

РАЗРАБОТКА ПРОИЗВОДИТЕЛЬНОГО ЧАТА. ОПЫТ Badoo

Всем привет, меня зовут

Антон Щукин

github.com/badoo/Chatto
techblog.badoo.com

github.com/AntonPalich
@AntonPalich



MagicLab[★]

 badoo

 bumble

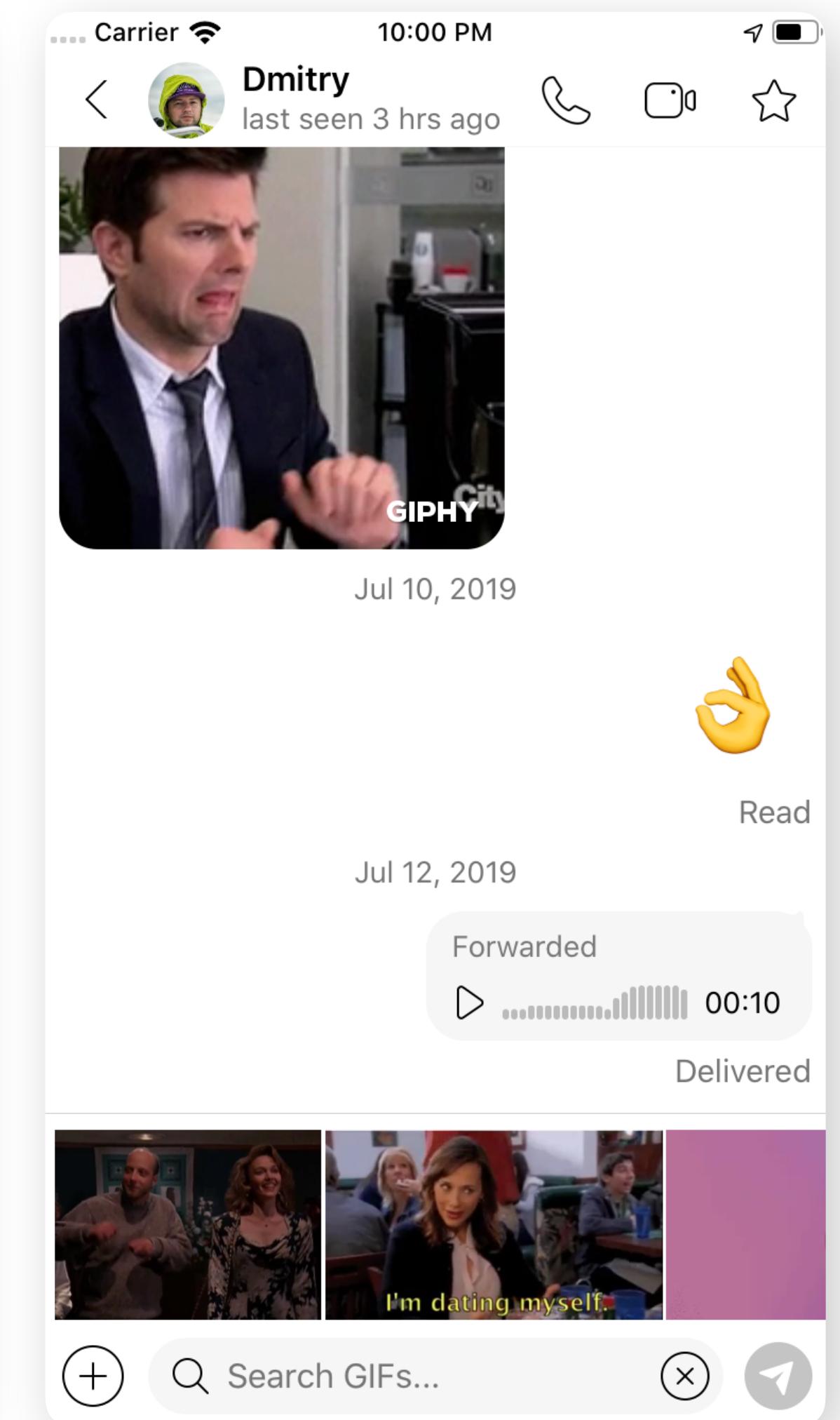
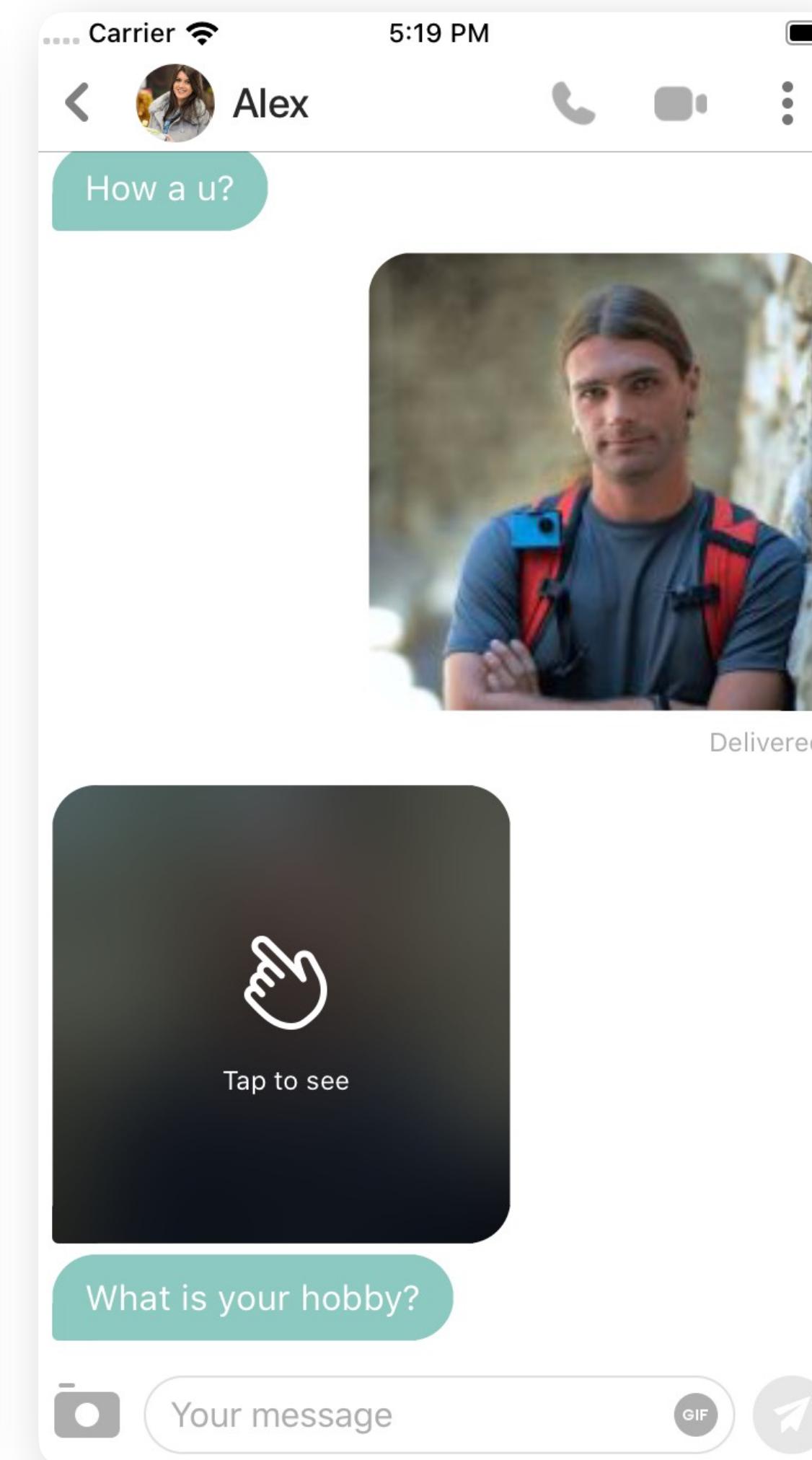
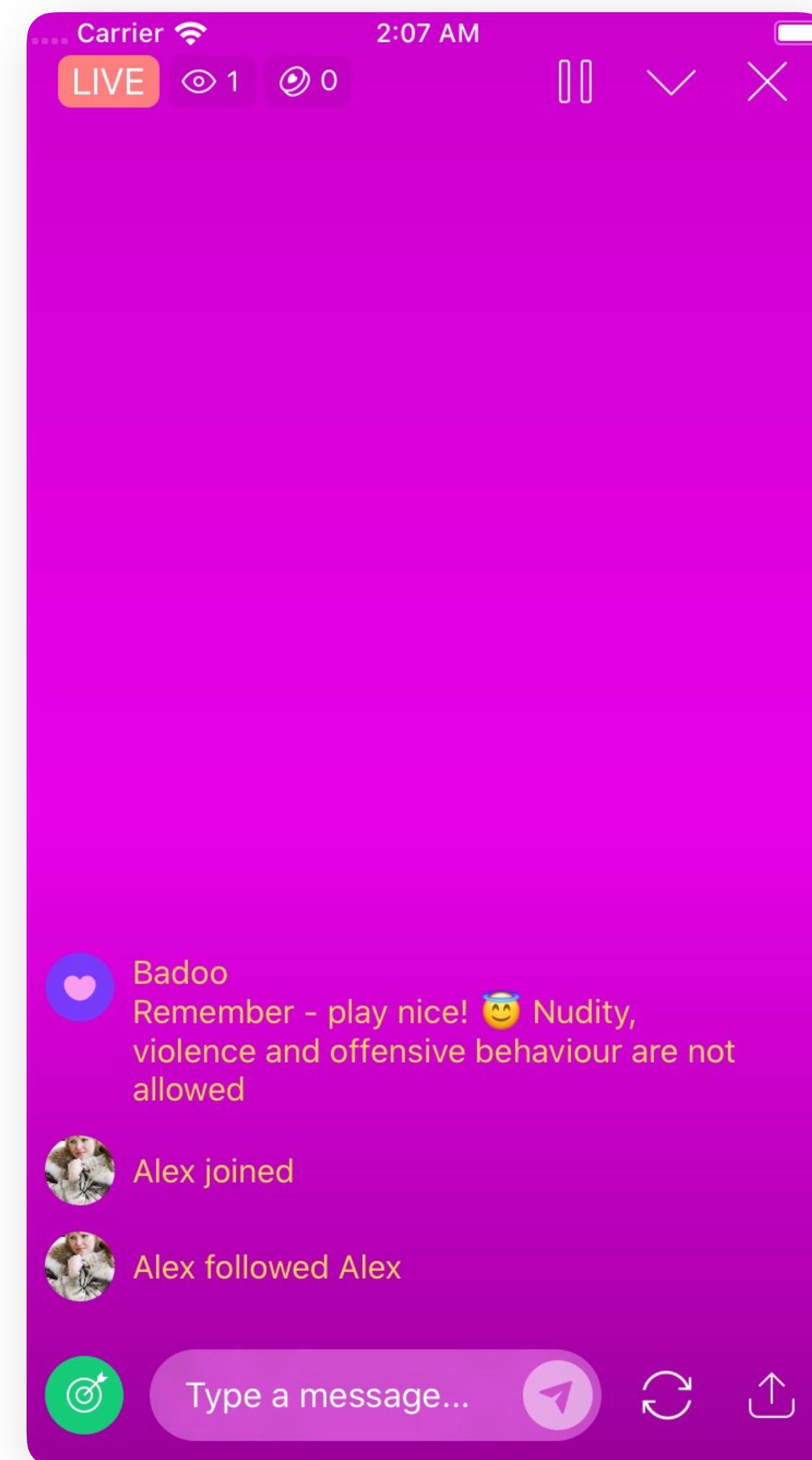
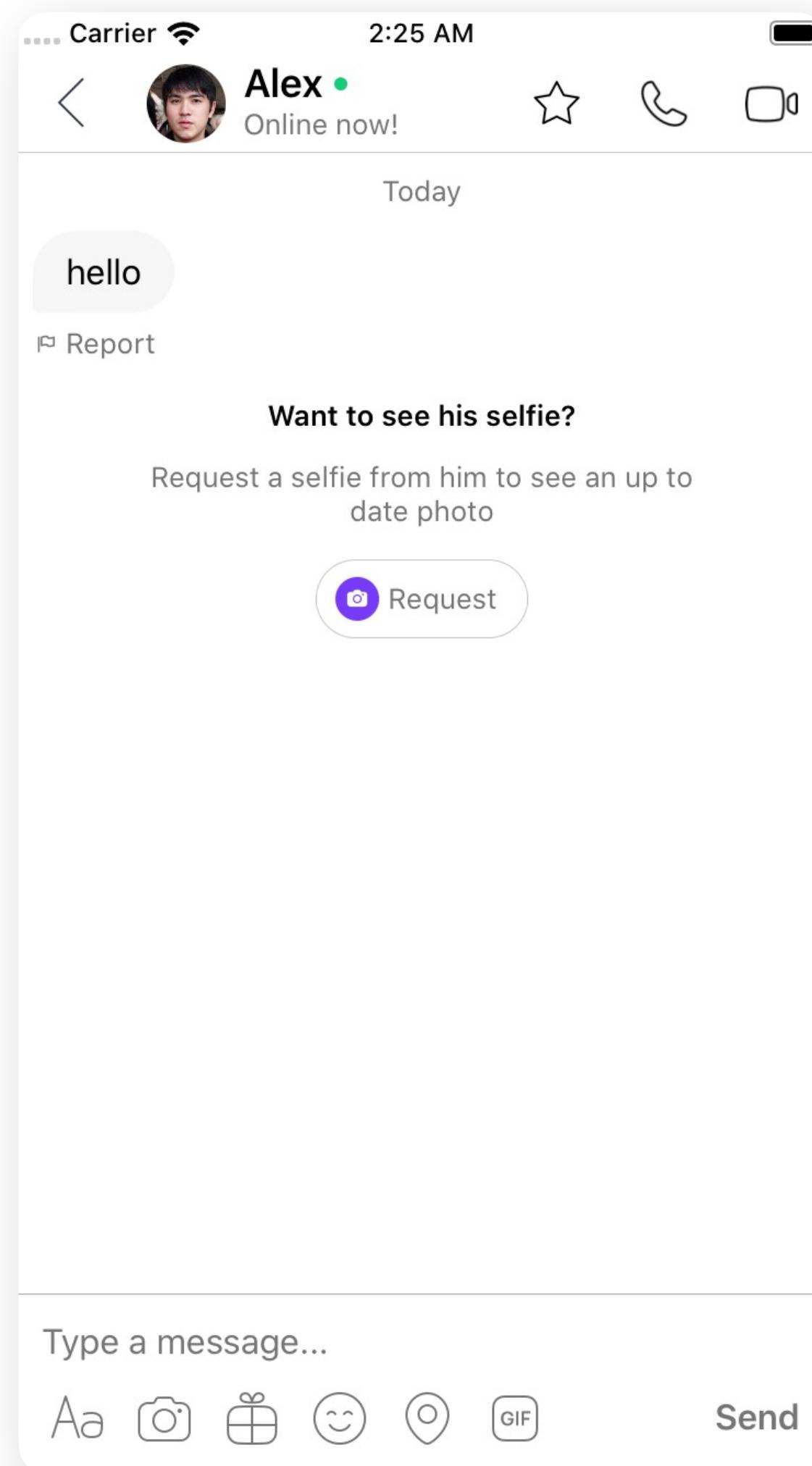
 Lumen

