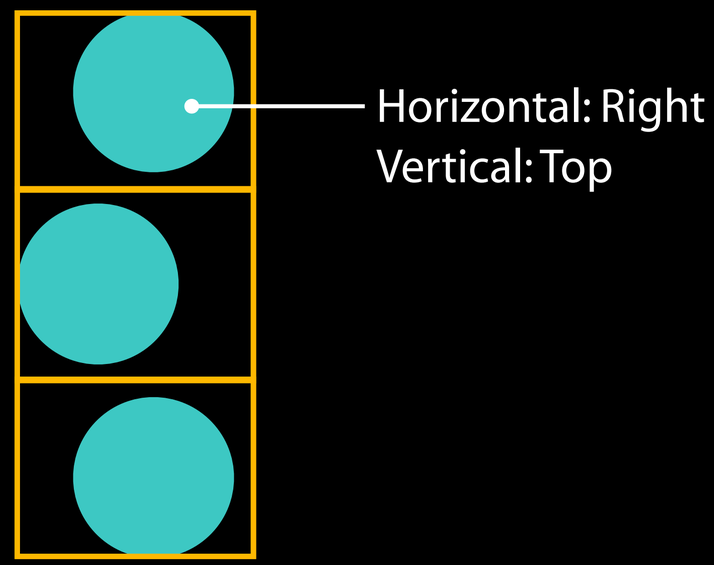
Alignment Inside Groups



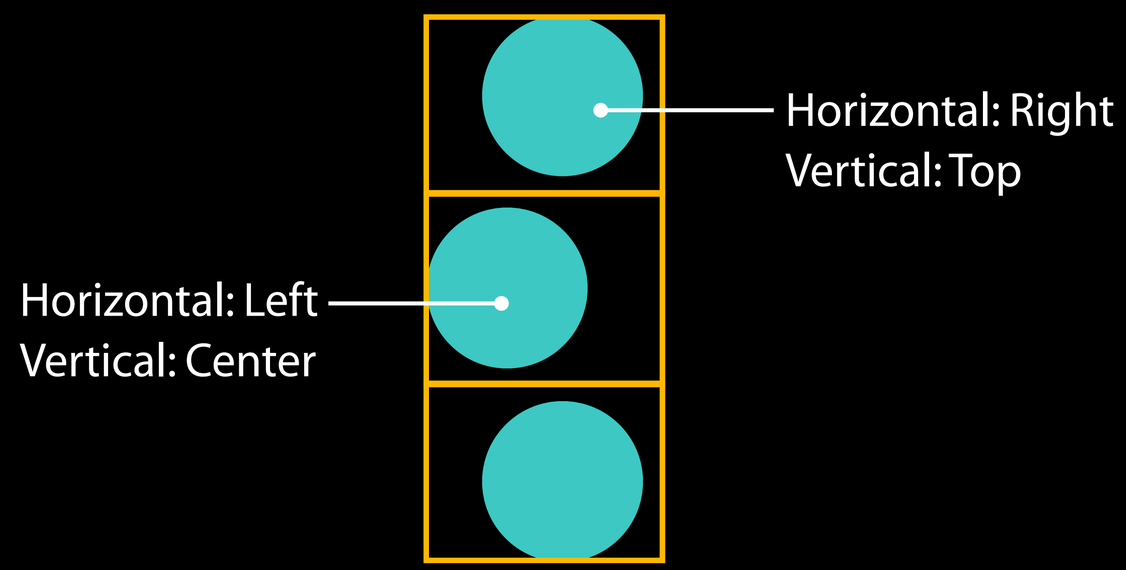
Alignment Inside Groups



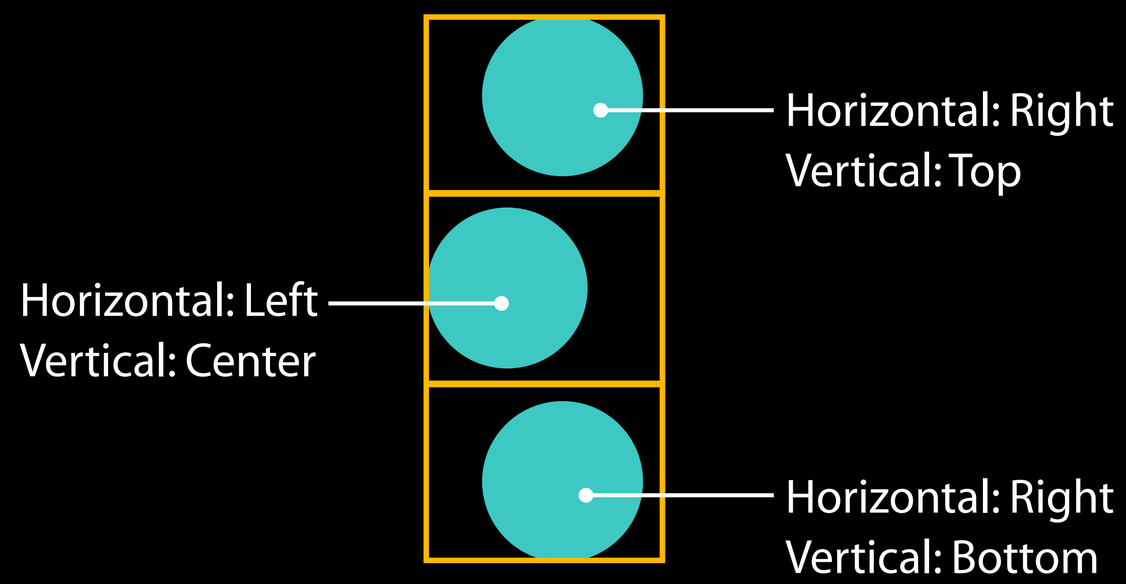
Alignment Inside Groups



Alignment Inside Groups



Alignment Inside Groups



Ingredients Controller Layout

Ingredients Controller Layout

Group nesting for complex layouts

Ingredients Controller Layout

Group nesting for complex layouts

Don’t abuse power of groups

Implications of Complex Layouts

Implications of Complex Layouts

WatchKit has no APIs for direct element creation

Implications of Complex Layouts

WatchKit has no APIs for direct element creation

What you describe in IB is created

Implications of Complex Layouts

WatchKit has no APIs for direct element creation

What you describe in IB is created

Hidden objects

Implications of Complex Layouts

WatchKit has no APIs for direct element creation

What you describe in IB is created

Hidden objects

- Creation cost

Implications of Complex Layouts

WatchKit has no APIs for direct element creation

What you describe in IB is created

Hidden objects

- Creation cost
- Save on layout cost

Using Images in Layouts

Using Images in Layouts

Don’t ignore transfer costs

Using Images in Layouts

Don’t ignore transfer costs

- watchOS 1 apps run extension on iPhone

Using Images in Layouts

Don’t ignore transfer costs

- watchOS 1 apps run extension on iPhone
- watchOS 2 apps need to install extension

Using Images in Layouts

Don’t ignore transfer costs

- watchOS 1 apps run extension on iPhone
- watchOS 2 apps need to install extension

Use appropriate sizes

Using Images in Layouts

Don’t ignore transfer costs

- watchOS 1 apps run extension on iPhone
- watchOS 2 apps need to install extension

Use appropriate sizes

Image slicing can accommodate various sizes

Animations

Tom Witkin
WatchKit Engineer

Existing Methods of Animation

Tables and animated images

Tables

Certain updates already animate

Tables

Certain updates already animate

Insert rows

Tables

Certain updates already animate

Insert rows

Remove rows

Tables

Certain updates already animate

Insert rows

Remove rows

Update row content


Recipes
10:09
Salmon
Pizza
Fish Tacos

Ranking	Alphabetical

Ranking	Alphabetical

Ranking | Alphabetical

Ranking	Alphabetical

Tables

Insert and remove rows

```
[self.recipeTable insertRowsAtIndexes:[NSIndexSet indexSetWithIndex:0]
withRowType:@"status"];

StatusRowController *rc = [self.recipeTable rowControllerAtIndex:0];
[rc setText:text];

self.statusRowTimer = [NSTimer scheduledTimerWithTimeInterval:2.0 target:self
selector:@selector(hideStatusRow) userInfo:nil repeats:NO];
```

Tables

Insert and remove rows

```
[self.recipeTable insertRowsAtIndexes:[NSIndexSet indexSetWithIndex:0]
withRowType:@"status"];
```

```
StatusRowController *rc = [self.recipeTable rowControllerAtIndex:0];
[rc setText:text];

self.statusRowTimer = [NSTimer scheduledTimerWithTimeInterval:2.0 target:self
selector:@selector(hideStatusRow) userInfo:nil repeats:NO];
```

Tables

Insert and remove rows

```
[self.recipeTable insertRowsAtIndexes:[NSIndexSet indexSetWithIndex:0]
withRowType:@"status"];
```

```
StatusRowController *rc = [self.recipeTable rowControllerAtIndex:0];
[rc setText:text];
```

```
self.statusRowTimer = [NSTimer scheduledTimerWithTimeInterval:2.0 target:self
selector:@selector(hideStatusRow) userInfo:nil repeats:NO];
```

Tables

Insert and remove rows

```
[self.recipeTable insertRowsAtIndexes:[NSIndexSet indexSetWithIndex:0]
withRowType:@"status"];

StatusRowController *rc = [self.recipeTable rowControllerAtIndex:0];
[rc setText:text];
```

```
self.statusRowTimer = [NSTimer scheduledTimerWithTimeInterval:2.0 target:self
selector:@selector(hideStatusRow) userInfo:nil repeats:NO];
```

