

Trait Collections

iOS 12 is compatible with these devices.

iPhone



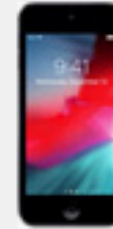
iPhone Xs
iPhone Xs Max
iPhone XR
iPhone X
iPhone 8
iPhone 8 Plus
iPhone 7
iPhone 7 Plus
iPhone 6s
iPhone 6s Plus
iPhone 6
iPhone 6 Plus
iPhone SE
iPhone 5s

iPad

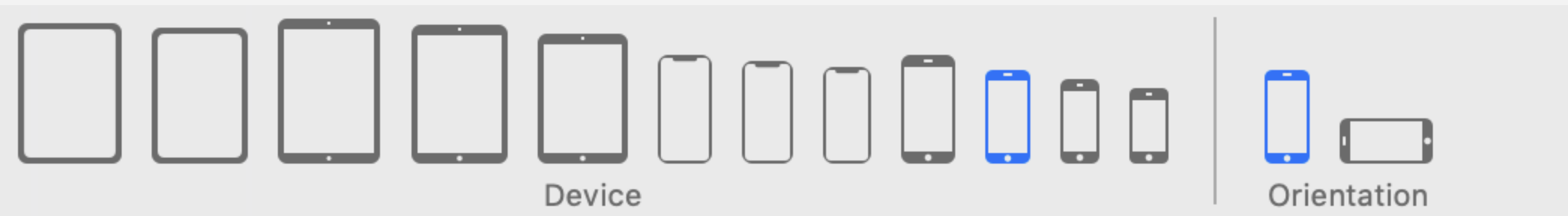


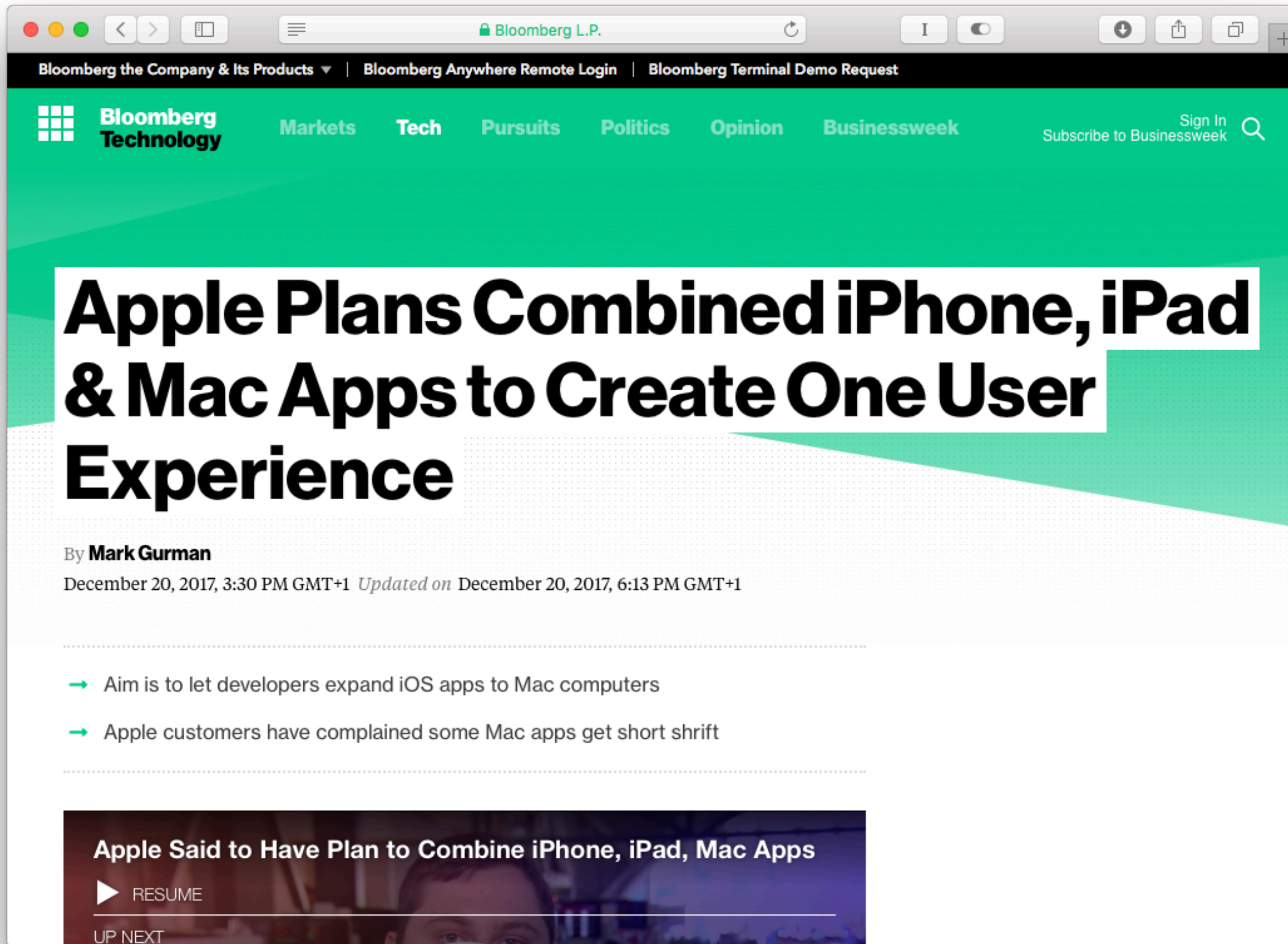
12.9-inch iPad Pro
2nd generation
12.9-inch iPad Pro
1st generation
10.5-inch iPad Pro
9.7-inch iPad Pro
iPad 6th generation
iPad 5th generation
iPad Air 2
iPad Air
iPad mini 4
iPad mini 3
iPad mini 2

iPod



iPod touch 6th
generation





Apple Plans Combined iPhone, iPad & Mac Apps to Create One User Experience

By **Mark Gurman**

December 20, 2017, 3:30 PM GMT+1 *Updated on* December 20, 2017, 6:13 PM GMT+1

- Aim is to let developers expand iOS apps to Mac computers
- Apple customers have complained some Mac apps get short shrift

