

# Tapmates.



# Mastering Auto Layout in Code

Robert Vojta

# Obsah

- Co je auto layout
- Proč ne Interface Builder
- Jak se definují vztahy
- Praktické ukázky

Co je  
auto layout?

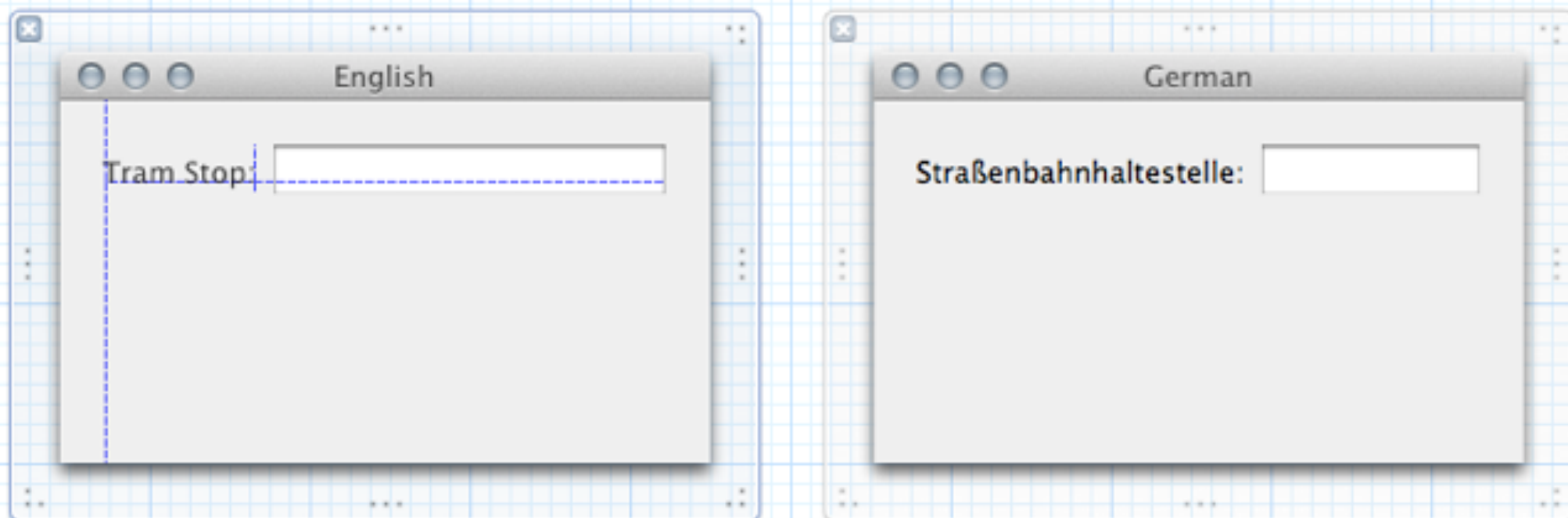
# Auto Layout

System pomocí kterého definujeme vztahy mezi jednotlivými prvky uživatelského rozhraní.

# Výhody?

Každý element má preferovanou velikost, kterou si určuje sám a auto layout nám to základě definovaných vztahů celé pěkně poskládá.