Tables

Insert and remove rows

Tables

Insert and remove rows

Insert or remove rows of any type

Tables

Insert and remove rows

Insert or remove rows of any type

To avoid animations

```
setRowTypes()
setNumberOfRows(_:, withRowType:)
```

Tables

Insert and remove rows

Insert or remove rows of any type

To avoid animations

```
setRowTypes()
setNumberOfRows(_:, withRowType:)
```

10:09


Recipes
10:09
BBQ Wings
Fish Tacos
Pizza

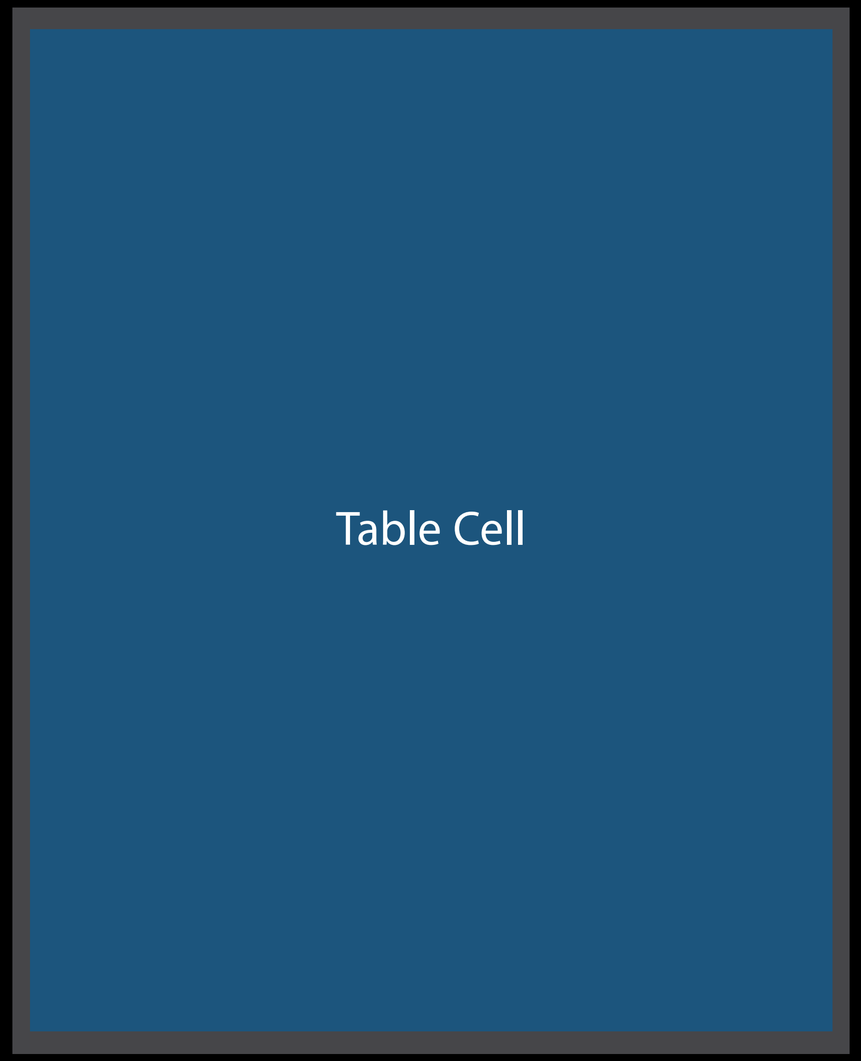

Recipes
10:09
BBQ Wings
Fish Tacos
Pizza


10:09
Celebrate with
delicious fish tacos....
Tap for more...
Servings
Ingredients

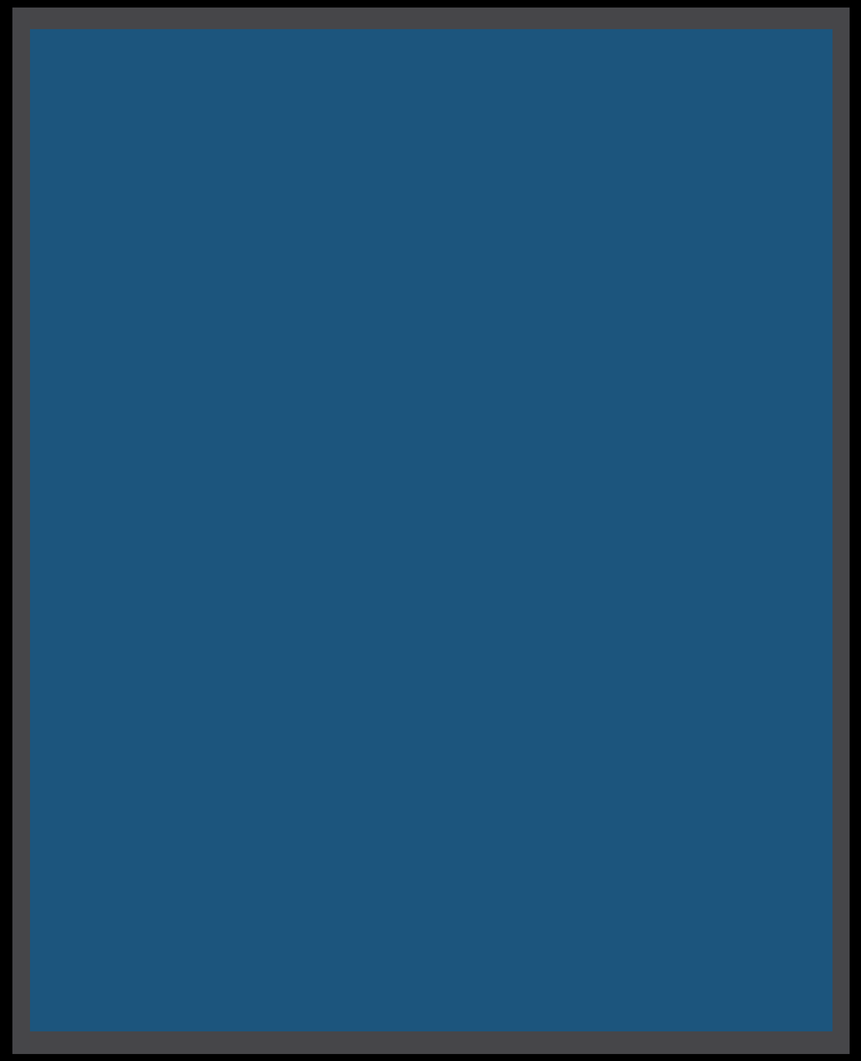

10:09
Celebrate with
delicious fish tacos....
Tap for more...
Servings
Ingredients