Apple Said to Have Plan to Combine iPhone, iPad, Mac Apps

▶ RESUME

UP NEXT



Finder File Edit View Go Window Help

FAVORITES
AirDrop
iCloud Drive
Applications
Aurelien
Pictures
DEVICES
Macbook Pro
Hard Drive

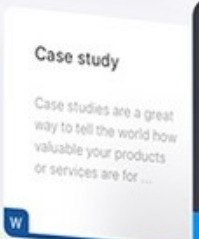
Aurelien
Documents



Applications

Design

Downloads



Open
Open With >
Move to Trash

Get Info

Rename
Duplicate
Make Alias
Share >

Copy "macOS UI Kit"

Tags



...



MacBook Pro

macOS app

AppKit

UIKit

Trackpad and mouse input

App window stoplights

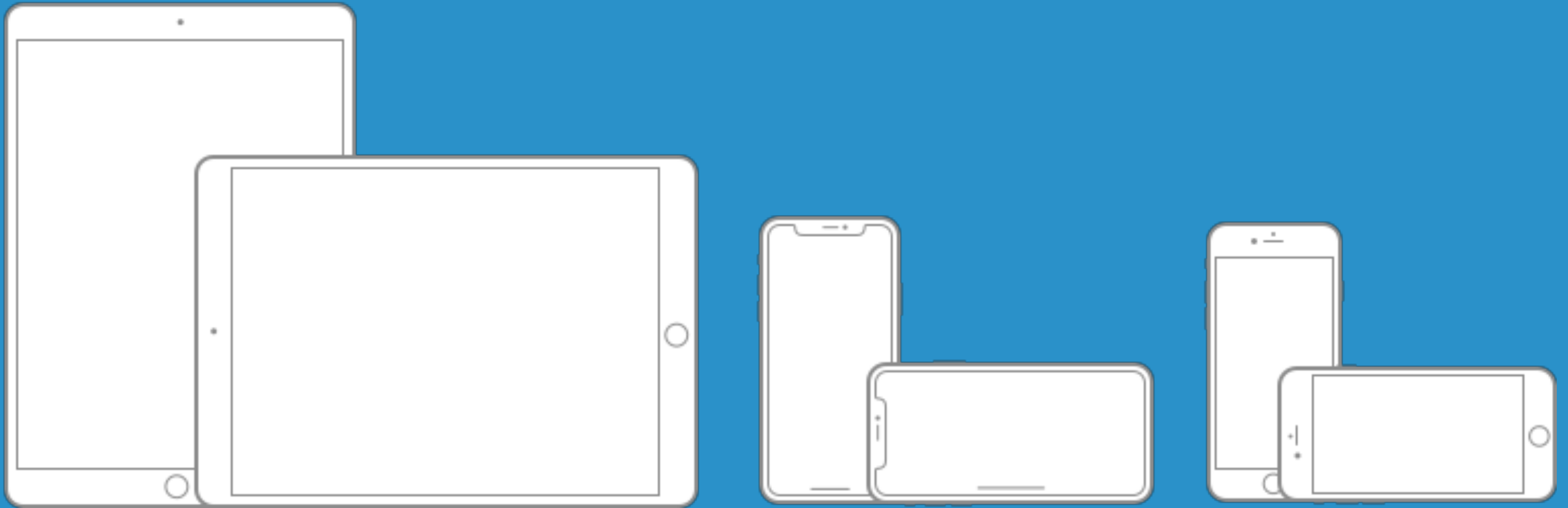
Window resizing

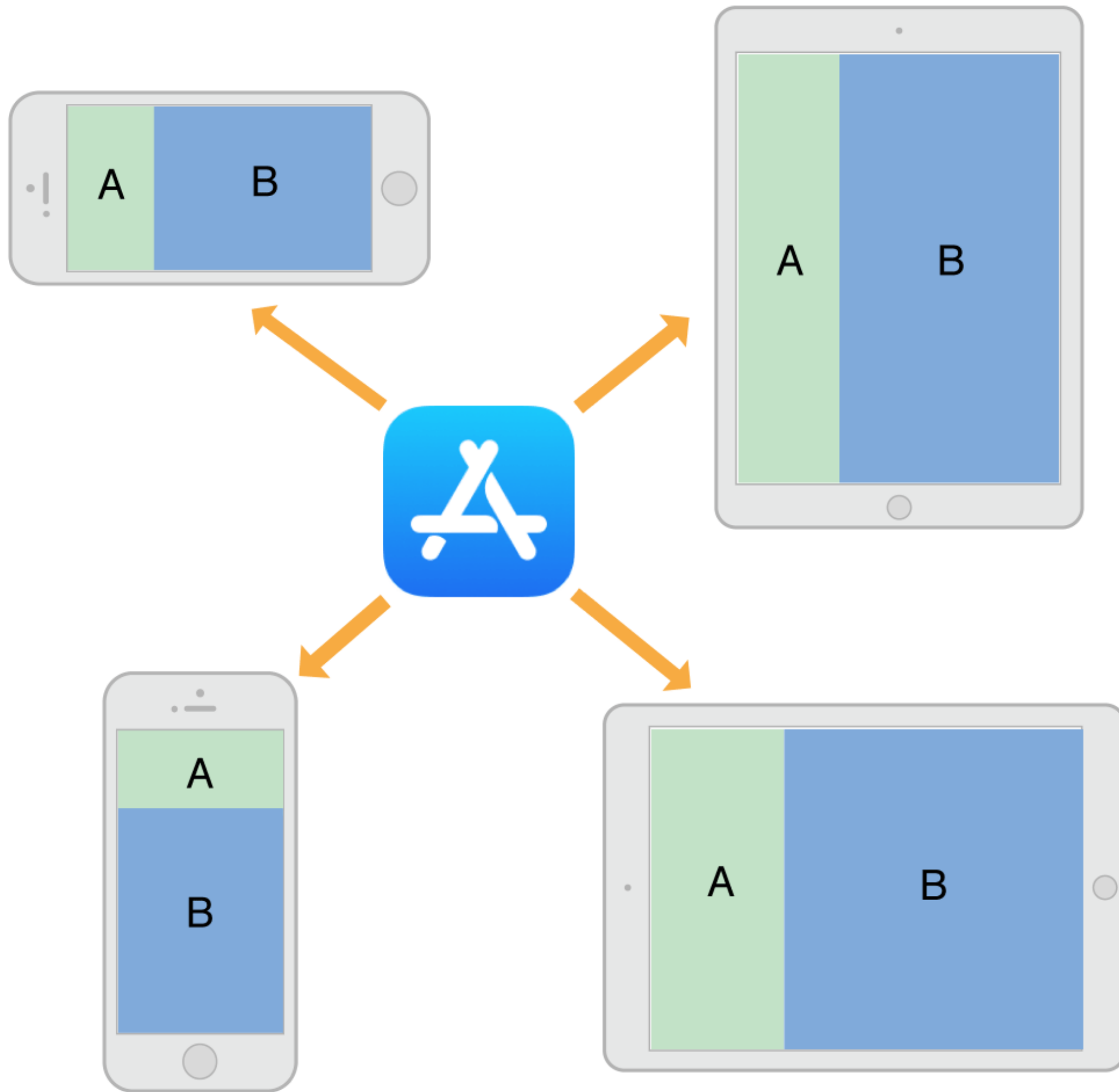
Scroll-bars

Copy and paste

Drag and drop

Trait collections allows us to adapt the interface, not only to screen sizes, but to different devices.





Traits:

- `horizontalSizeClass`
- `verticalSizeClass`
- `displayScale`
- `displayGamut`
- `forceTouchCapability`
- `preferredContentSizeCategory`
- `userInterfaceIdiom`

Use these traits to customize your layout.

Traits:

- All these properties are defined in `UITraitCollection`.

Traits:

