

# AsyncDisplayKit

Магия

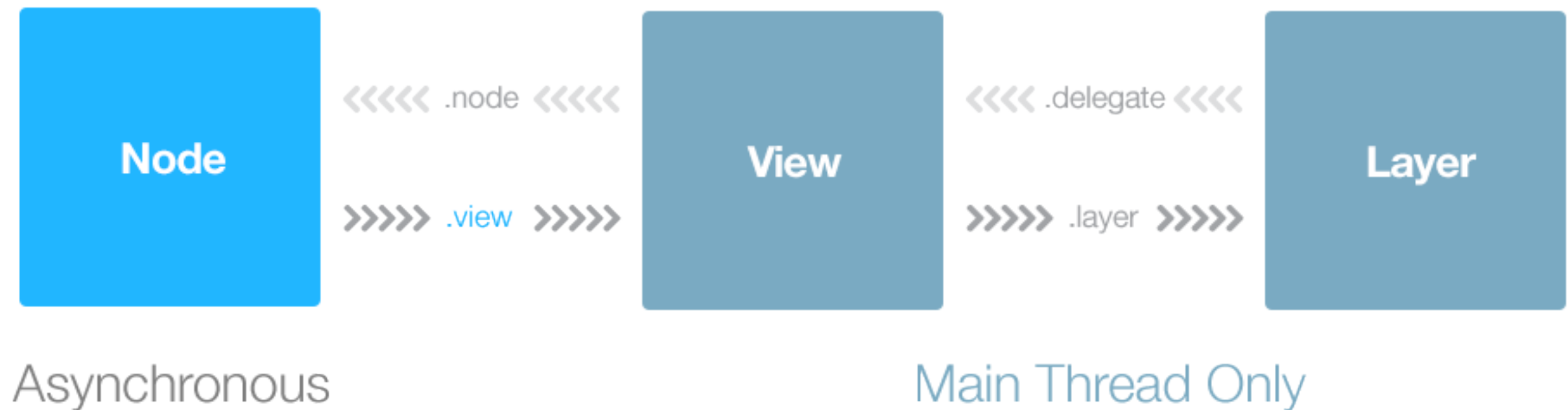
AnjLab **TechTalks** 2016 г. Владимир

Королев Юрий

# Немного истории

- Разработан в Facebook для Paper
- Используется в Facebook messenger
- Pinterest

# Основная идея



# Интерфейсы схожи с UIKit

UIView	ASDisplayKit
UIImageView	ASImageNode, ASNetworkImageNode
UILabel	ASTextNode
UITableViewCell, UICollectionViewCell	ASCellNode
UITableView	ASTableView/Node
UICollectionView	ASCollectionView/Node
UIButton	ASButtonNode

# ASTextNode

порт ComponentTextKit

# ASTextNode

- ТОЛЬКО **attributedString**
- МОЖНО указывать **truncationAttributedString**,  
**additionalTruncationMessage**,  
**textNodeTappedTruncationToken**
- **maximumNumberOfLines** = 0
- поддержка тапов по ссылкам,  
**passthroughNonLinkTouches**, подсветка
- **placeholders**

# ASImageNode

- placeholderColor
- эффективный кроп
- imageModificationBlock:
  - ASImageNodeRoundBorderModificationBlock
    - (borderWidth, borderColor)
  - ASImageNodeTintColorModificationBlock
    - (color)

# ASNetworkImageNode

- NSURL
- подключаемый загрузчик  
(ASImageDownloaderProtocol)
- подключаемый кеш (ASImageCacheProtocol)
- есть pod WebASDKImageManager



# UIButtonNode

- `laysOutHorizontally :`)