Groups as Spacing Elements

Groups as Spacing Elements



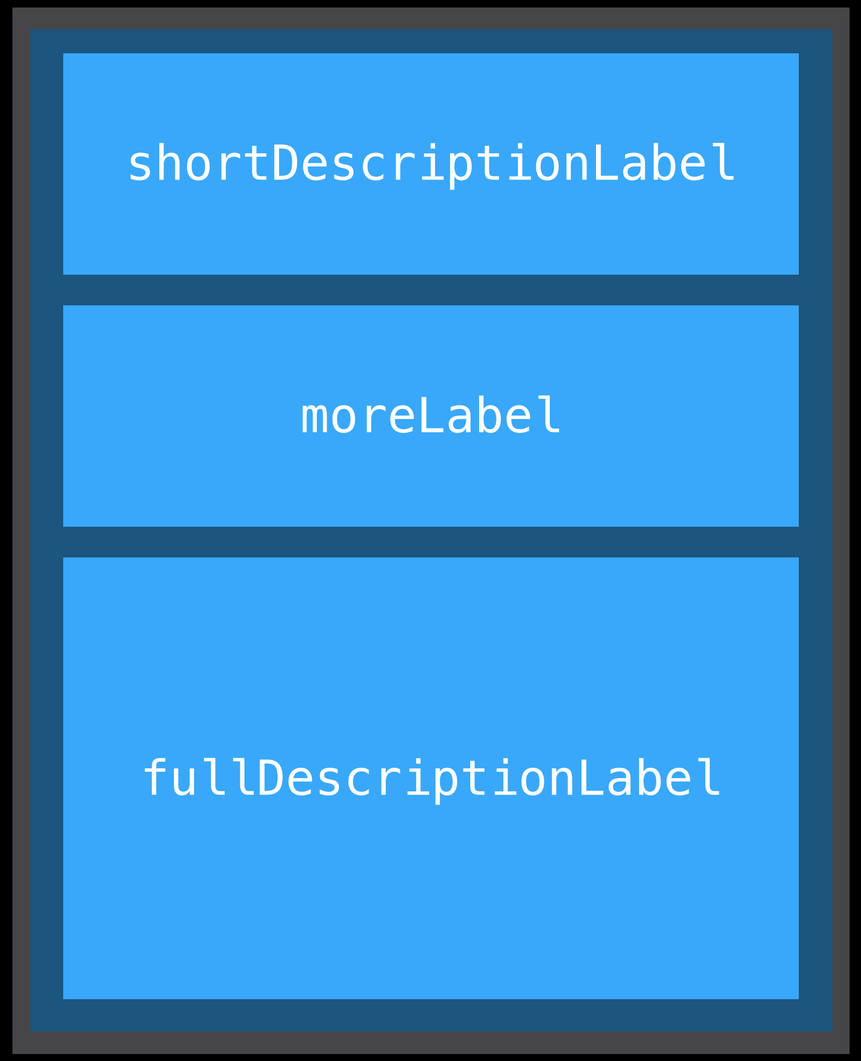
Groups as Spacing Elements



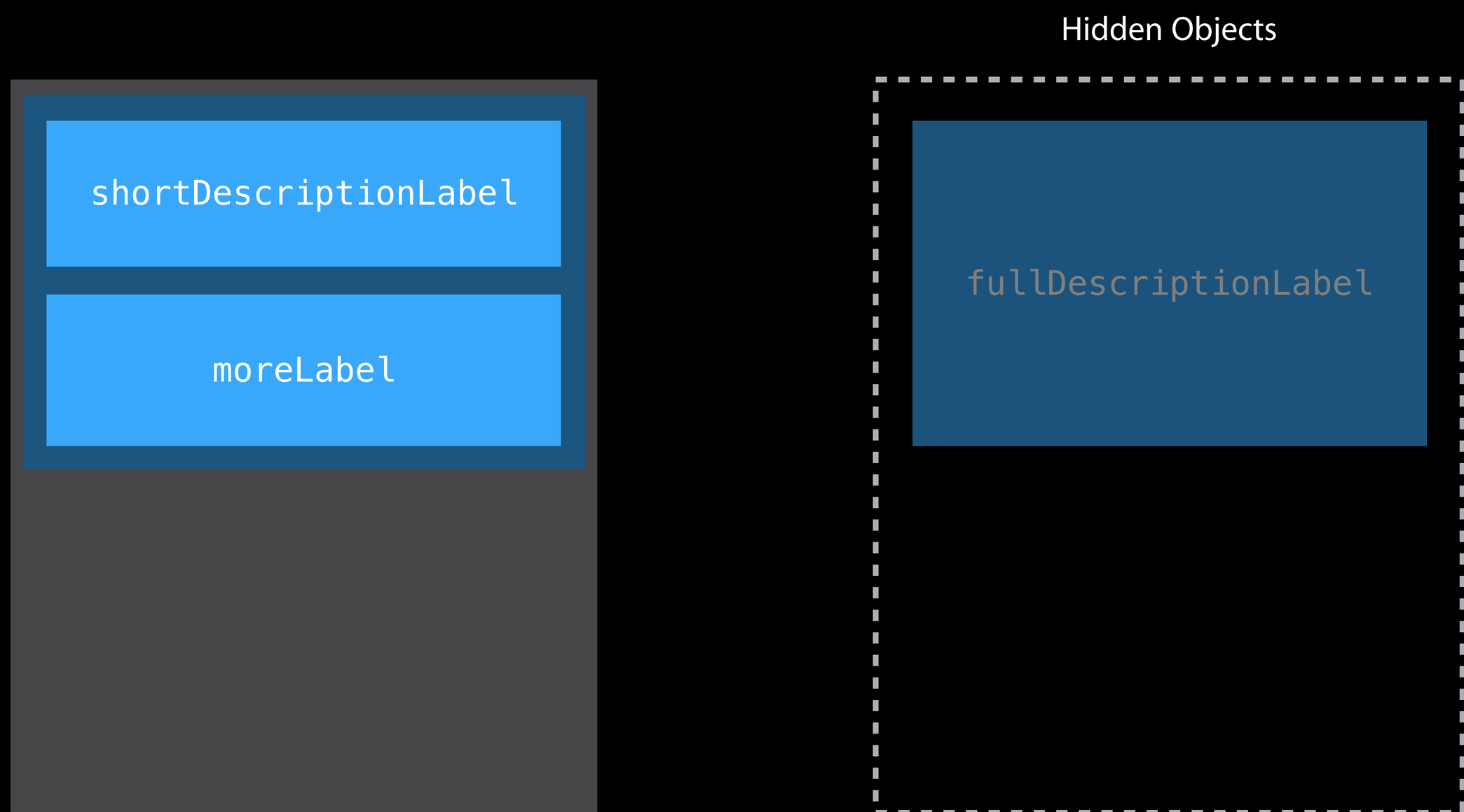
Groups as Spacing Elements

Groups as Spacing Elements

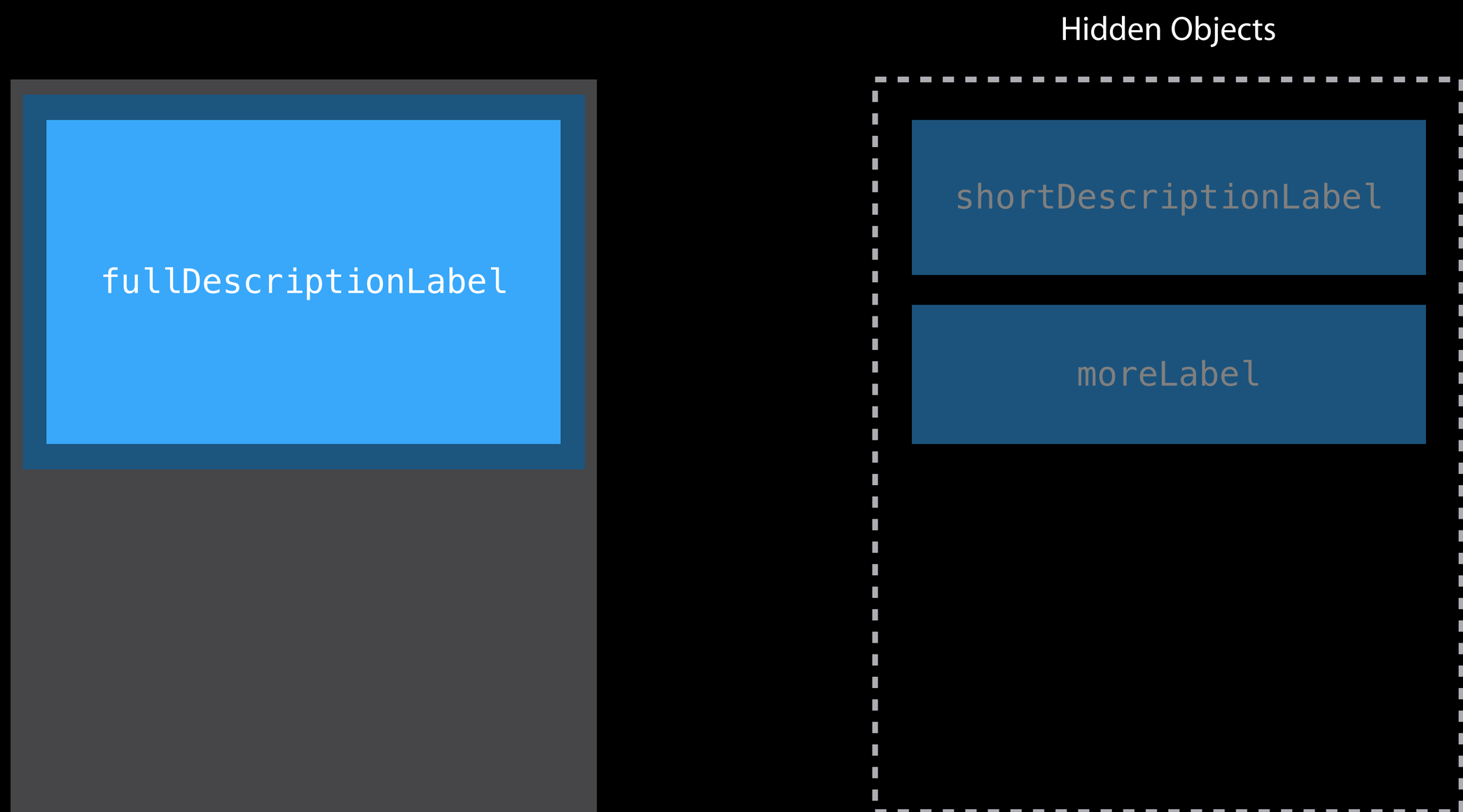
Groups as Spacing Elements



Groups as Spacing Elements



Groups as Spacing Elements



Tables

Reload row content

```
DescriptionRowController *rc = [self.table rowControllerAtIndex:0];
[rc.fullDescriptionLabel setHidden:NO];
[rc.shortDescriptionLabel setHidden:YES];
[rc.moreLabel setHidden:YES];
```

Tables

Reload row content

Tables

Reload row content

Rows reload when content changes in height

Tables

Reload row content

Rows reload when content changes in height

Make sure rows have size to fit height

Tables

Reload row content

Rows reload when content changes in height

Make sure rows have size to fit height

10:09


Recipes
10:09
BBQ Wings
Fish Tacos
Pizza


Recipes
10:09
BBQ Wings
Fish Tacos
Pizza


ALPHABETICAL
Fish Tacos
Pizza


ALPHABETICAL
Fish Tacos
Pizza


ALPHABETICAL
Fish Tacos
Pizza

Animated Images

Animated Images

Cycle through a series of images

Animated Images

Cycle through a series of images

Repeat and reverse animations

Animated Images

Cycle through a series of images

Repeat and reverse animations

Optimizing images is important

- Reduce size and number of images

WKInterfacePicker

NEW

WKInterfacePicker

NEW

Related Session

WatchKit In-Depth, Part 2 — WWDC15 Videos

NEW

Animation API

Animatable Properties

NEW

Animatable Properties

NEW

Opacity

Animatable Properties

NEW

Opacity

Width / height

Animatable Properties

NEW

Opacity

Width / height

Alignment

Animatable Properties

NEW

Opacity

Width / height

Alignment

Background color

Animatable Properties

NEW

Opacity

Width / height

Alignment

Background color

Color / tint color

Animatable Properties

NEW

Opacity

Width / height

Alignment

Background color

Color / tint color

Group insets

New API

WKInterfaceController

NEW

```
func animateWithDuration(duration: NSTimeInterval, animations: () -> Void)
```

Animation Examples


10:09
Celebrate with delicious fish tacos. For the engineers that want to add a bit of spice to their life.
Servings
Ingredients


10:09
Celebrate with delicious fish tacos. For the engineers that want to add a bit of spice to their life.
Servings
Ingredients


10:09
want to add a bit of spice to their life.
Servings
Ingredients
Notes


10:09
want to add a bit of spice to their life.
Servings
Ingredients
Notes


Servings
10:09
4


Servings
10:09
4

Sequential Animations

```
for (NSInteger i = 0; i < self.outerGroups.count; i++) {

    WKInterfaceGroup *group = self.outerGroups[i];

    dispatch_time_t time = dispatch_time(DISPATCH_TIME_NOW, (int64_t)delay);
    dispatch_after(time, dispatch_get_main_queue(), ^{

        [self animateWithDuration:duration animations:^{
            [group setAlpha:alpha];
        }];

    });

}
```

Sequential Animations

```
for (NSInteger i = 0; i < self.outerGroups.count; i++) {

    WKInterfaceGroup *group = self.outerGroups[i];

    dispatch_time_t time = dispatch_time(DISPATCH_TIME_NOW, (int64_t)delay);
    dispatch_after(time, dispatch_get_main_queue(), ^{

        [self animateWithDuration:duration animations:^{
            [group setAlpha:alpha];
        }];

    });

}
```

Sequential Animations

```
for (NSInteger i = 0; i < self.outerGroups.count; i++) {

    WKInterfaceGroup *group = self.outerGroups[i];

    dispatch_time_t time = dispatch_time(DISPATCH_TIME_NOW, (int64_t)delay);
    dispatch_after(time, dispatch_get_main_queue(), ^{

        [self animateWithDuration:duration animations:^{
            [group setAlpha:alpha];
        }];

    });

}
```

Sequential Animations

```
for (NSInteger i = 0; i < self.outerGroups.count; i++) {

    WKInterfaceGroup *group = self.outerGroups[i];

    dispatch_time_t time = dispatch_time(DISPATCH_TIME_NOW, (int64_t)delay);
    dispatch_after(time, dispatch_get_main_queue(), ^{

        [self animateWithDuration:duration animations:^{
            [group setAlpha:alpha];
        }];

    });

}
```