- The `traitCollection` can be accessed on any class that conforms to:

```
@protocol UITraitEnvironment <NSObject>
```

```
@property UITraitCollection *traitCollection;
```

```
- (void)traitCollectionDidChange:(UITraitCollection *)previousTraitCollection;
```

```
@end
```

Traits:

All these classes conform to `UITraitEnvironment`

- `UIScreen`
- `UIWindow`
- `UIViewController`
- `UIPresentationController`
- `UIView`

Traits:

- The traits should have a correct value when added to the view/viewController hierarchy, not at init time.
- You can override them using `setOverrideTraitCollection:forChildViewController:`

Size Classes:

They are just one of the many traits available

Size Classes:

```
typedef NSInteger, UIUserInterfaceSizeClass) {  
    UIUserInterfaceSizeClassUnspecified = 0,  
    UIUserInterfaceSizeClassCompact    = 1,  
    UIUserInterfaceSizeClassRegular     = 2,  
} NS_ENUM_AVAILABLE_IOS(8_0);
```

Some recommendations:

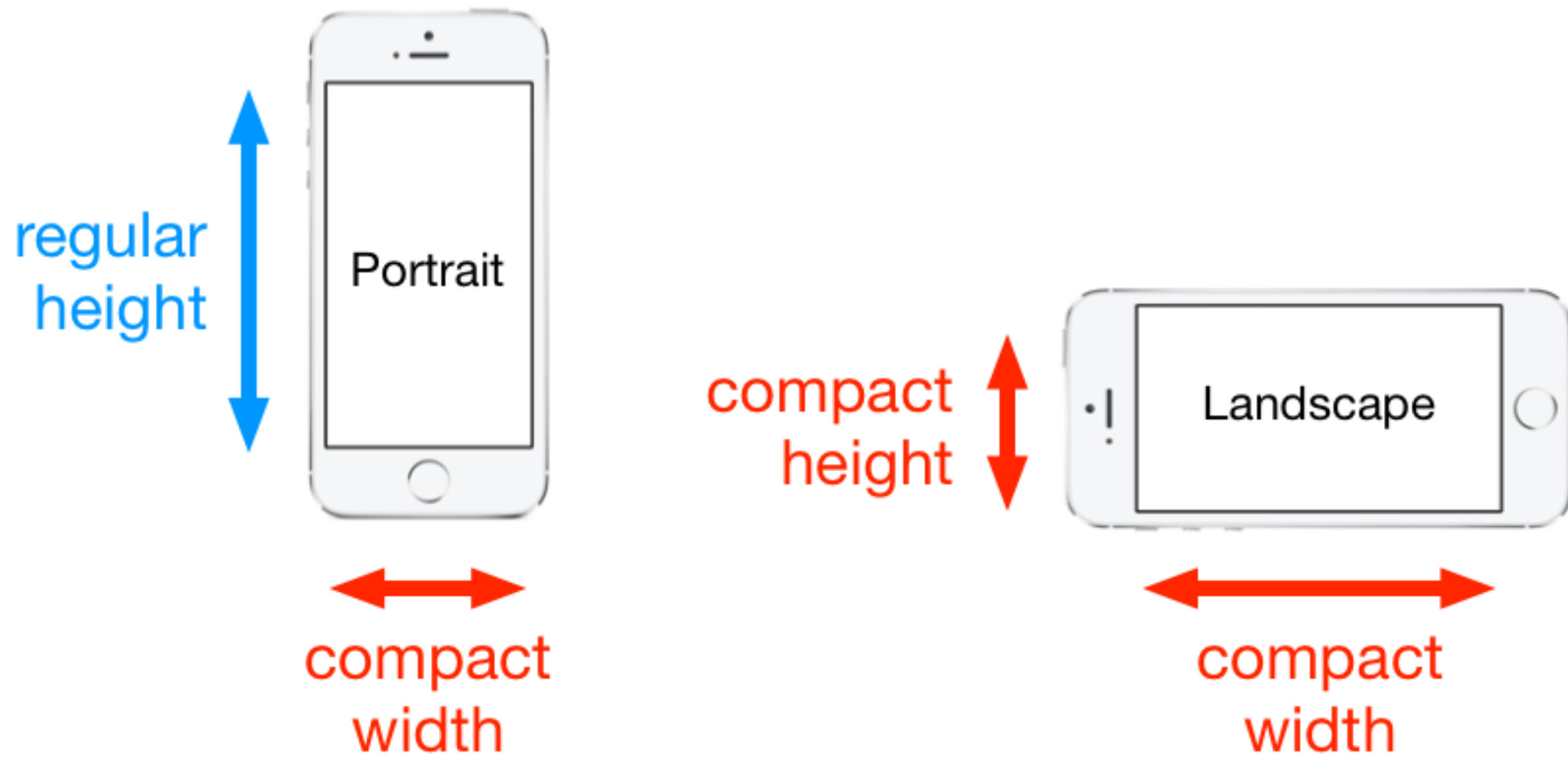
Never assume that a size class corresponds to the specific width or height of a view.