 CHAPPY



>550 000 000

people all over the world use our apps



Badoo Chat

Badoo Livestream

Bumble Chat

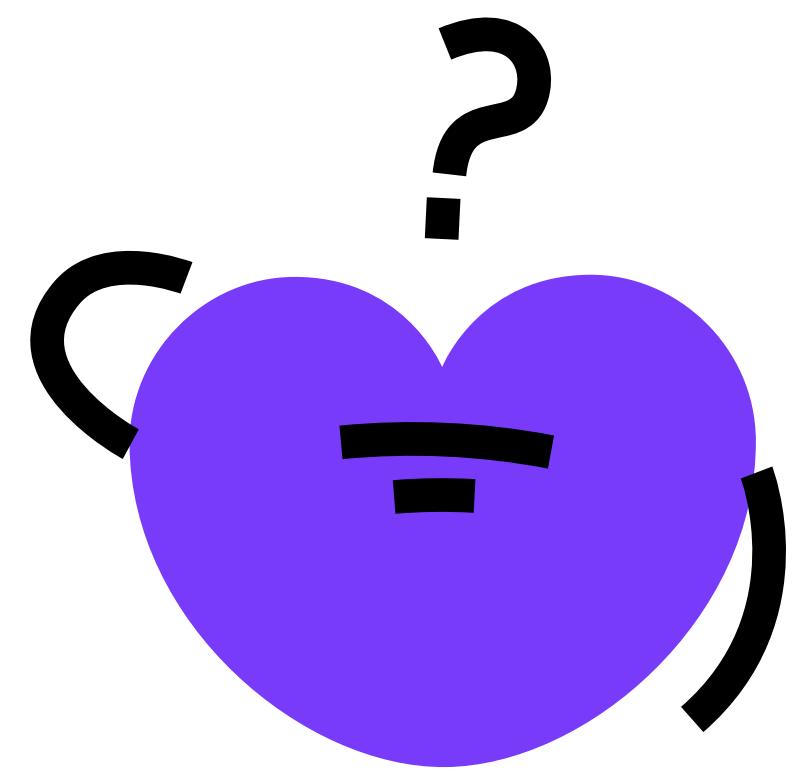
<Redacted> Chat

1. Выбор UI движка
2. Chatto
3. UICollectionView
4. Оптимизация UICollectionView
5. Оптимизация UI для сообщений

Выбор UI движка

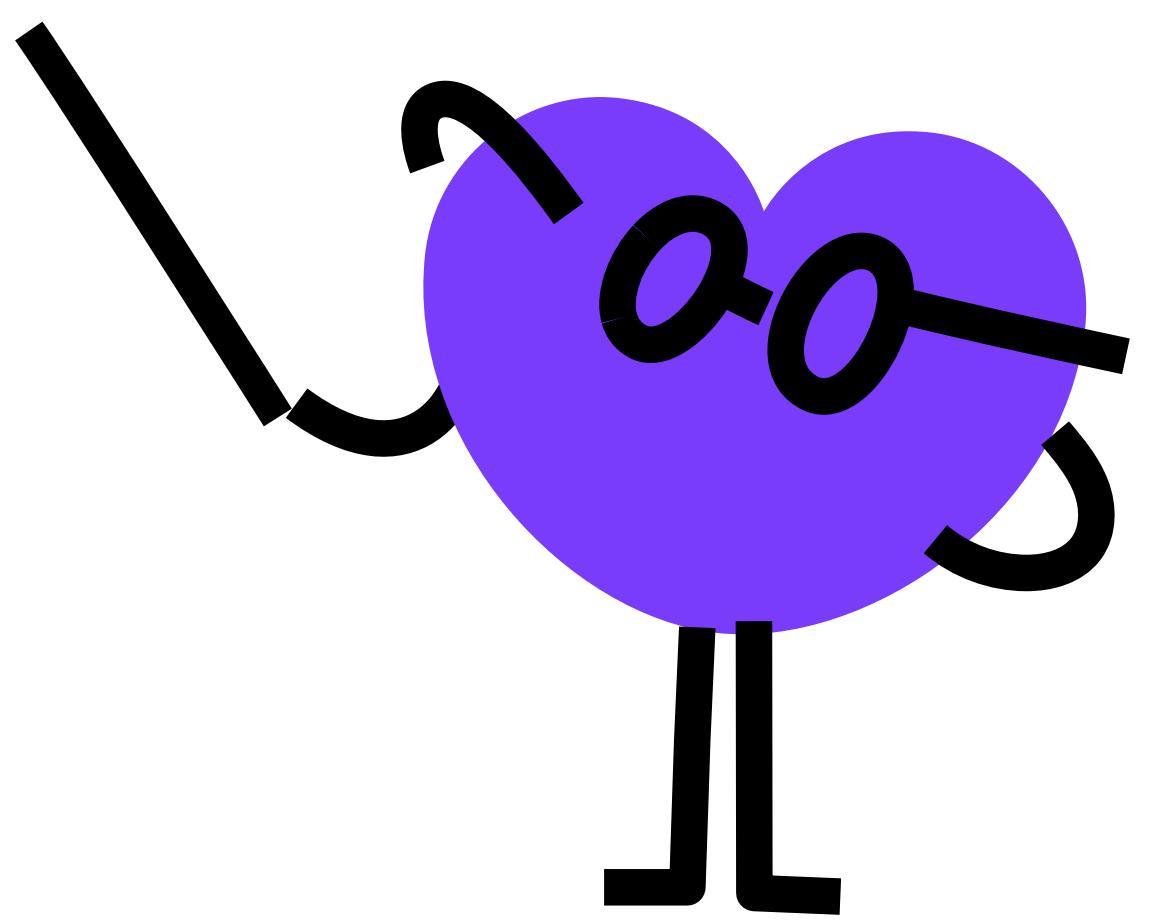
Почему с нуля?