New API

WKInterfaceController

NEW

```
- (void)didAppear;
- (void)willDisappear;
```

New API

WKInterfaceController

NEW

```
- (void)didAppear;
- (void)willDisappear;
```

*Coming in a future watchOS 2 seed

Sequential Animations

Sequential Animations

Sequential Animations

Use `-didAppear`, not `-willActivate`

Sequential Animations

Use `-didAppear`, not `-willActivate`

Stagger animations using timers or GCD

Sequential Animations

Use `-didAppear`, not `-willActivate`

Stagger animations using timers or GCD

- Interface controller must be active

Sequential Animations

Use `-didAppear`, not `-willActivate`

Stagger animations using timers or GCD

- Interface controller must be active
- Keep total duration short

Sequential Animations

Use `-didAppear`, not `-willActivate`

Stagger animations using timers or GCD

- Interface controller must be active
- Keep total duration short

Set initial animation values in storyboard

Sequential Animations

Use `-didAppear`, not `-willActivate`

Stagger animations using timers or GCD

- Interface controller must be active
- Keep total duration short

Set initial animation values in storyboard

10:09


10:09
want to add a bit of spice to their life.
Servings
Ingredients
Notes


10:09
want to add a bit of spice to their life.
Servings
Ingredients
Notes

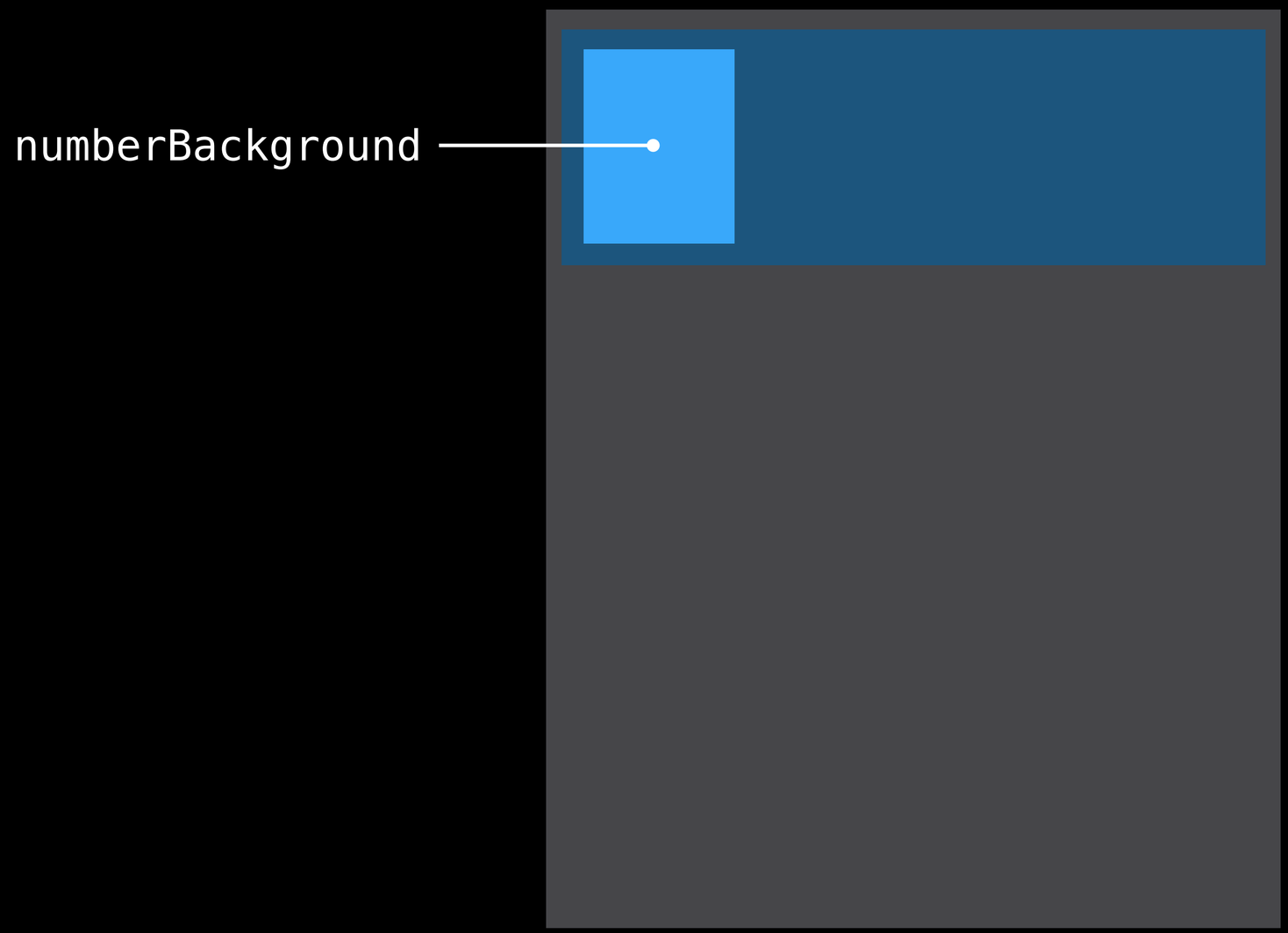

Ingredients
10:09


Ingredients
10:09

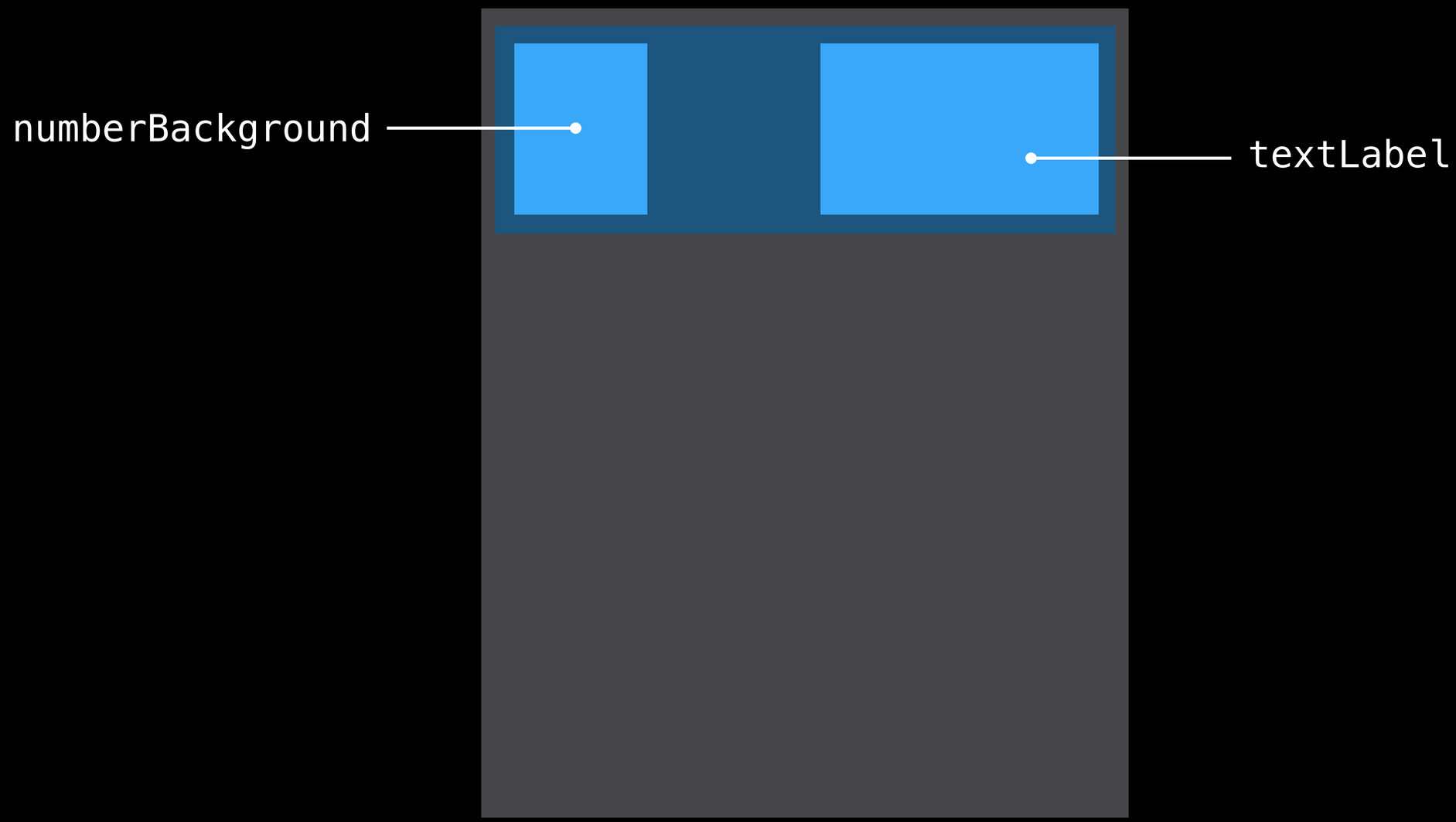
Groups as Spacing Elements

Groups as Spacing Elements

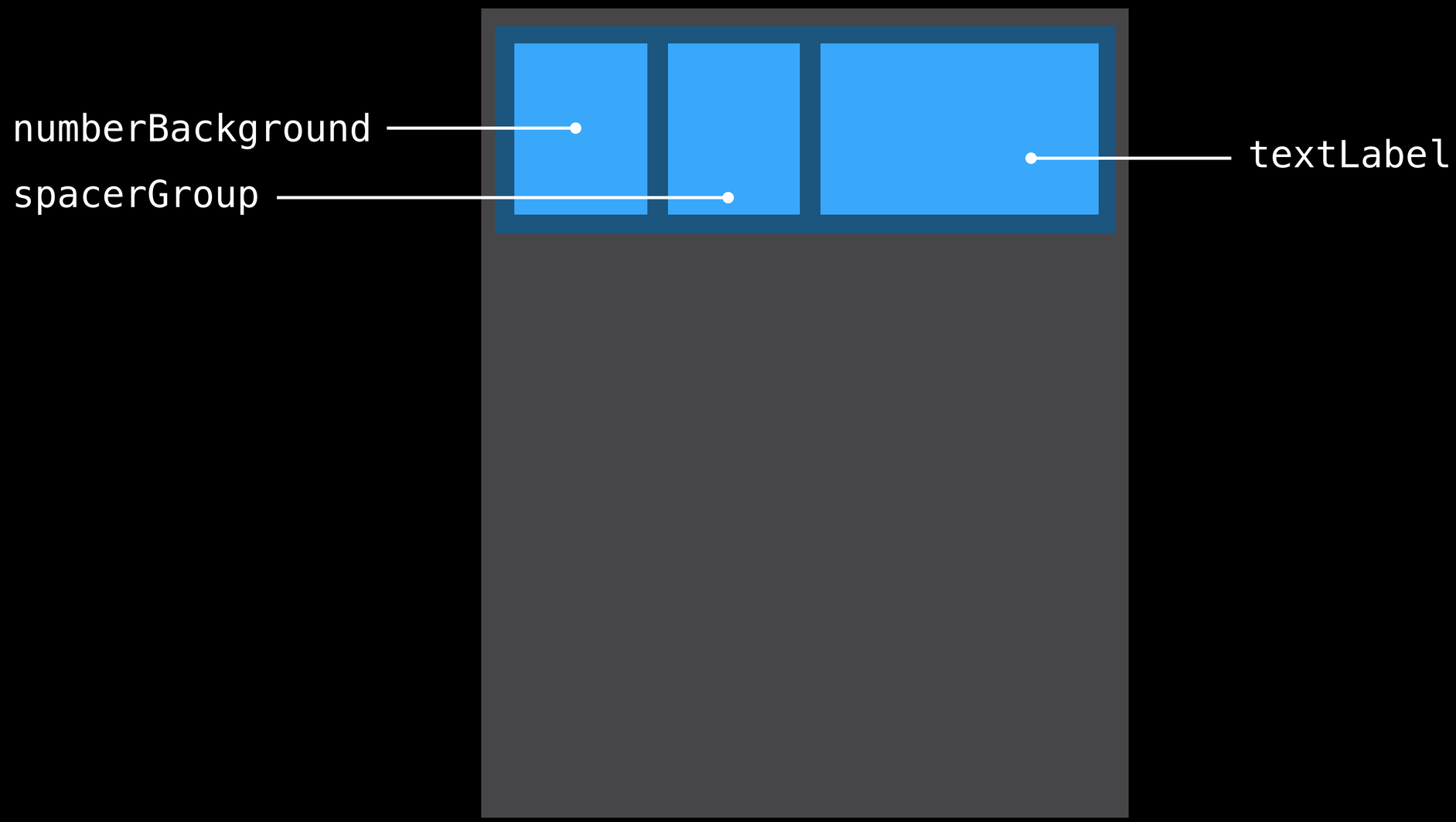
Groups as Spacing Elements



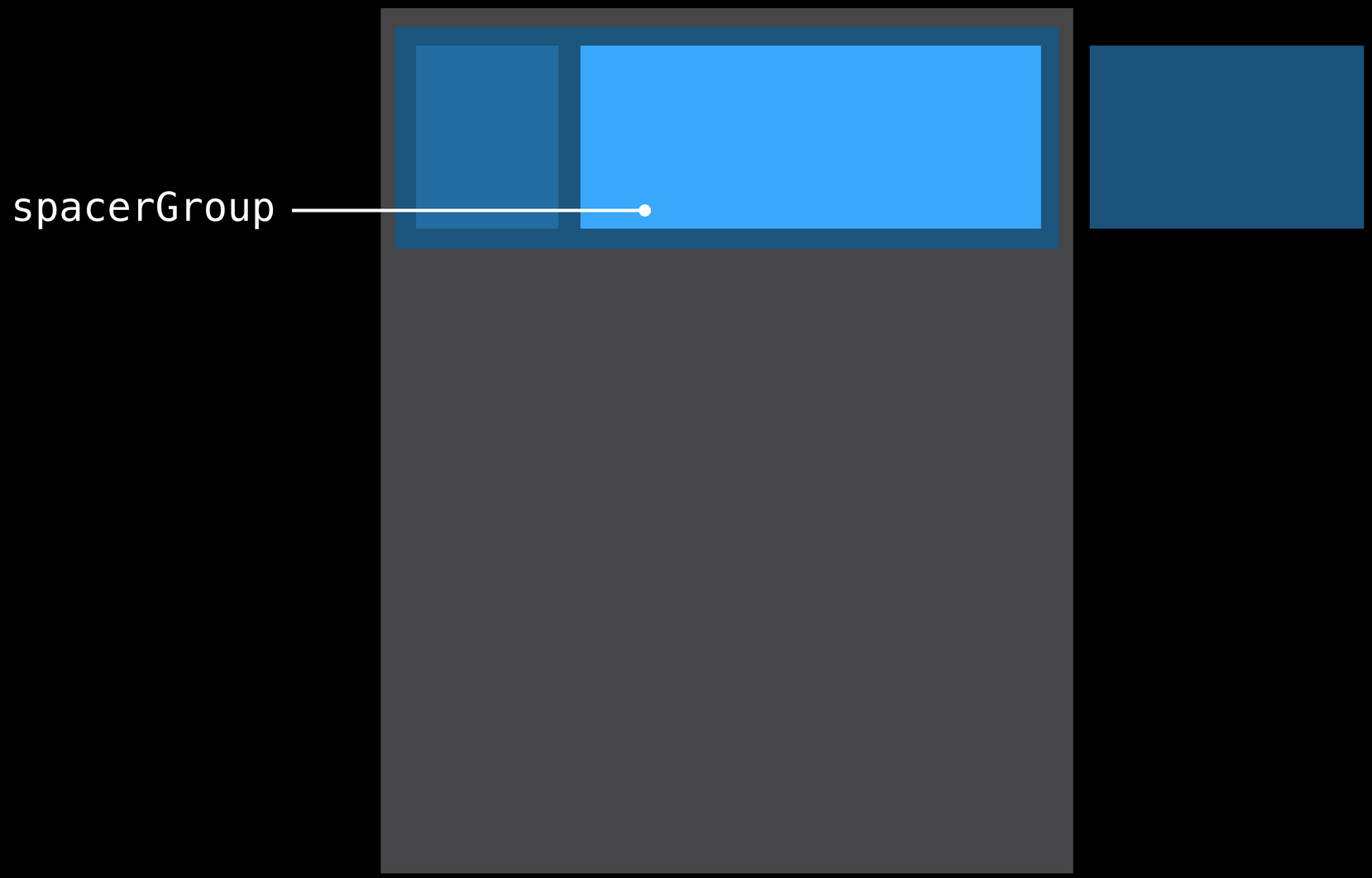
Groups as Spacing Elements



Groups as Spacing Elements

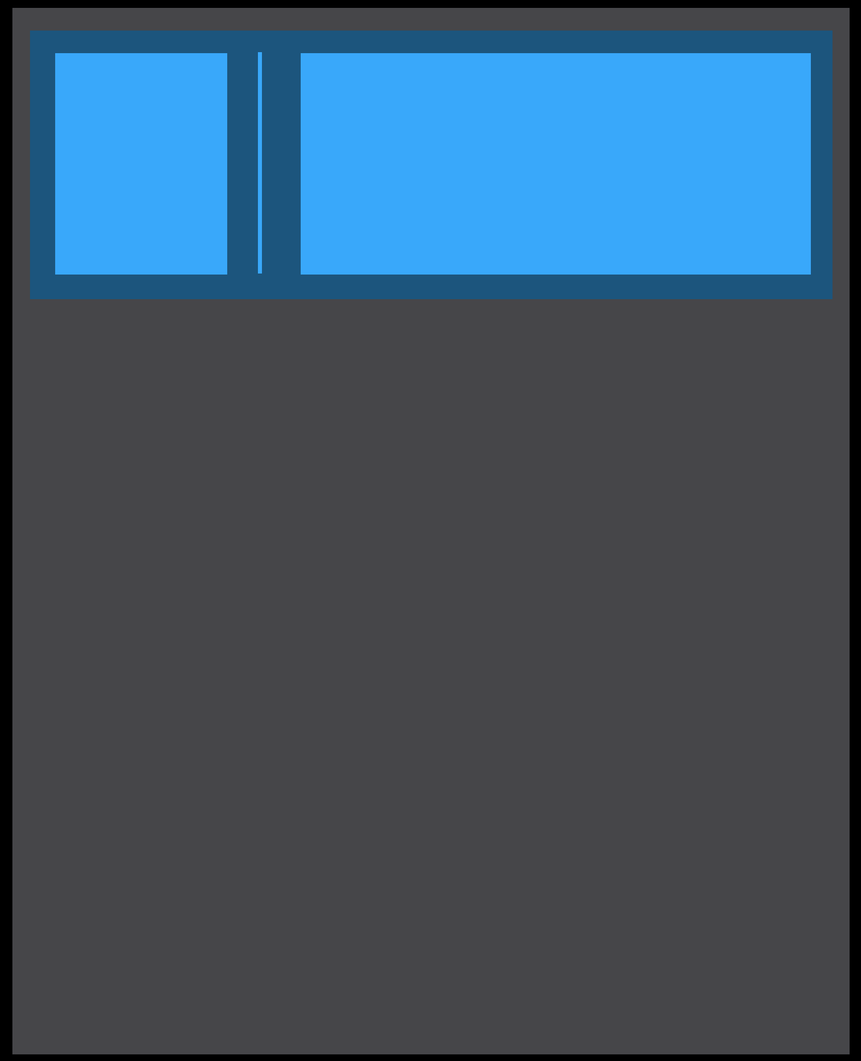


Groups as Spacing Elements

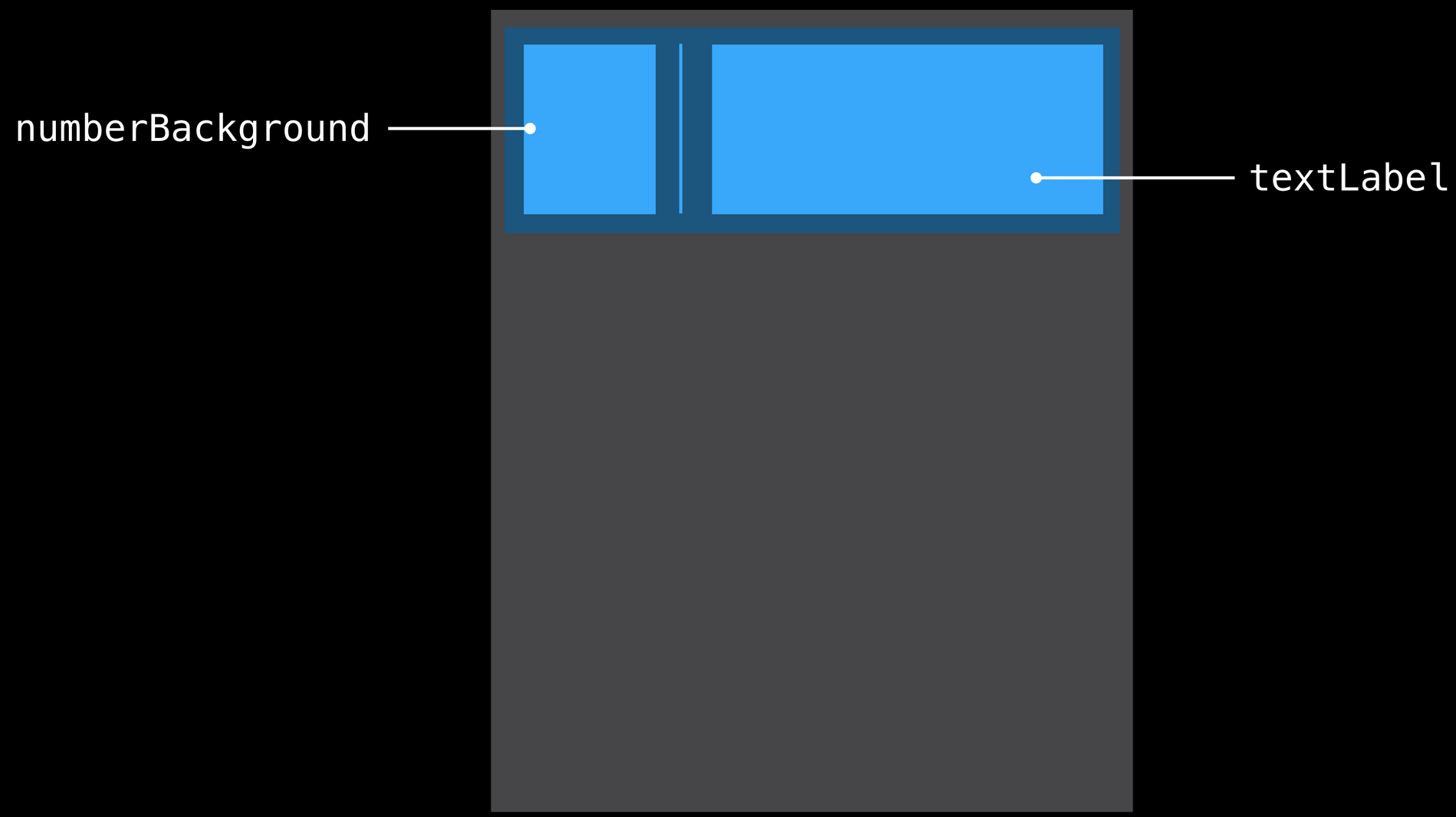


Groups as Spacing Elements

Groups as Spacing Elements



Groups as Spacing Elements



Groups as Spacing Elements

```
[self animateWithDuration:0.3 animations:^{

   for (NSInteger i = 0 ; i < self.instructionsTable.numberOfRows; i++) {

       IngredientRowController *rowController = [self.instructionsTable
       rowControllerAtIndex:i];

       [rowController.numberBackground setAlpha:1.0];
       [rowController.textLabel setAlpha:1.0];
       [rowController.spacerGroup setWidth:0.0];

   }

}];
```