**Avoid using idiom
information to make
decisions about the layout
or content of your
interface.**

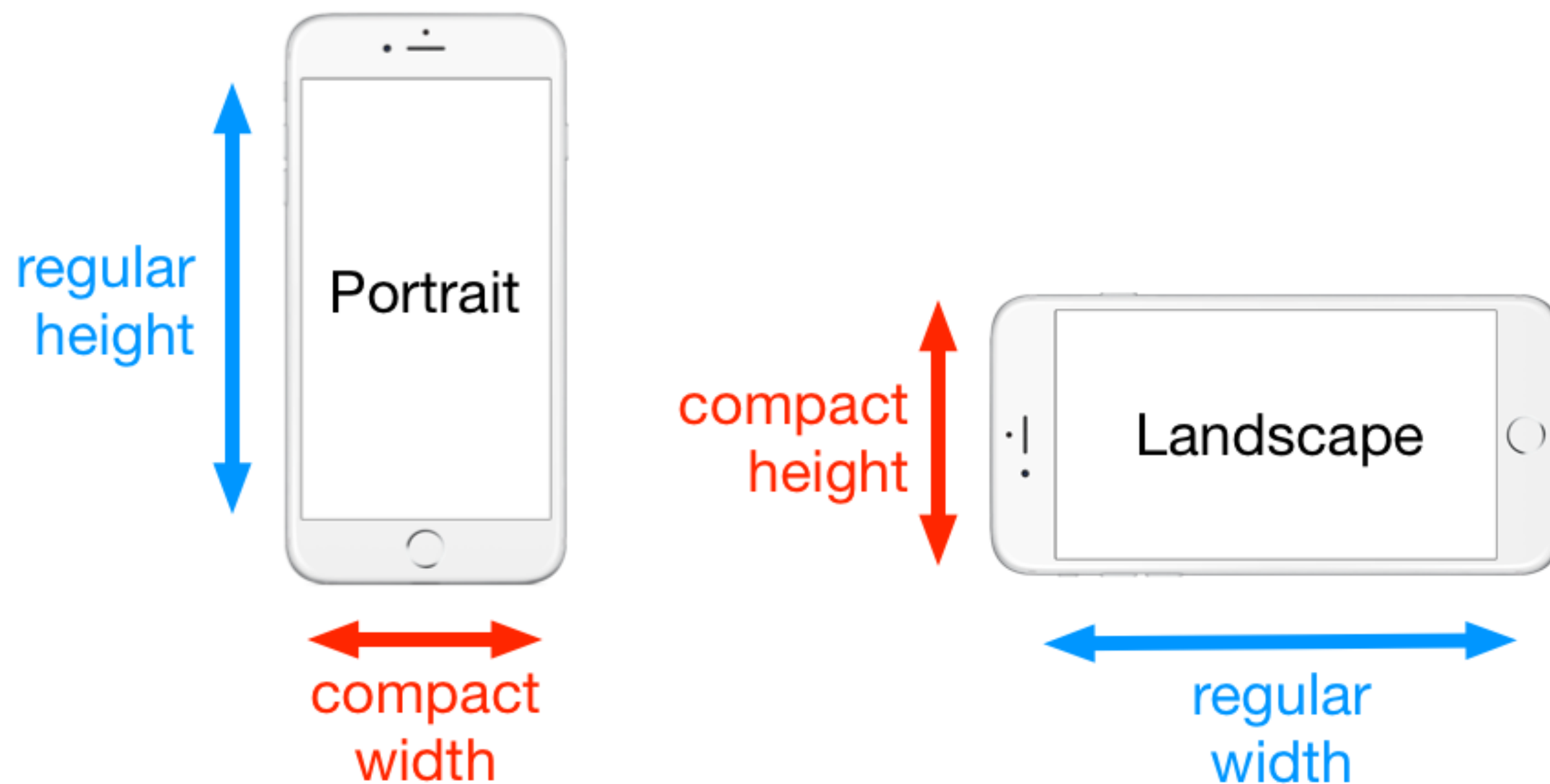
Be prepared for the
unspecified value.

Override the traits if you want a different layout than what is provided by the system.