- Технический долг
 - Шел 2015 год
 - Massive View Controller
 - Не расширяемая архитектура
 - Objective-C



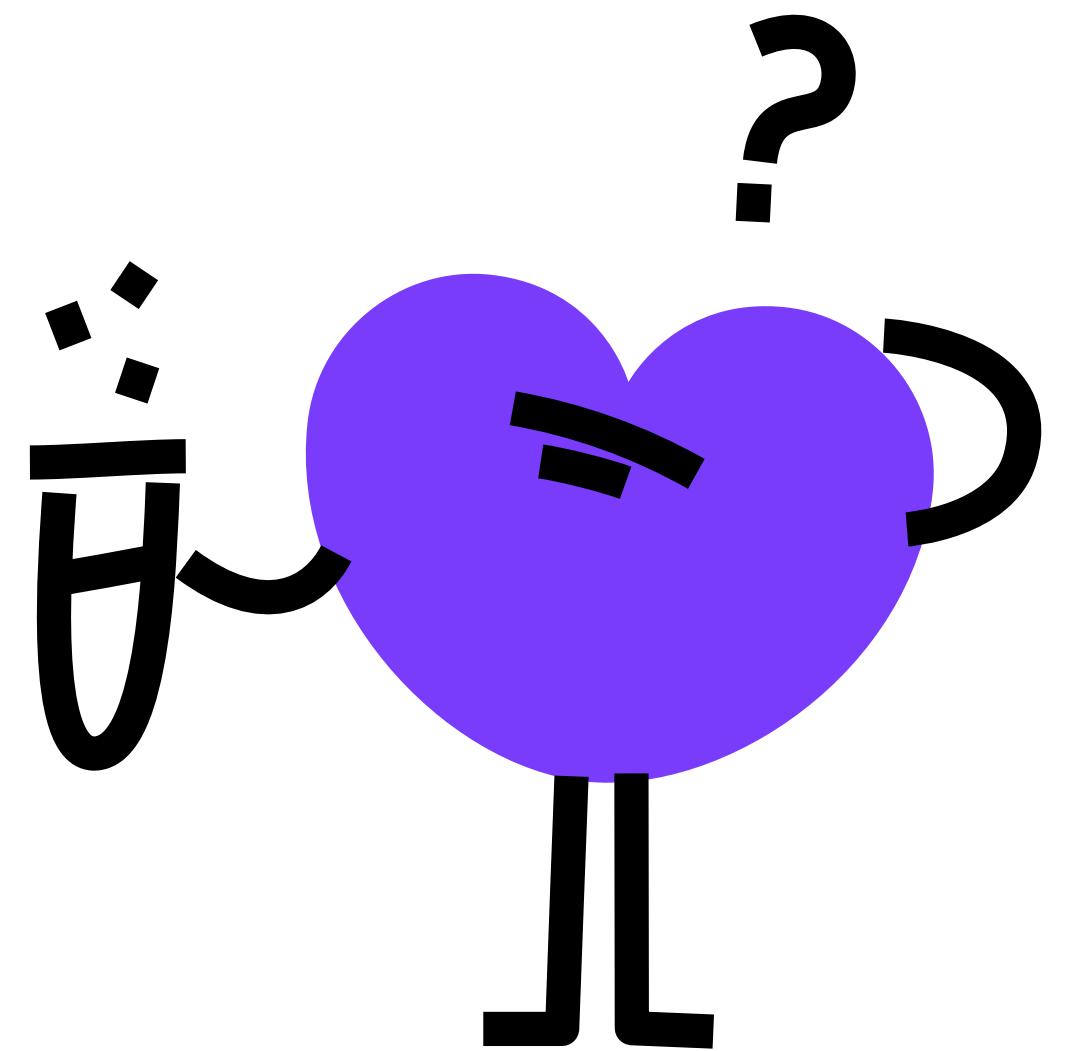
Основные требования

- Swift (2.0)
- Расширяемость
- Производительность (60 FPS - iPhone 4S)