ASTableView/Node

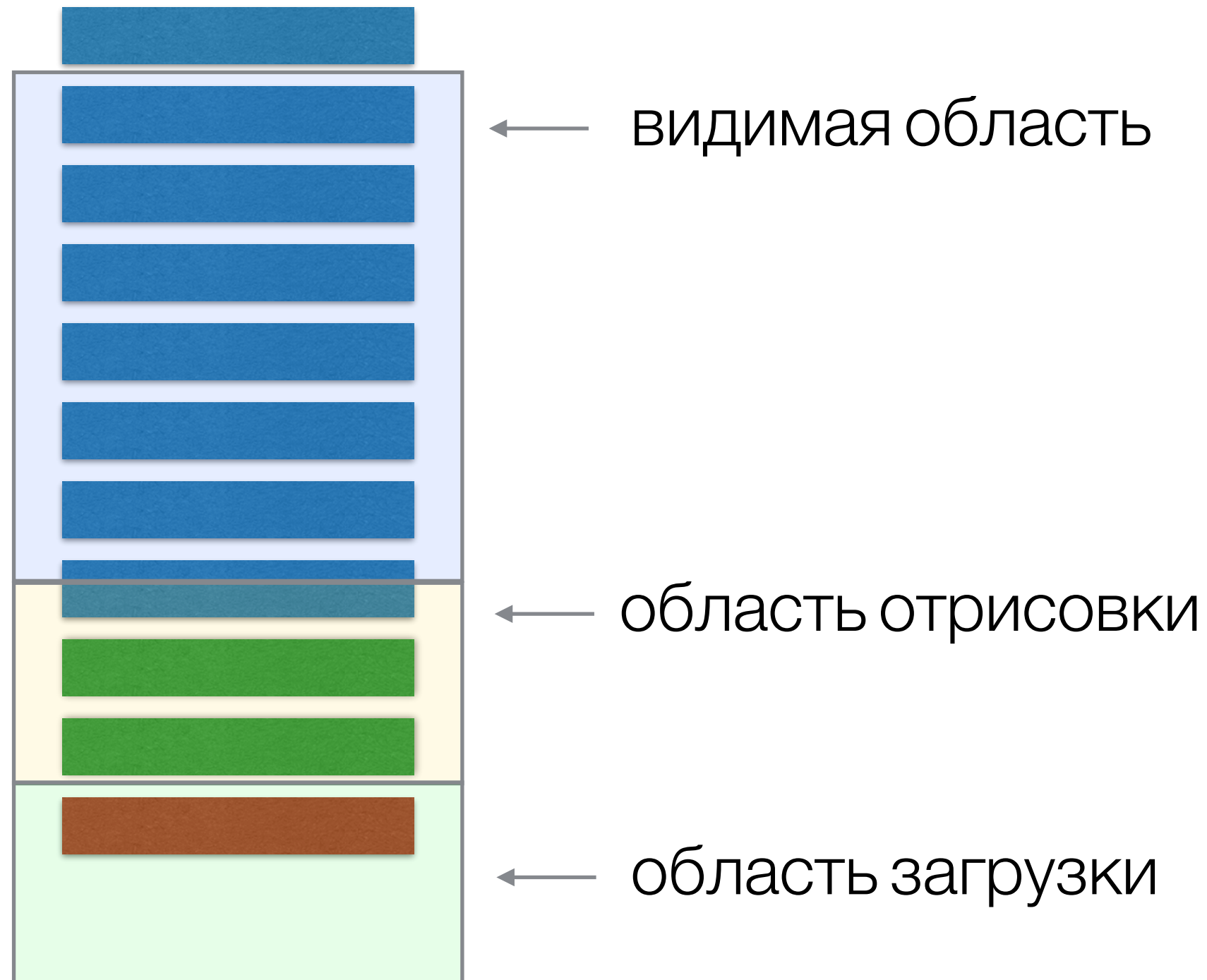
# ASTableView/Node

- ASTableViewDelegate - расширенный UITableViewDelegate
- ASTableViewDataSource - расширенный UITableViewDataSource