All other iPhone models




iPhone Plus models



[Back](#) **Inbox** [Edit](#)

 **Peter Parker** *Wednesday*
Awesome Photos

 **Wade Winston Wilson** *7/31/17*
Check this out!

Tony Stark *7/27/17*
Let me know when you are free

Chyornaya Vdova *7/18/17*
I am having a blast

Danny Rand *7/12/17*
Hurt my hand :-{



Updated Just Now



From: **Peter Parker** >

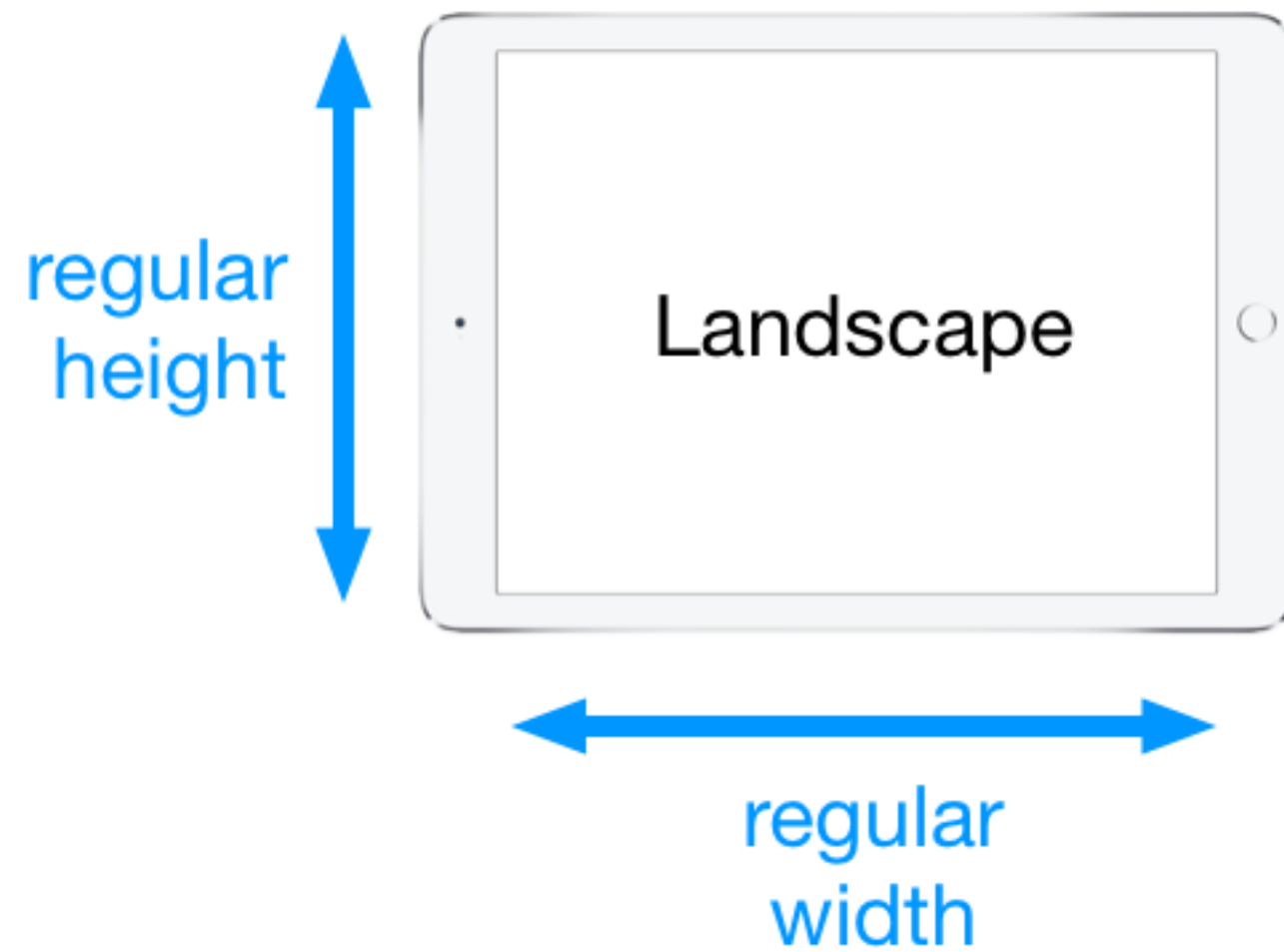
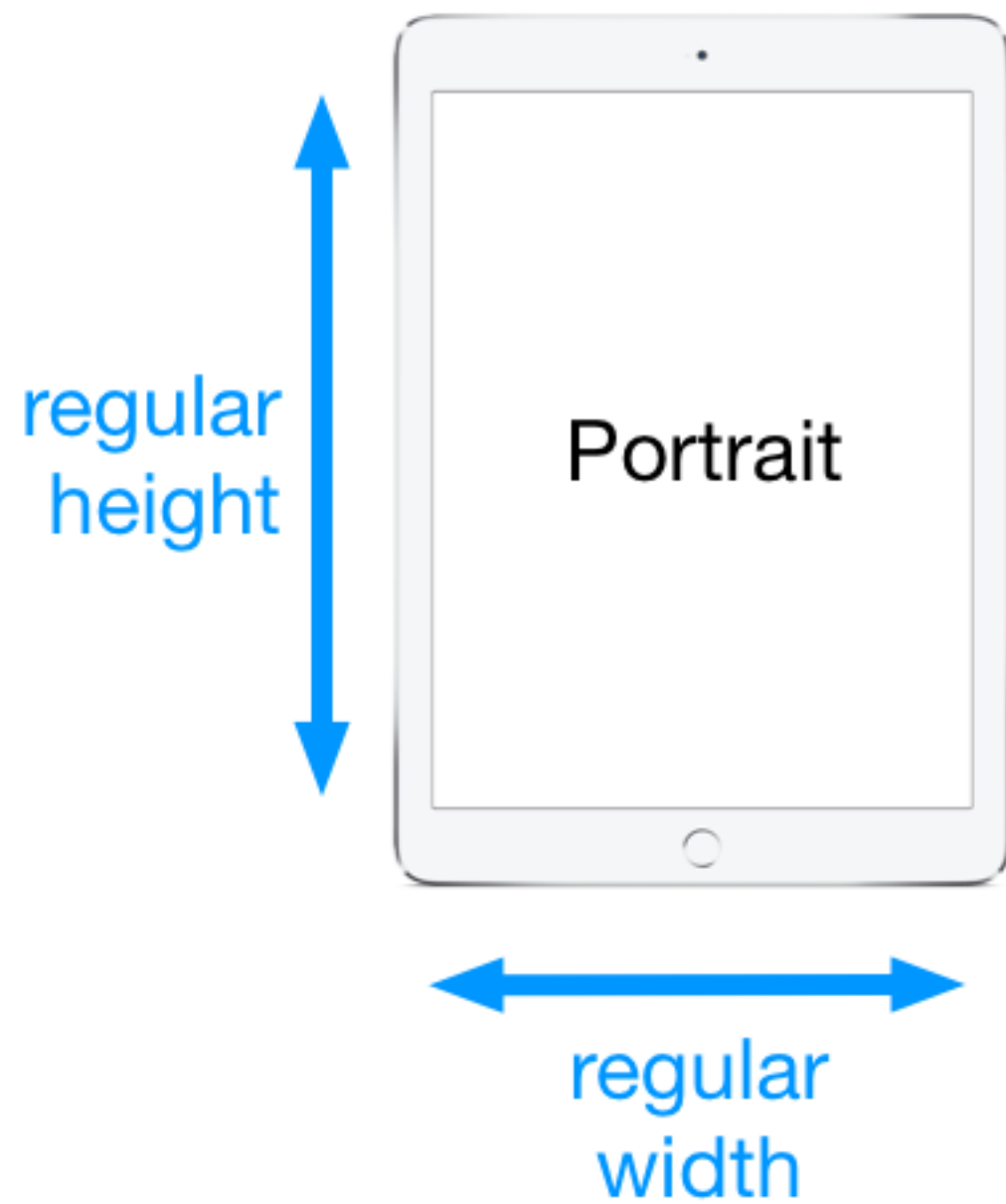
To: * **Kar** >