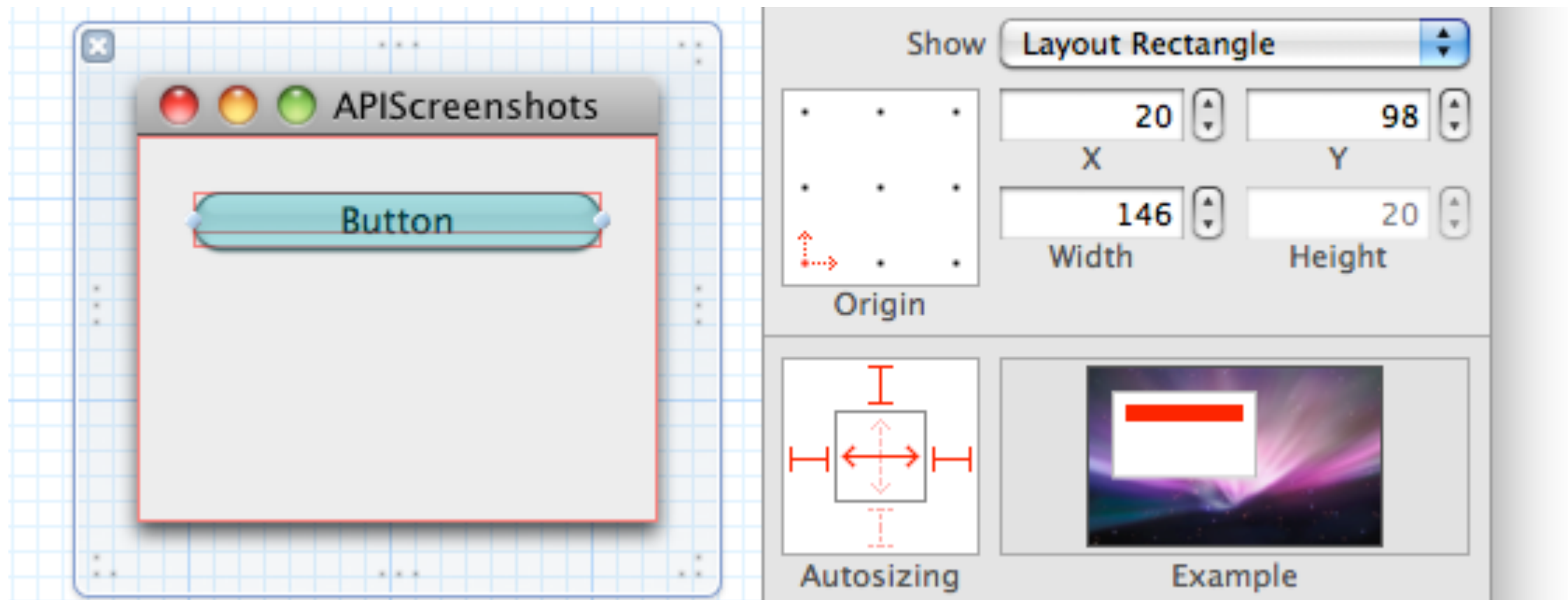
Bezproblémová lokalizace uživatelského rozhraní včetně RTL jazyků.



# Adaptace projektu

Může být postupná a není potřeba adaptovat celý projekt najednou.

Možnost kombinovat auto layout a starý známý způsob nazývaný “springs and struts”.



# Interface Builder?

Osobně ho nemám vůbec rád.

Dobře, nesnáším ho.












# Why?











- uklikáte se k smrti
- jak se má člověk trefit na 1px tlustou čáru
- designer posune element a vztahy jsou v ...
- nepřehledný seznam
- vybrané vztahy nejdou smazat

# Klikáme

## Align

-  Left Edges
-  Right Edges
-  Top Edges
-  Bottom Edges
-  Horizontal Centers
-  Vertical Centers
-  Baselines
-  Horizontal Center in Container
-  Vertical Center in Container

## Pin

-  Width
-  Height
-  Horizontal Spacing
-  Vertical Spacing
-  Leading Space to Superview
-  Trailing Space to Superview
-  Top Space to Superview
-  Bottom Space to Superview
-  Widths Equally
-  Heights Equally

When Resizing Views Apply Constraints to...

- Siblings and Ancestors
- ✓ Descendants

☒ Start Moment at login

☒ Sound effects

Global shortcut

Custom SRRecorderControl

Set a shortcut that brings Moment in front of all other windows.

Custom View

☒ Make posted photos look better (auto enhance)

☒ Reduce size of posted photos for faster upload

☒ Copy posted moment link to clipboard

Log Out

Check for Updates

Quit



# WTF?!?

Bezva přehled,  
že?