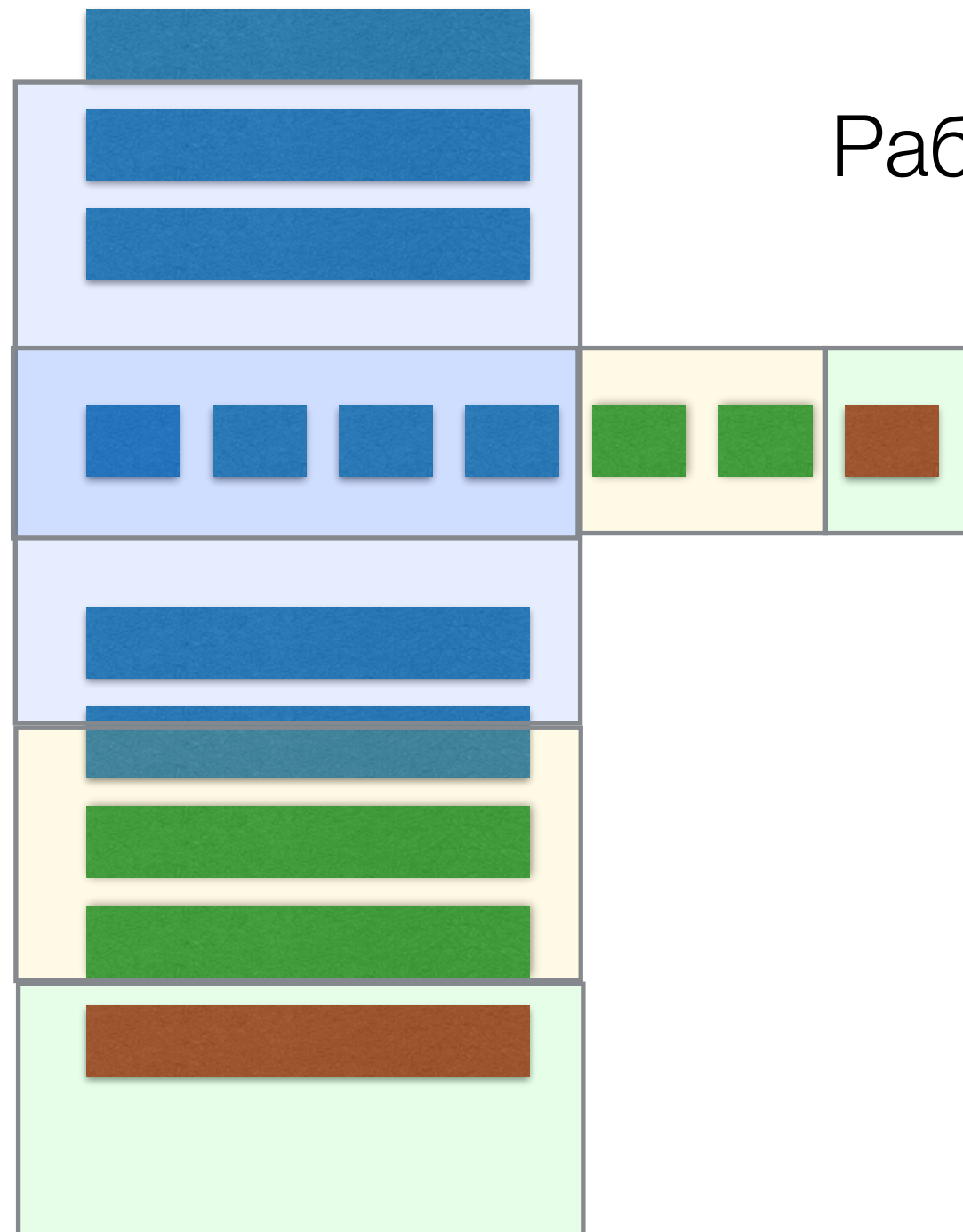
# ASTableView/Node

- ASCellNode создаются все заранее (нет проблем с реюзом)
- Поддержка подзагрузки с компенсацией скролла

# ASTableView/Node



# ASTableView/Node



Работает в 2d

# ASTableViewDataSource

- Пакетная загрузка (Batch fetch)
  - shouldBatchFetchForTableView
  - tableView(tableView:willBeginBatchFetchWithContext:)
  - ASBatchContext -  
completeBatchFetching(didComplete: Bool)

# ASLayoutSpec

порт ComponentKit/LayoutComponents



# ASLayoutSpec

- ASLayoutSpec
- ASInsetLayoutSpec
- ASRatioLayoutSpec
- ASCenterLayoutSpec
- ASOverlayLayoutSpec
- ASBackgroundLayoutSpec
- ASStaticLayoutSpec
- ASStackLayoutSpec

# ASLayoutable

- spacingBefore
- spacingAfter
- flexGrow
- flexShrink
- flexBasis
- alignSelf
- ascender
- descender
- sizeRange

# ASInsetLayoutSpec

`init(insets: UIEdgeInsets, child: ASLayoutable)`



# ASRatioLayoutSpec

init(ratio: CGFloat, child: ASLayoutable)

ratio - height / width

ratio: 2.0



ratio: 0.5



# ASCenterLayoutSpec

```
init(  
    centeringOptions: ASCenterLayoutSpecCenteringOptions,  
    sizingOptions: ASCenterLayoutSpecSizingOptions,  
    child: ASLayoutable  
)
```