Groups as Spacing Elements

```
[self animateWithDuration:0.3 animations:^{

    for (NSInteger i = 0 ; i < self.instructionsTable.numberOfRows; i++) {

        IngredientRowController *rowController = [self.instructionsTable
        rowControllerAtIndex:i];

        [rowController.numberBackground setAlpha:1.0];
        [rowController.textLabel setAlpha:1.0];
        [rowController.spacerGroup setWidth:0.0];

    }

}];
```

Groups as Spacing Elements

```
[self animateWithDuration:0.3 animations:^{

    for (NSInteger i = 0 ; i < self.instructionsTable.numberOfRows; i++) {

        IngredientRowController *rowController = [self.instructionsTable
        rowControllerAtIndex:i];

        [rowController.numberBackground setAlpha:1.0];
        [rowController.textLabel setAlpha:1.0];
        [rowController.spacerGroup setWidth:0.0];

    }

}];
```

Groups as Spacing Elements

```
[self animateWithDuration:0.3 animations:^{

    for (NSInteger i = 0 ; i < self.instructionsTable.numberOfRows; i++) {

        IngredientRowController *rowController = [self.instructionsTable
        rowControllerAtIndex:i];

        [rowController.numberBackground setAlpha:1.0];
        [rowController.textLabel setAlpha:1.0];
        [rowController.spacerGroup setWidth:0.0];

    }

}];
```

Groups as Spacing Elements

Groups as Spacing Elements

Groups as Spacing Elements

Invisible spacer groups

Groups as Spacing Elements

Invisible spacer groups

Adjust width, height, or alignment

Groups as Spacing Elements

Invisible spacer groups

Adjust width, height, or alignment

Animating will re-layout entire interface

Groups as Spacing Elements

Invisible spacer groups

Adjust width, height, or alignment

Animating will re-layout entire interface

10:09


10:09
want to add a bit of spice to their life.
Servings
Ingredients
Notes


10:09
want to add a bit of spice to their life.
Servings
Ingredients
Notes


Notes
10:09
Add Note


Notes
10:09
Add Note


Cancel
10:09
This is tasty.
Watch out! This is spicy.
I need to make this again.


Cancel
10:09


Cancel
10:09
Watch out! This is spicy.
Confirm


Cancel
10:09
Watch out! This is spicy.
Confirm


Cancel
10:09

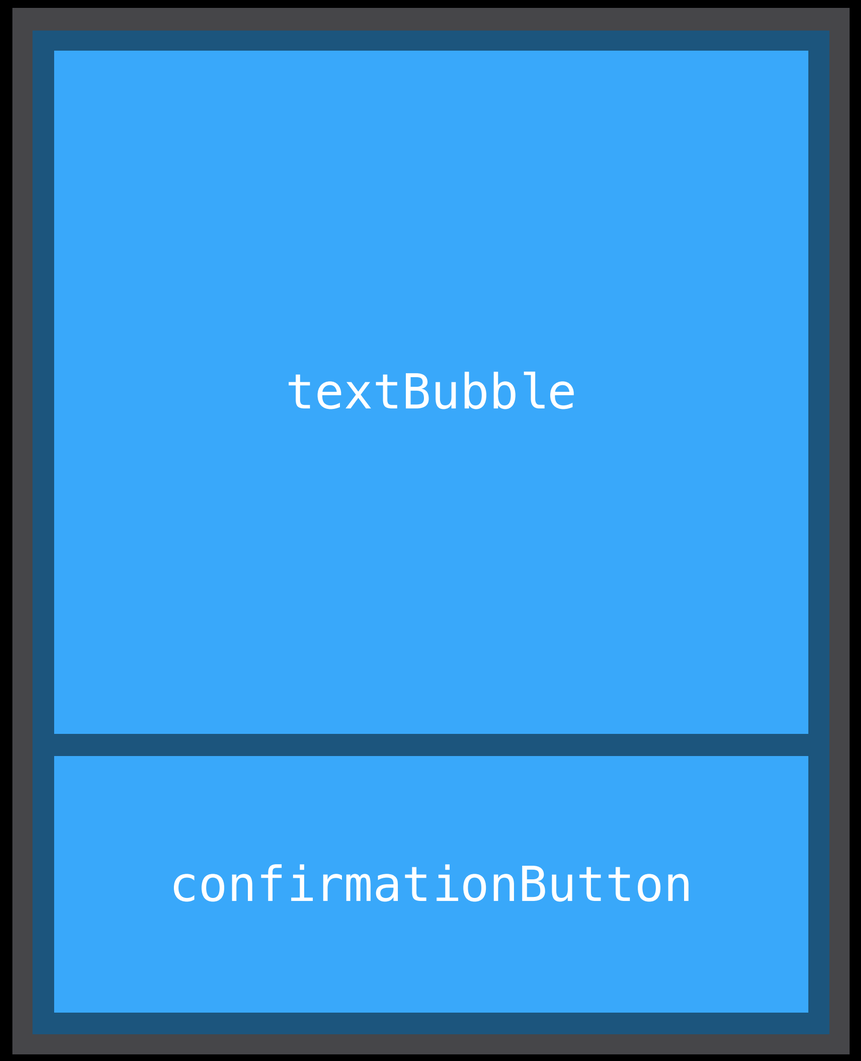

Cancel
10:09

Complex Interface Transitions

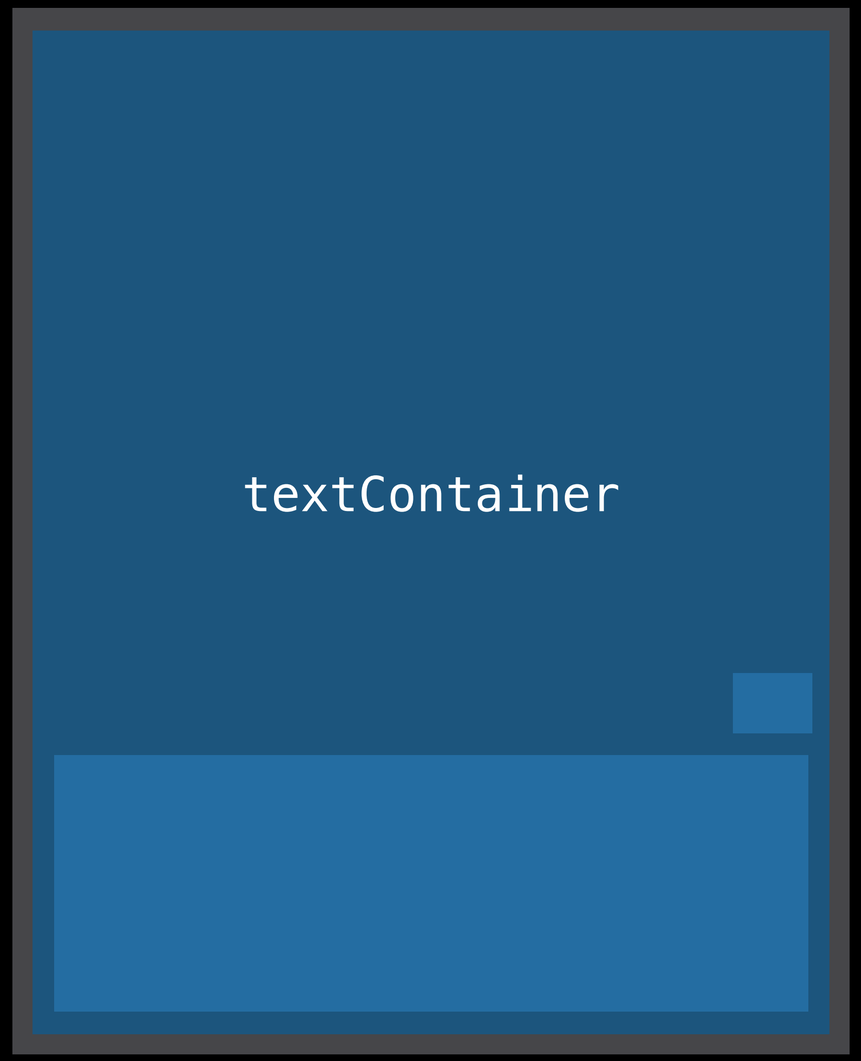
Complex Interface Transitions

Complex Interface Transitions

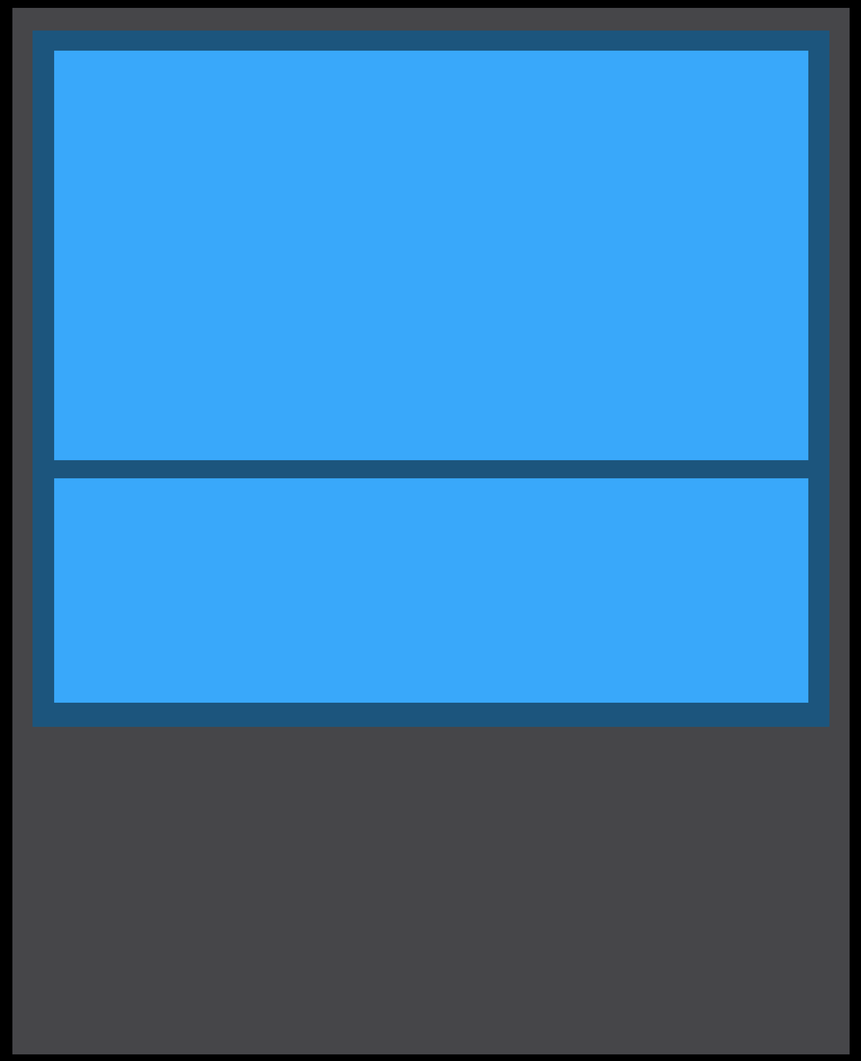
Complex Interface Transitions



Complex Interface Transitions



Complex Interface Transitions



Complex Interface Transitions

Complex Interface Transitions

```
[self animateWithDuration:duration animations:^{

    [self.textBubble setRelativeWidth:1.0 withAdjustment:0.0];
    [self.textBubble sizeToFitHeight];
    [self.textBubble setAlpha:1.0];

    [self.confirmationButton setAlpha:1.0];

    [self.textContainer sizeToFitHeight];

}];
```