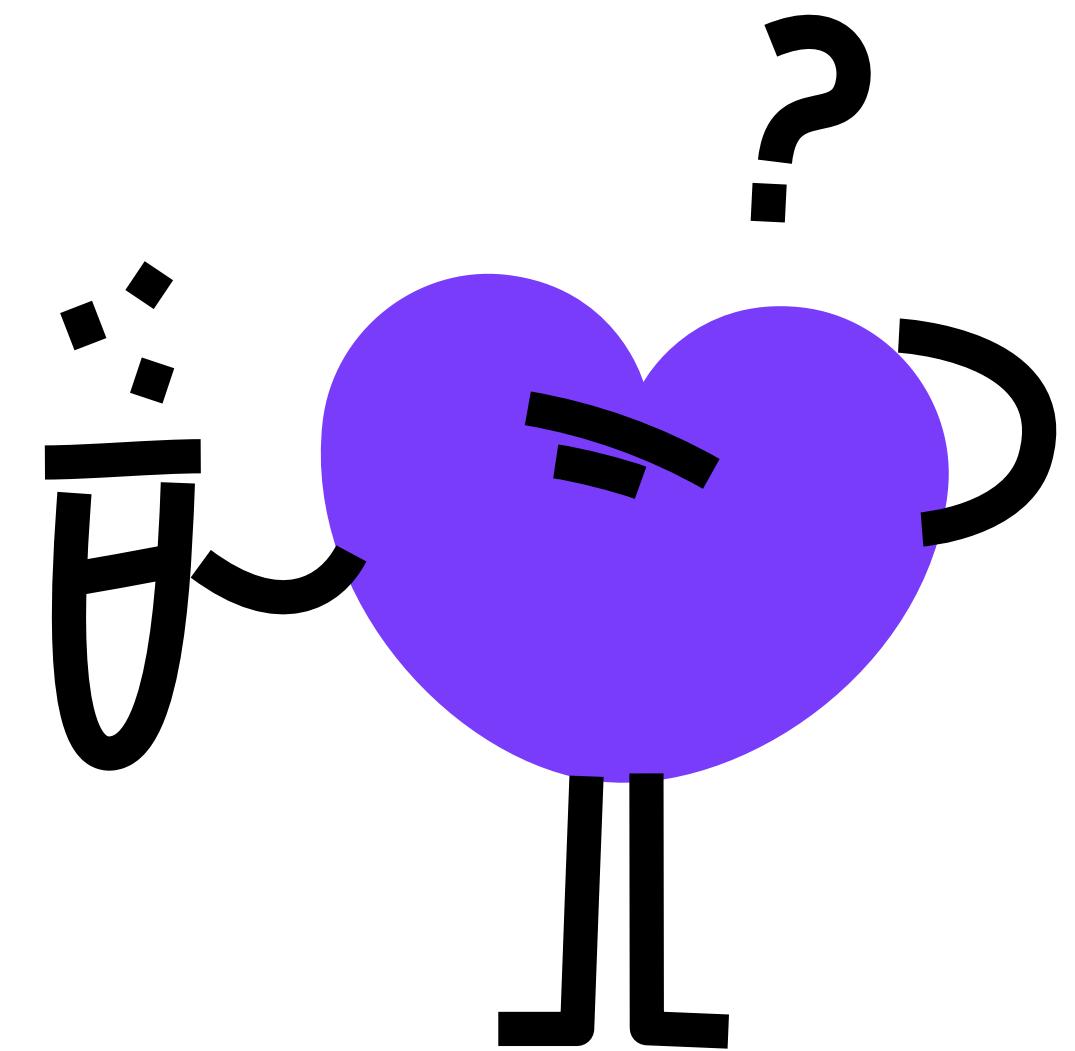
Варианты

- 3rd Party
 - AsyncDisplayKit, IGListKit и т.д.



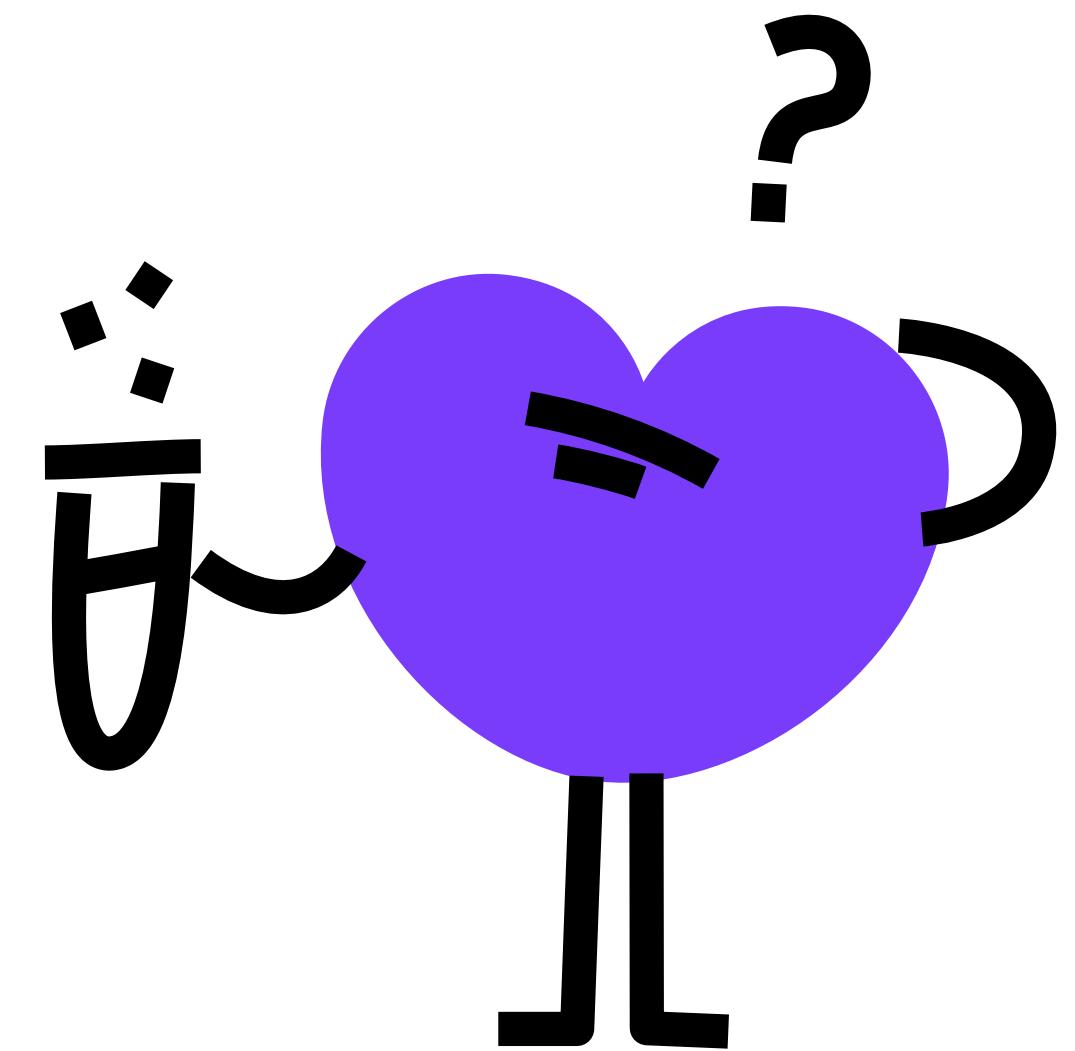
Варианты

- ~~3rd Party~~
 - ~~AsyncDisplayKit, IGListKit и т.д.~~
- **In-house**
 - **UITableView**
 - **UICollectionView**
 - **Своя коллекция на основе UIScrollView**