[Hide](#)

Awesome Photos
August 2, 2017 at 2:13 PM



iPad all models full screen

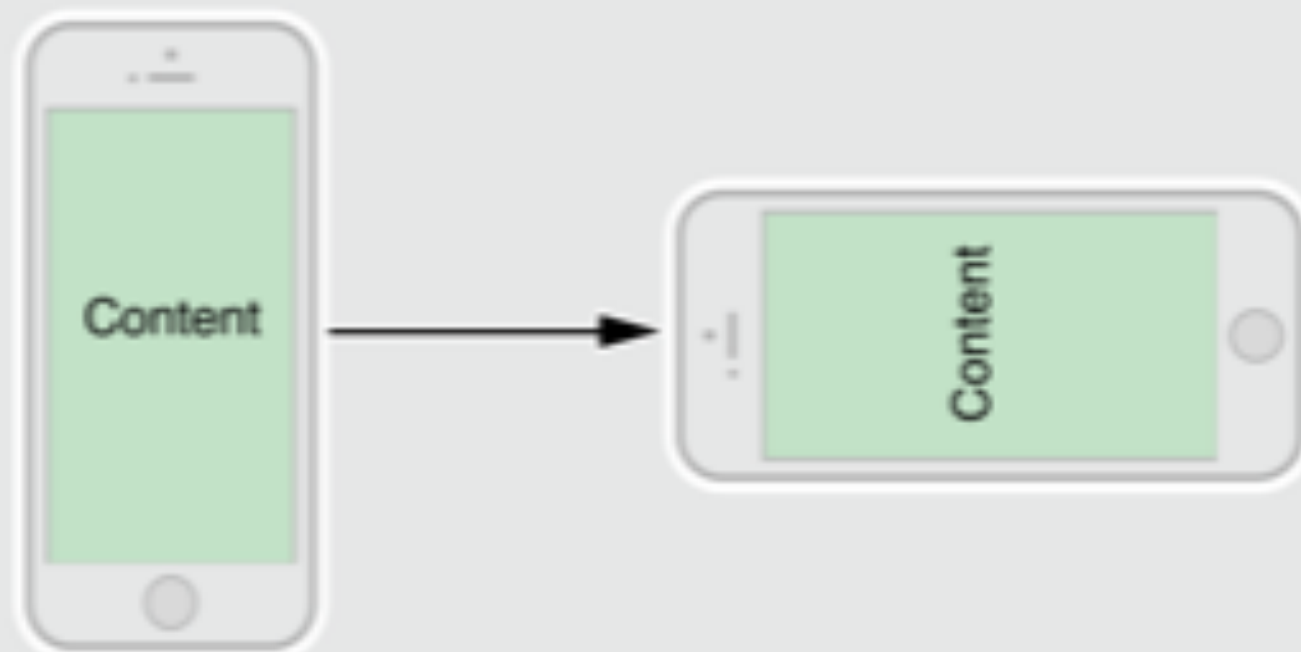


Adapting to change

Changes to trait collection can happen because:

- Was overridden by parent viewController**
- Rotation**
- Multitasking**

1) User rotates device



Current Size Classes

h: Compact

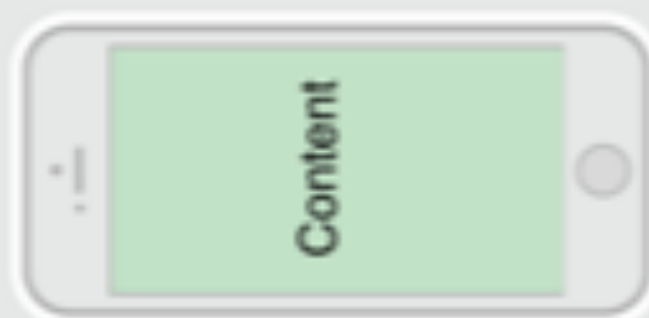
v: Regular

Current View Size

w: 320

h: 568

2) `willTransitionToTraitCollection:`
`withTransitionCoordinator:`



(Proposed Size
Class Change)
Vertical: Compact

Current Size Classes

h: Compact

v: Regular

Current View Size

w: 320

h: 568

3) `viewwillTransitionToSize:
withTransitionCoordinator`



(Proposed View
Size Change)