- ▼ Constraints
  - Vertical Space – Check Box – General
  - Horizontal Space – Check Box – General
  - Vertical Space (14) – Check Box – Check Box
  - Horizontal Space – Check Box – General
  - Vertical Space – Image View – Check Box
  - Leading Alignment – Image View – Static Text – Set a s...
  - Trailing Alignment – Image View – Image View
  - Vertical Space (225) – General – Custom View
  - Horizontal Space (64) – Custom View – General
  - Vertical Space (8) – Custom View – Image View
  - Center X Alignment – Custom View – Image View
  - Leading Alignment – Static Text – Set a shortcut that b...
  - Horizontal Space – General – Static Text – Set a shortc...
  - Leading Alignment – Image View – Image View
  - Vertical Space – Image View – Static Text – Set a shortc...
  - Center X Alignment – Image View – Check Box
  - Trailing Alignment – Image View – Image View
  - Horizontal Space – Check Box – General
  - Horizontal Space – General – Check Box
  - Vertical Space – Check Box – Image View
  - Horizontal Space – Check Box – General
  - Horizontal Space – General – Check Box
  - Vertical Space (14) – Check Box – Check Box
  - Horizontal Space – General – Check Box
  - Horizontal Space – Check Box – General
  - Vertical Space (14) – Check Box – Check Box
  - Vertical Space (16) – Image View – Check Box

**Jak se definují  
vztahy?**

$$y = m * x + b$$

- **y** a **x** jsou atributy
- atribut může být left, right, top, bottom, leading, trailing, width, height, centerX, centerY, baseline
- LTR - leading = left, trailing = right
- RTL - leading = right, trailing = left
- **m** a **b** jsou konstanty

# NSLayoutConstraint

```
[NSLayoutConstraint constraintWithItem:button1
                                attribute:NSLayoutAttributeRight
                                relatedBy:NSLayoutRelationEqual
                                toItem:button2
                                attribute:NSLayoutAttributeLeft
                                multiplier:1.0
                                constant:-12.0];
```

**y = m \* x + b**

y = attribute (1st - item)

m = multiplier

x = attribute (2nd - relatedBy)

b = constant

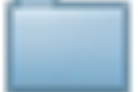





# Vizuální jazyk

```
NSDictionary *viewsDictionary =  
NSDictionaryOfVariableBindings(button1, button2);  
  
NSArray *constraints = [NSLayoutConstraint  
    constraintsWithVisualFormat:@"[button1]-[button2]"  
        options:0  
        metrics:nil  
        views:viewsDictionary];
```