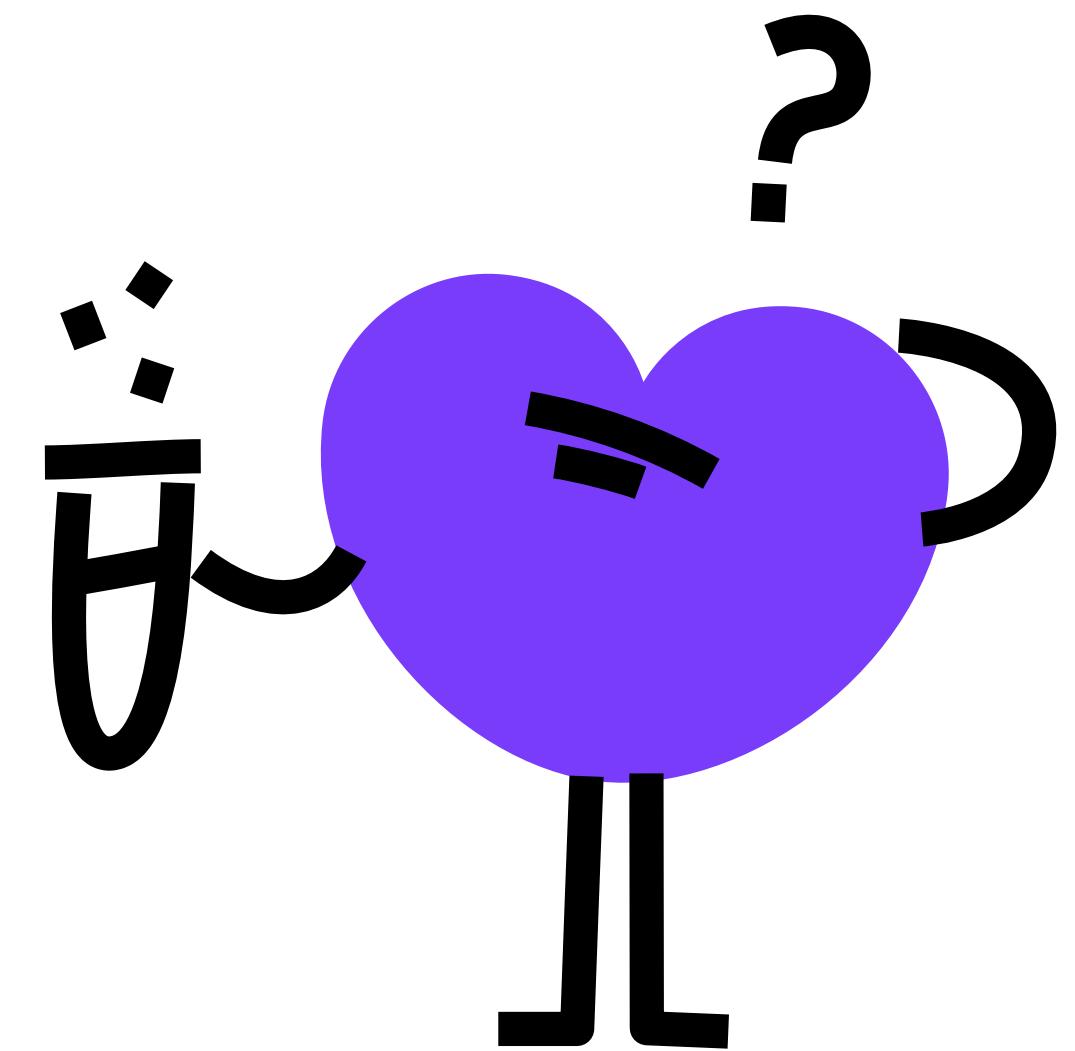
Варианты

- ~~3rd Party~~
 - ~~AsyncDisplayKit, IGListKit и т.д.~~
- **In-house**
 - **UITableView**
 - **UICollectionView**
 - ~~Своя коллекция на основе UIScrollView~~



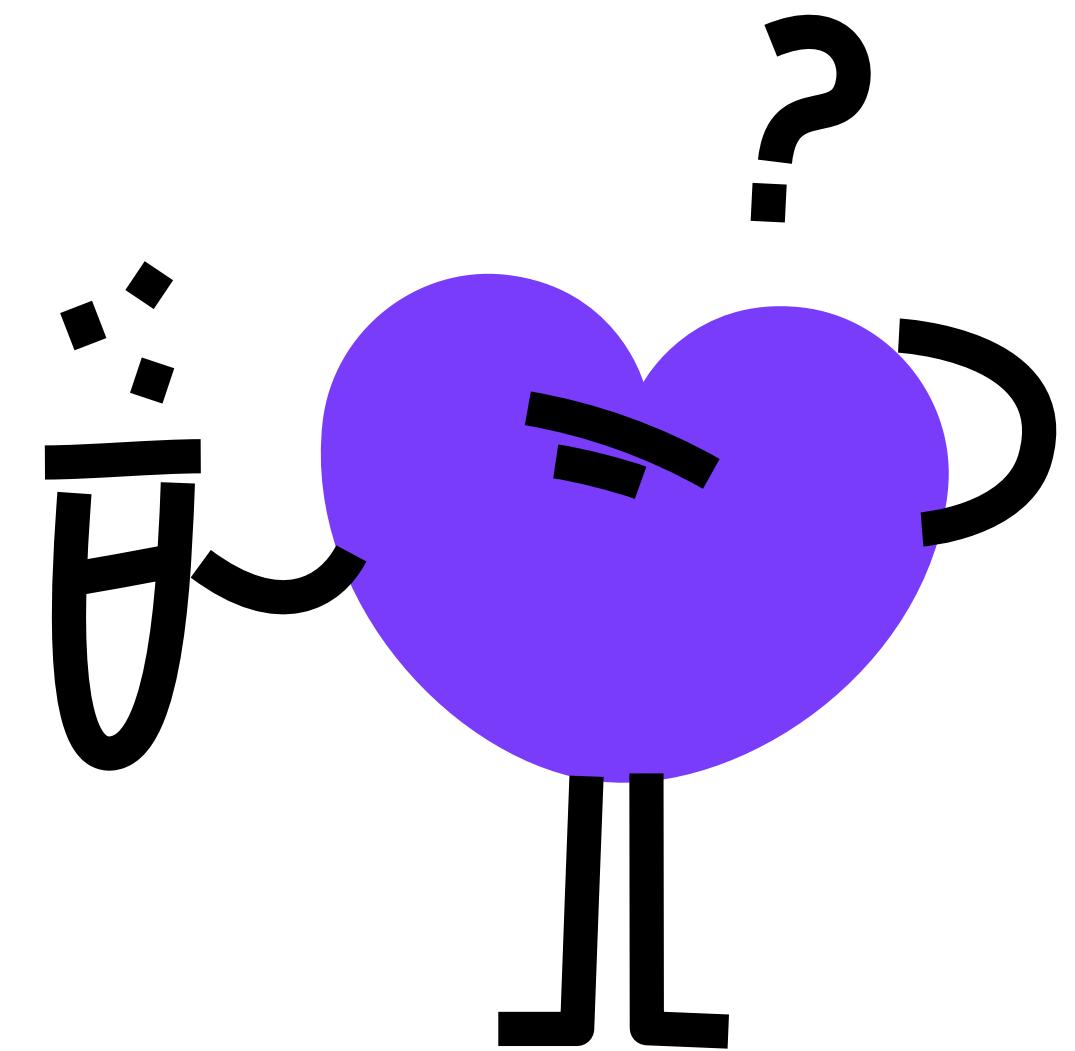
Варианты

- ~~3rd Party~~
 - ~~AsyncDisplayKit, IGListKit и т.д.~~
- **In-house**
 - ~~UITableView~~
 - **UICollectionView**
 - ~~Своя коллекция на основе UIScrollView~~