w: 568
h: 320

Current Size Classes
h: Compact
v: Regular

Current View Size
w: 320
h: 568

4) Animate Changes and
update layout



Current Size Classes
h: Compact
v: Compact

Current View Size
w: 568
h: 320

5) `traitCollectionDidChange:`



(Previous Size Class)
Vertical: Regular

Current Size Classes
h: Compact
v: Compact

Current View Size
w: 568
h: 320

Demo

ViewController Presentation

Navigation

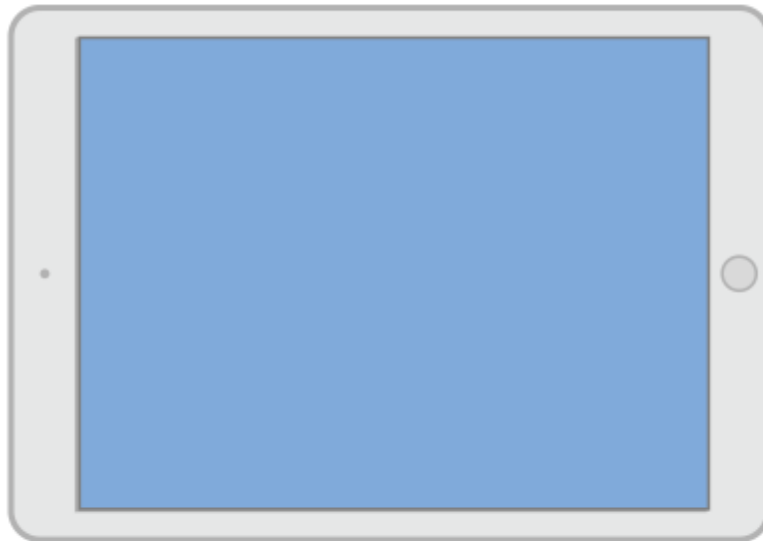
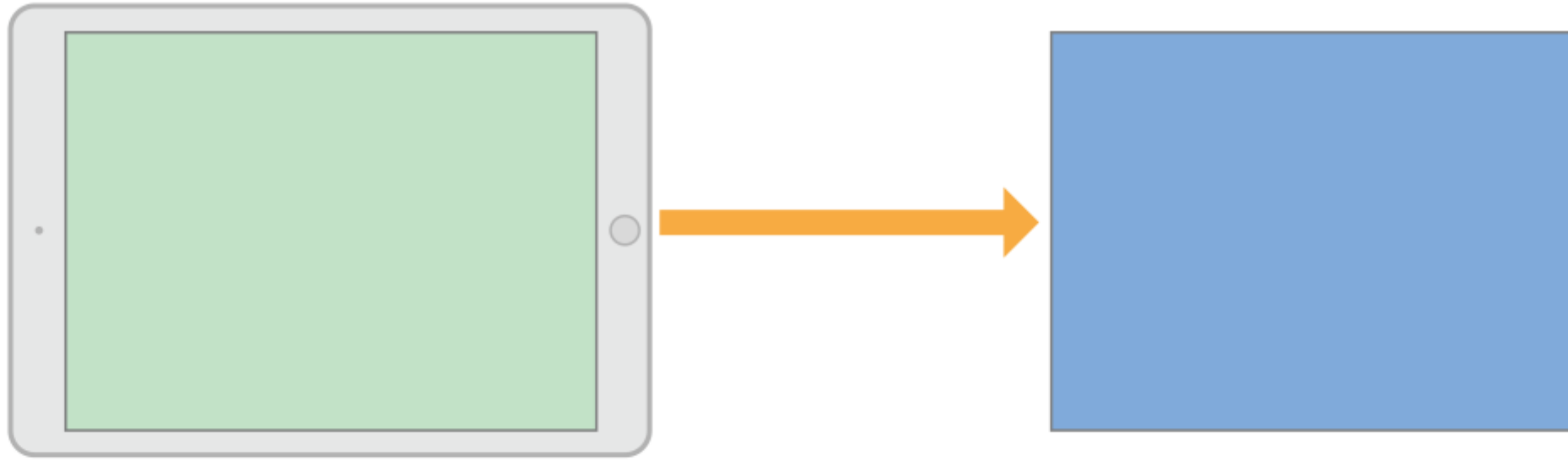
- **Don't explicitly use UINavigationController's methods.**¹
 - **Instead use** `showViewController:sender:`
`and showDetailViewController:sender:`
 - **These allow customization of the presentation for a given context overriding**
`targetViewControllerForAction:sender:`

¹ `targetViewControllerForAction:sender:` is smarter than it seems

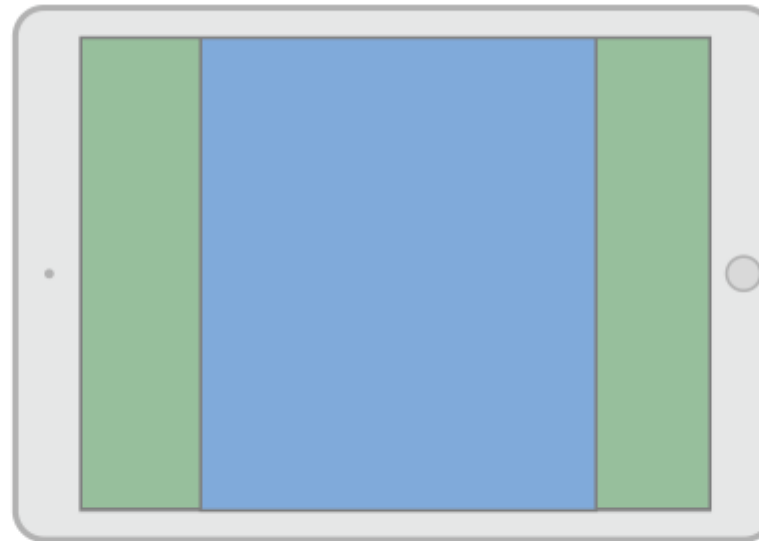
Navigation

- **Adopt `UISplitViewController` when appropriate.**
 - **Allows better integration when shown on `.regular` horizontal size classes.**