Complex Interface Transitions

```
[self animateWithDuration:duration animations:^{

    [self.textBubble setRelativeWidth:1.0 withAdjustment:0.0];
    [self.textBubble sizeToFitHeight];
    [self.textBubble setAlpha:1.0];

    [self.confirmationButton setAlpha:1.0];

    [self.textContainer sizeToFitHeight];

}];
```

Complex Interface Transitions

```
[self animateWithDuration:duration animations:^{

    [self.textBubble setRelativeWidth:1.0 withAdjustment:0.0];
    [self.textBubble sizeToFitHeight];
    [self.textBubble setAlpha:1.0];

    [self.confirmationButton setAlpha:1.0];

    [self.textContainer sizeToFitHeight];

}];
```

Complex Interface Transitions

```
[self animateWithDuration:duration animations:^{

    [self.textBubble setRelativeWidth:1.0 withAdjustment:0.0];
    [self.textBubble sizeToFitHeight];
    [self.textBubble setAlpha:1.0];

    [self.confirmationButton setAlpha:1.0];

    [self.textContainer sizeToFitHeight];

}];
```

Complex Interface Transitions

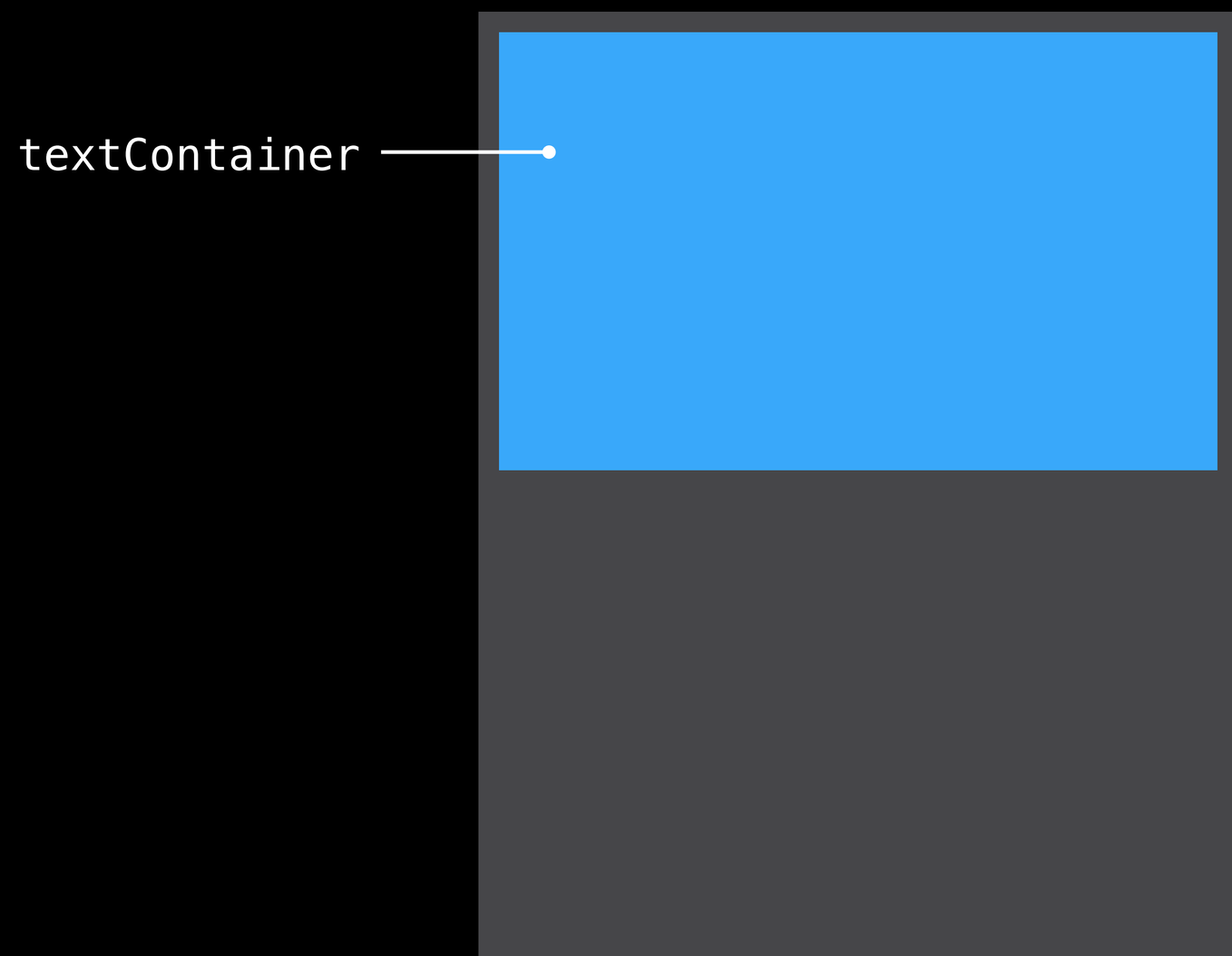
```
[self animateWithDuration:duration animations:^{

    [self.textBubble setRelativeWidth:1.0 withAdjustment:0.0];
    [self.textBubble sizeToFitHeight];
    [self.textBubble setAlpha:1.0];

    [self.confirmationButton setAlpha:1.0];

    [self.textContainer sizeToFitHeight];

}];
```