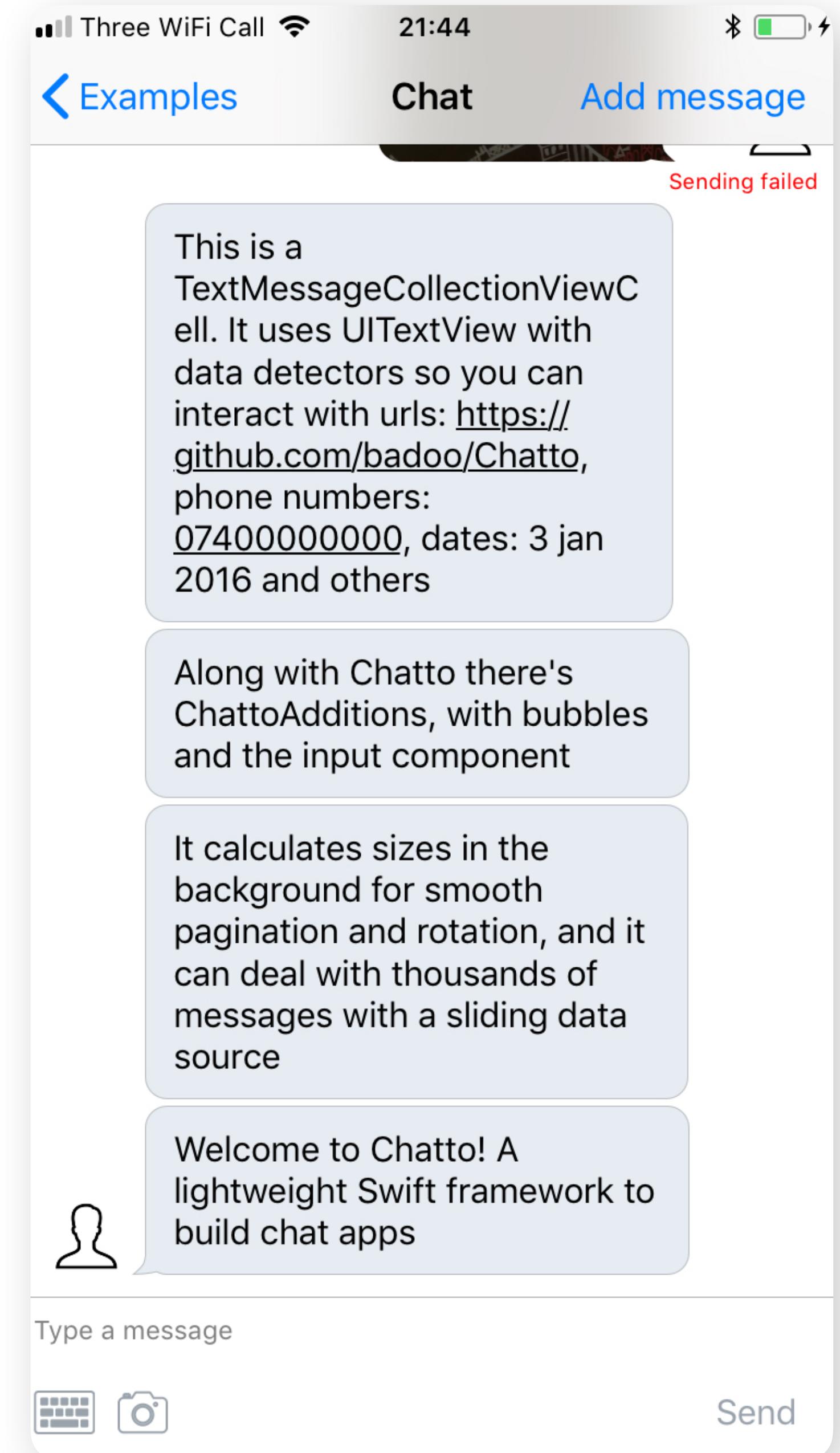
Варианты

- UICollectionView
 - отдельный layout объект
 - batch updates
 - использовался в iMessage