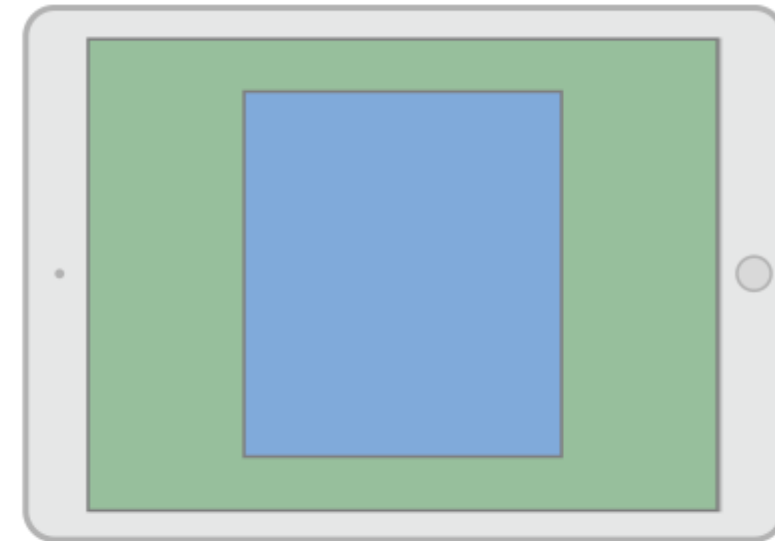
Presentation



UIModalPresentationFullscreen

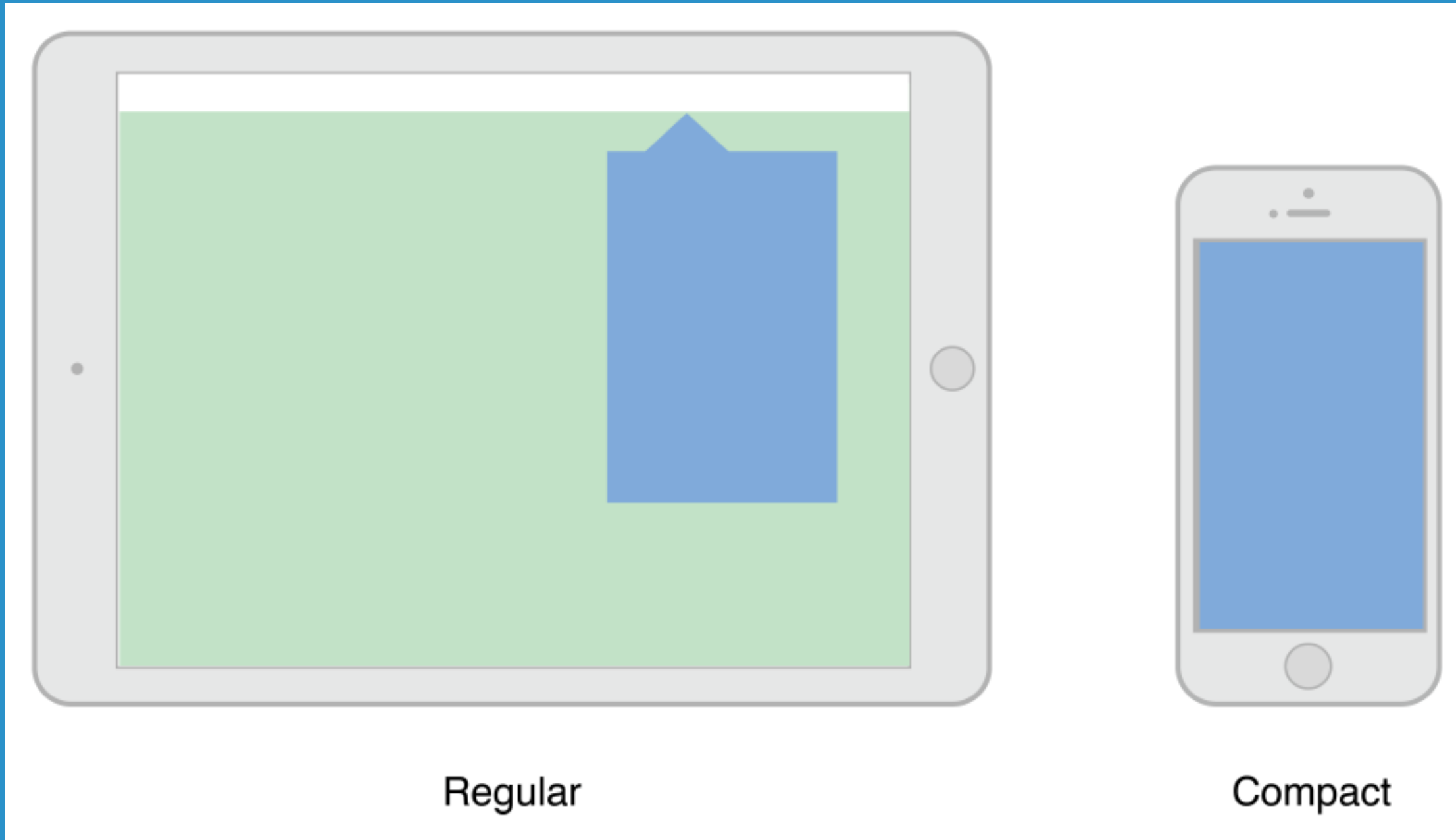


UIModalPresentationPageSheet



UIModalPresentationFormSheet

Popovers



Popovers

- **Use** `UIModalPresentationPopover`
- **Customize through**
`UIPopoverPresentationController`
 - **Origin of the popover**
 - `preferredContentSize`
 - **Margins**

Recommendations:

**Work closely with your
designer**

**Think in terms of
proportions, not pixels/
points**

**It's easier to test
transitions on device**