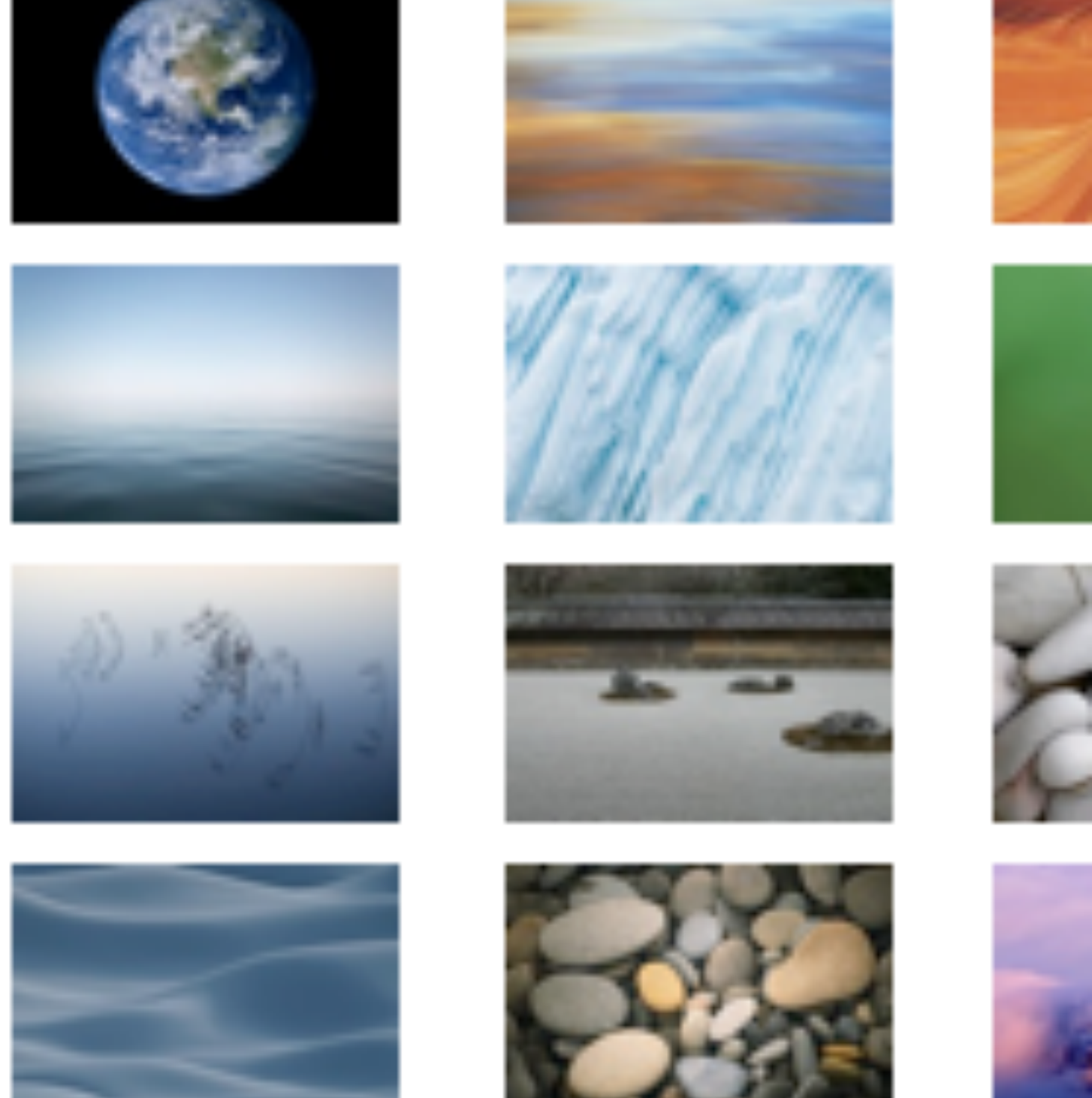


# Další možnosti

- vztah může být definován jako nerovnost  
šířka  $\geq 200$  a šířka  $\leq 400$
- vztah může mít prioritu
- vztah mohou mít objekty na různé úrovni celé hierarchie
- element se může bránit zmenšování či zvětšování
- až na minimum věci immutable

-  Plants
-  Art
-  Black & White
-  Abstract
-  Patterns
-  Solid Colors

iPhoto  
Folders



- ☐ Change picture: Every 30 m
- ☐ Random order
- ☐ Translucent menu bar

# fittingSize

Minimální velikost view při které je možné dodržet všechny definované vztahy (constraints).

## iOS

- `(CGSize)systemLayoutSizeFittingSize:(CGSize)targetSize`

## Mac OS X

- `(NSSize)fittingSize`

# intrinsicContentSize

Přirozená velikost elementu bez ohledu na okolí - na to kde je umístěn a kolik je pro něj místa.

## iOS

```
-(CGSize)intrinsicContentSize  
UIViewNoIntrinsicMetric
```

## Mac OS X

```
-(NSSize)intrinsicContentSize  
NSViewNoIntrinsicMetric
```

# Element se brání

## iOS

`contentHuggingPriorityForAxis:`  
`contentCompressionResistancePriorityForAxis:`

## Mac OS X

`contentHuggingPriorityForOrientation:`  
`contentCompressionResistancePriorityForOrientation:`

# Priorita - iOS

```
enum {  
    UILayoutPriorityRequired = 1000,  
    UILayoutPriorityDefaultHigh = 750,  
    UILayoutPriorityDefaultLow = 250,  
    UILayoutPriorityFittingSizeLevel = 50,  
};  
typedef float UILayoutPriority;
```

# Priorita - Mac

```
enum {  
    NSLayoutPriorityRequired = 1000,  
    NSLayoutPriorityDefaultHigh = 750,  
    NSLayoutPriorityDragThatCanResizeWindow = 510,  
    NSLayoutPriorityWindowSizeStayPut = 500,  
    NSLayoutPriorityDragThatCannotResizeWindow = 490,  
    NSLayoutPriorityDefaultLow = 250,  
    NSLayoutPriorityFittingSizeCompression = 50,  
};  
typedef float NSLayoutPriority;
```

# Praktické ukázky