# ASOverlayLayoutSpec

```
init(child: ASLayoutable, overlay: ASLayoutable?)
```



# ASBackgroundLayoutSpec

```
init(child: ASLayoutable, background: ASLayoutable?)
```



# ASStaticLayoutSpec

```
init(children: [ASStaticLayoutable])
```

- ASStaticLayoutable:
  - sizeRange: ASRelativeSizeRange
  - layoutPosition: CGPoint



# ASStackLayoutSpec

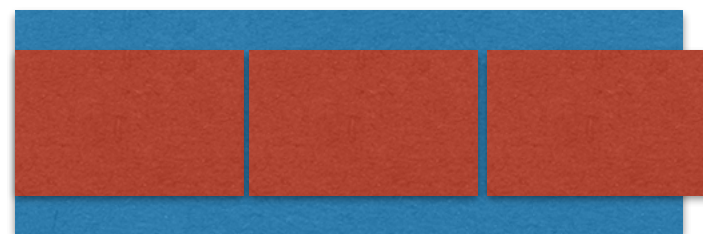
похож на UIStackView

# ASStackLayoutSpec

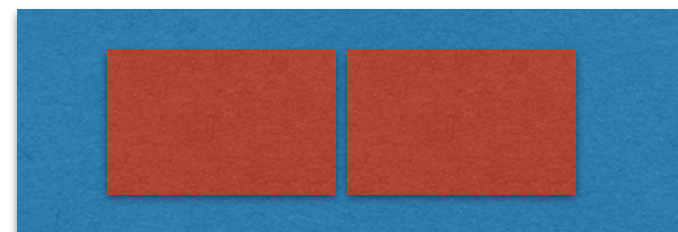
```
init(  
    direction: ASStackLayoutDirection,  
    spacing: CGFloat,  
    justifyContent: ASStackLayoutJustifyContent,  
    alignItems: ASStackLayoutAlignItems,  
    children: [ASLayoutable])
```

# ASStackLayoutSpec

## ASStackLayoutJustifyContent



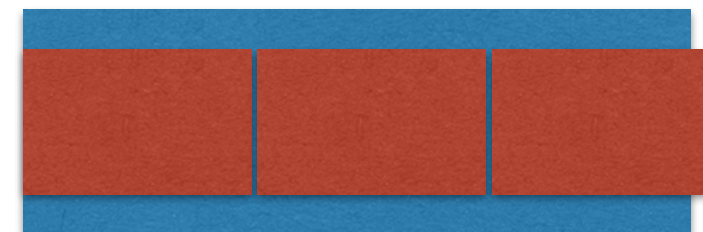
.Start



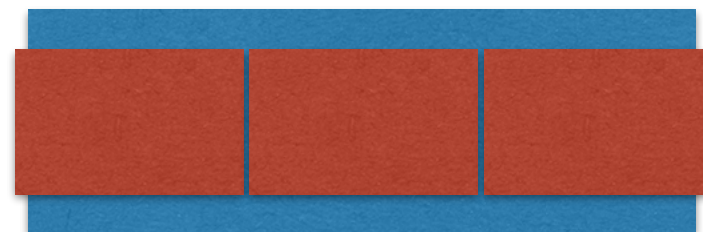
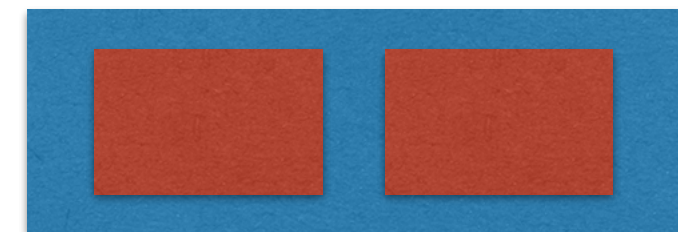
.Center



.End



.SpaceBetween



.SpaceAround

# ASStackLayoutSpec

`ASStackLayoutAlignItems`

- `.Start`
- `.End`
- `.Center`
- `.Stretch`
- `.BaselineFirst` (Horizontal stack only)
- `.BaselineLast` (Horizontal stack only)

Demo

Вопросы?