Cancel
10:09
Watch out! This is spicy.
Confirm

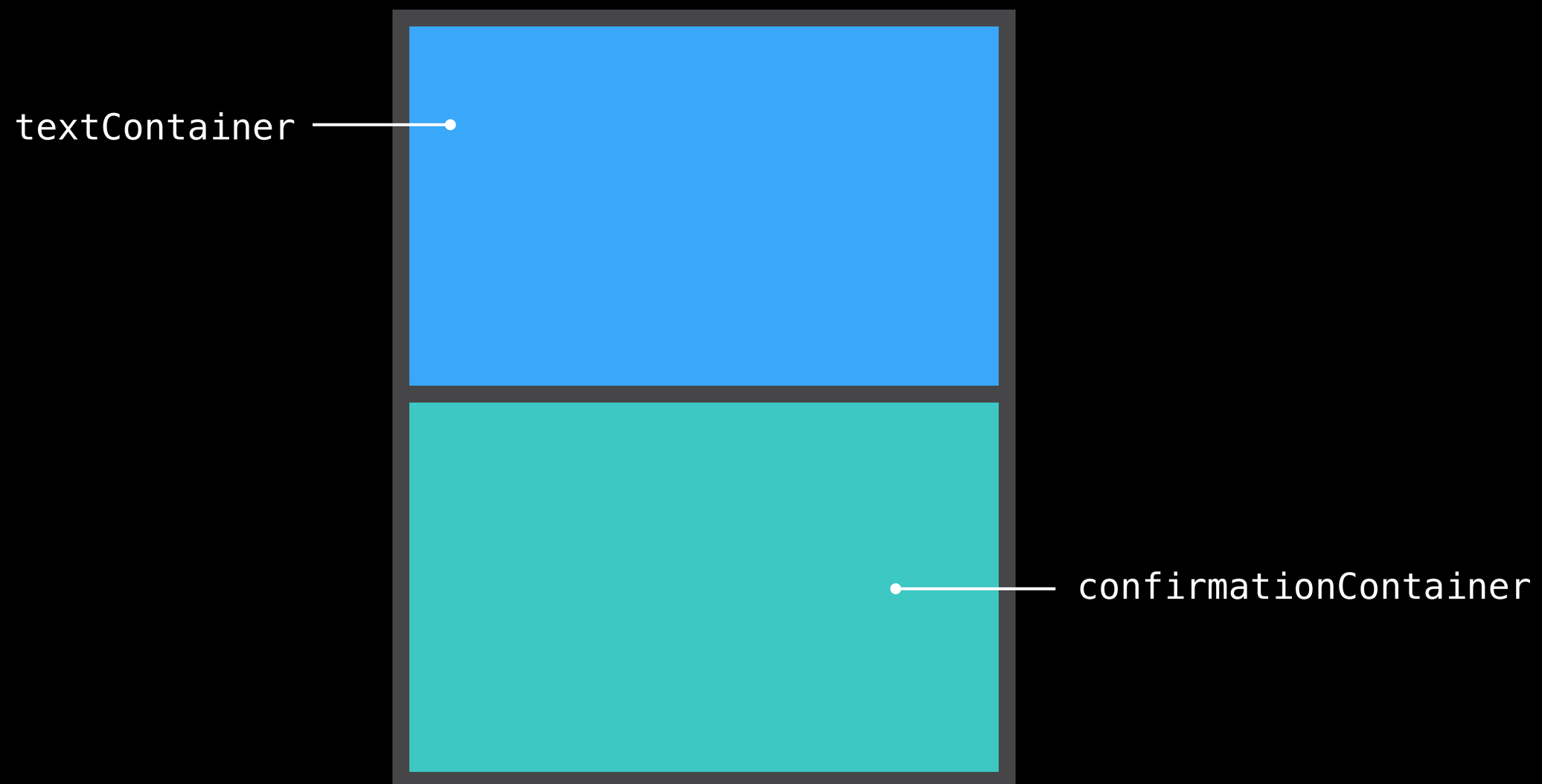

Cancel
10:09
Watch out! This is spicy.
Confirm

Complex Interface Transitions

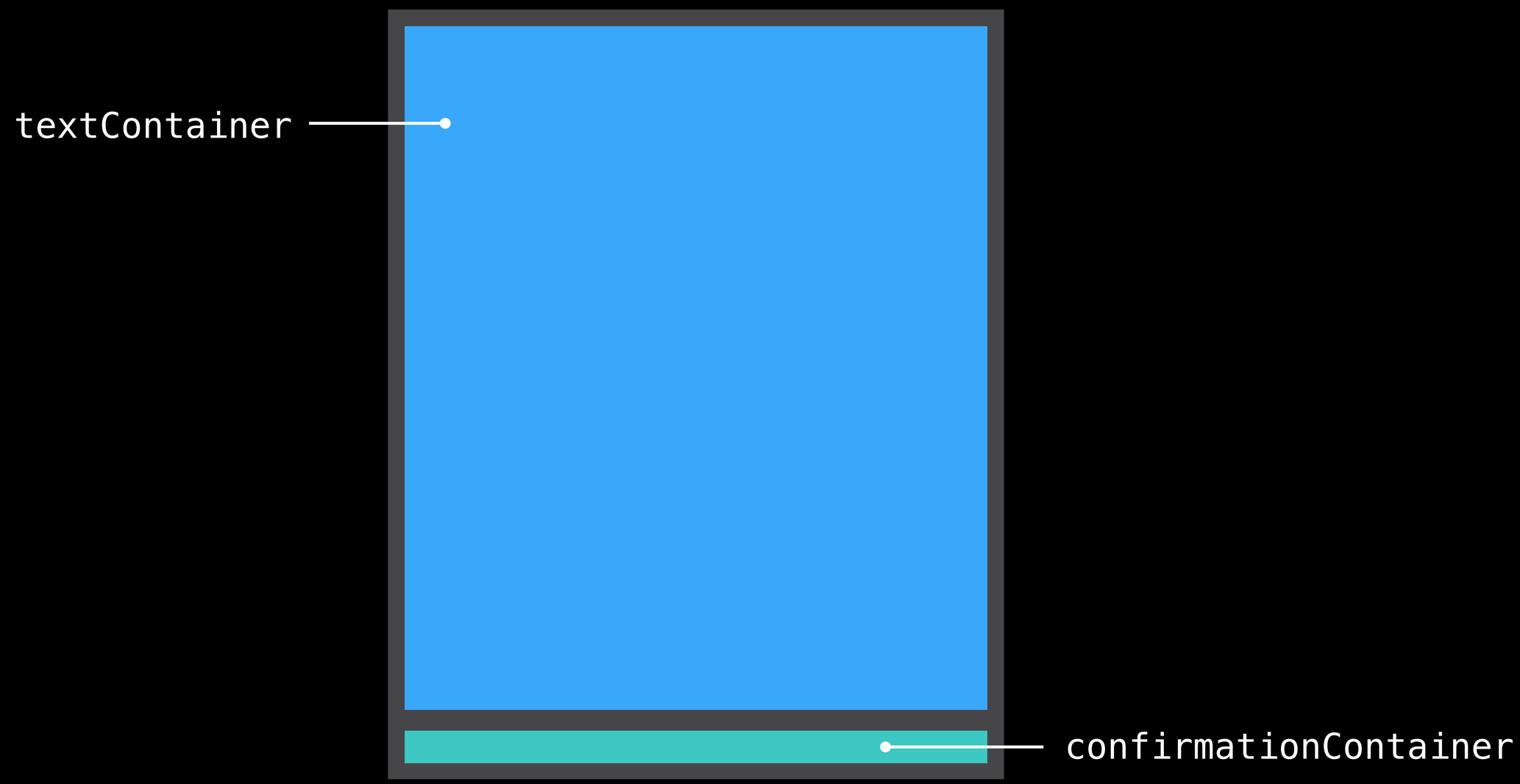
Complex Interface Transitions



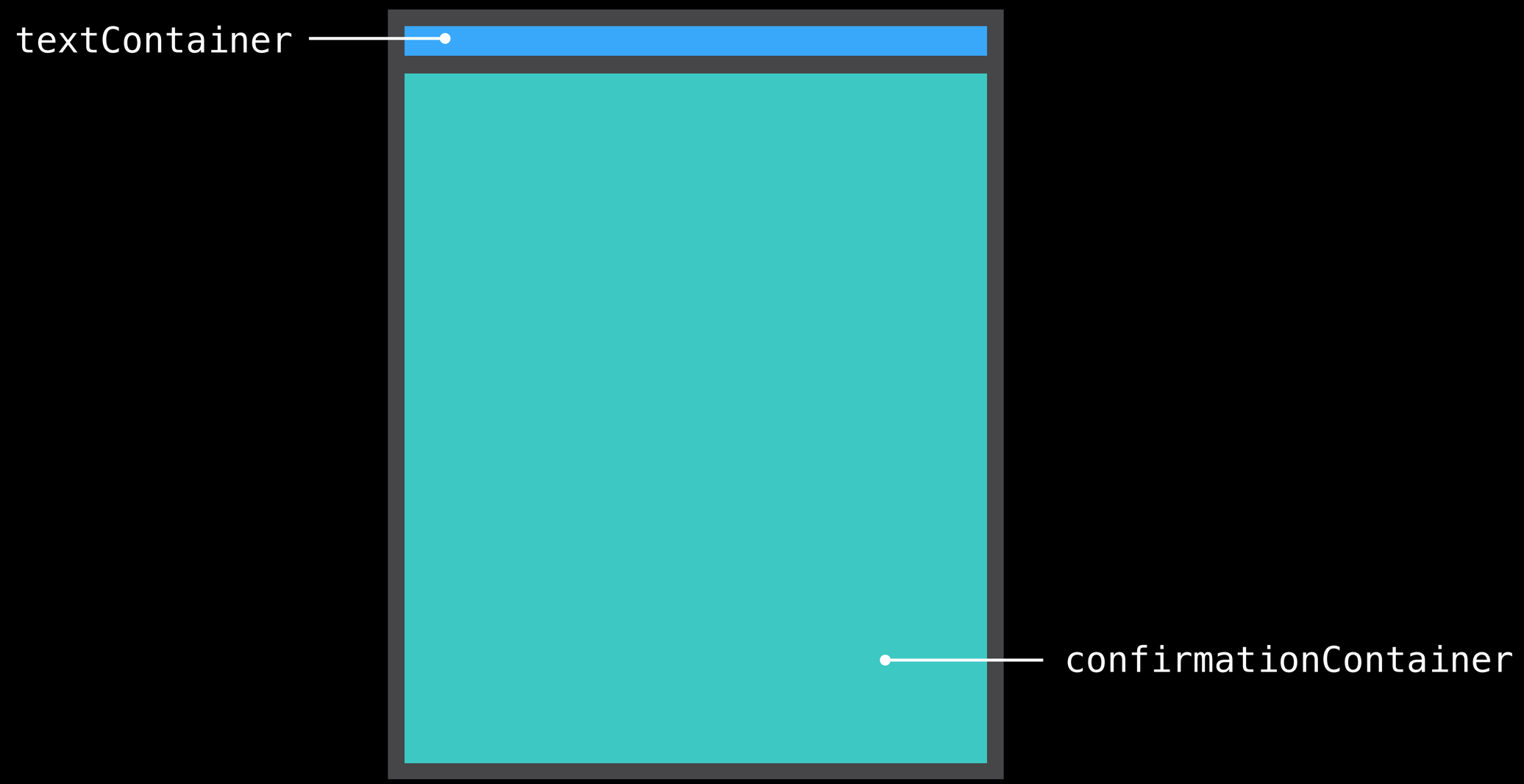
Complex Interface Transitions



Complex Interface Transitions



Complex Interface Transitions



Complex Interface Transitions

```
[self animateWithDuration:duration animations:^{

    [self.textContainer setAlpha:0.0];
    [self.textContainer setHeight:0.0];

    [self.confirmationContainer setRelativeHeight:1.0 withAdjustment:0.0];

}];
```

Complex Interface Transitions

```
[self animateWithDuration:duration animations:^{

    [self.textContainer setAlpha:0.0];
    [self.textContainer setHeight:0.0];

    [self.confirmationContainer setRelativeHeight:1.0 withAdjustment:0.0];

}];
```

Complex Interface Transitions

```
[self animateWithDuration:duration animations:^{

    [self.textContainer setAlpha:0.0];
    [self.textContainer setHeight:0.0];

    [self.confirmationContainer setRelativeHeight:1.0 withAdjustment:0.0];

}];
```

Complex Interface Transitions

```
[self animateWithDuration:duration animations:^{

    [self.textContainer setAlpha:0.0];
    [self.textContainer setHeight:0.0];

    [self.confirmationContainer setRelativeHeight:1.0 withAdjustment:0.0];

}];
```

Animation

A few notes

Animation

A few notes

Any update that affects sizing can animate layout

Animation

A few notes

Any update that affects sizing can animate layout

- Example—WKInterfaceLabel text

Animation

A few notes

Any update that affects sizing can animate layout

- Example—WKInterfaceLabel text

Concurrent animations and complex layouts affect performance

Animation

A few notes

Any update that affects sizing can animate layout

- Example—WKInterfaceLabel text

Concurrent animations and complex layouts affect performance

- Test on hardware!

Animation

A few notes

Any update that affects sizing can animate layout

- Example—WKInterfaceLabel text

Concurrent animations and complex layouts affect performance

- Test on hardware!

API functions within apps, not Glances or dynamic notifications

Animation

A few more notes

Animation

A few more notes

Use with restraint

Animation

A few more notes

Use with restraint

Should never be the focus

Animation

A few more notes

Use with restraint

Should never be the focus

Keep duration short

Related Session

Designing with Animation	Presidio	Thursday 3:30PM

Summary

Summary

Layout

Summary

Layout

- Specified at design time

Summary

Layout

- Specified at design time
- Flow-based

Summary

Layout

- Specified at design time
- Flow-based
- Groups are powerful

Summary

Layout

- Specified at design time
- Flow-based
- Groups are powerful

Animation

Summary

Layout

- Specified at design time
- Flow-based
- Groups are powerful

Animation

- Add liveliness and feedback to your app

Summary

Layout

- Specified at design time
- Flow-based
- Groups are powerful

Animation

- Add liveliness and feedback to your app
- Tables and images can already animate

Summary

Layout

- Specified at design time
- Flow-based
- Groups are powerful

Animation

- Add liveliness and feedback to your app
- Tables and images can already animate
- New API in watchOS 2

More Information

Documentation

watchOS 2 Transition Guide

WatchKit Programming Guide

Sample Code

WKRecipes

WatchKit Catalog

http://developer.apple.com/watchOS

Technical Support

Apple Developer Forums

Developer Technical Support

General Inquiries

Jake Behrens,watchOS Frameworks Evangelist

behrens@apple.com

Related Sessions

Introducing WatchKit for watchOS 2	WWDC15 Videos	
Building Watch Apps	WWDC15 Videos	
WatchKit In-Depth, Part 1	WWDC15 Videos	
WatchKit In-Depth, Part 2	WWDC15 Videos	
Designing for Apple Watch	WWDC15 Videos	
WatchKit Tips and Tricks	Presidio	Friday 10:00AM
Apple Watch Design Tips and Tricks	Presidio	Friday 3:30PM

Labs

WatchKit Layout and Animation Lab	Frameworks Lab B	Thursday 3:30PM
WatchKit and ClockKit Complications Lab	Frameworks Lab A	Friday 1:30PM

WWDC15