Chatto

- github.com/badoo/Chatto
- использует UICollectionView



Chatto

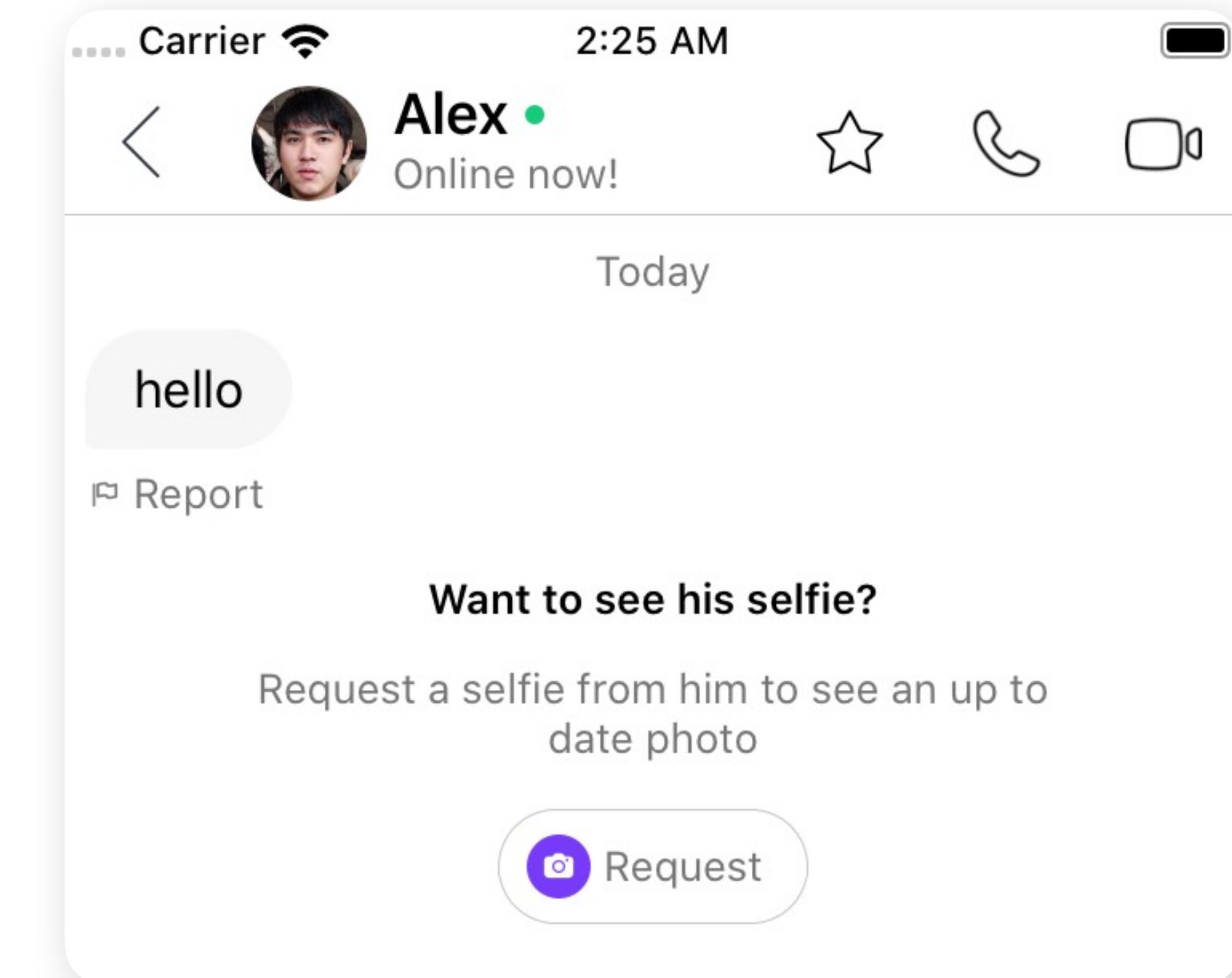
Data-Driven UICollectionView

TimeSeparatorItem

ChatMessage

ReportItem

PromoBlockItem



Model

- **ChatItemProtocol**
 - **отдельный элемент в чате**
 - **имеет уникальный идентификатор**
 - **имеет определенный тип**

View

- **UICollectionViewCell**
 - **любая реализация**
 - **мы используем в связке с "тонкой ViewModel"**

Presenter

- **ChatItemPresenterProtocol**
 - знает как сконфигурировать View для Model
 - знает как посчитать высоту View для Model
 - отвечает за взаимодействие с UICollectionView
 - регистрация и dequeue ячеек
 - реакция на показ и скрытие ячейки
 - поддержка контекстного меню

TextMessage

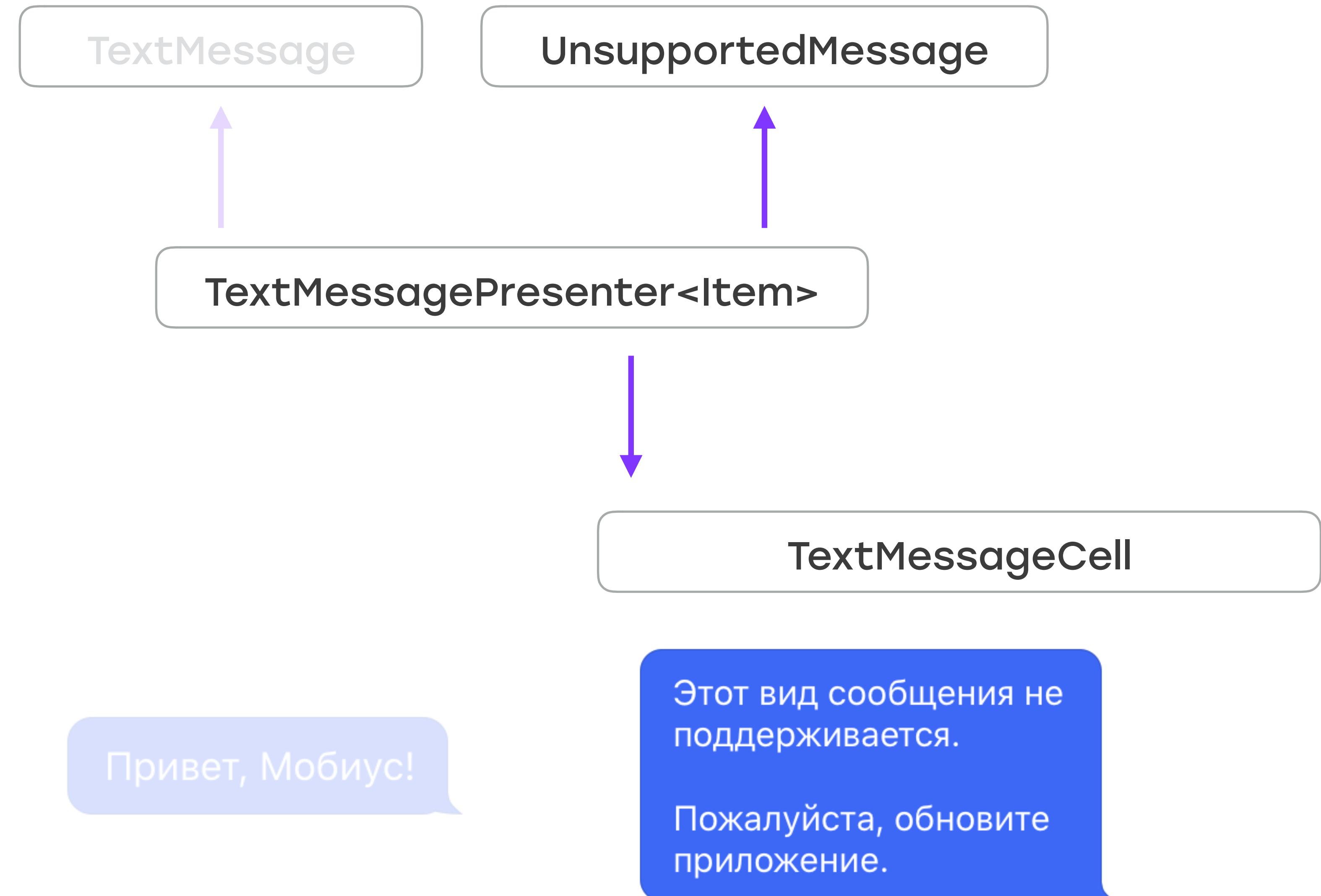


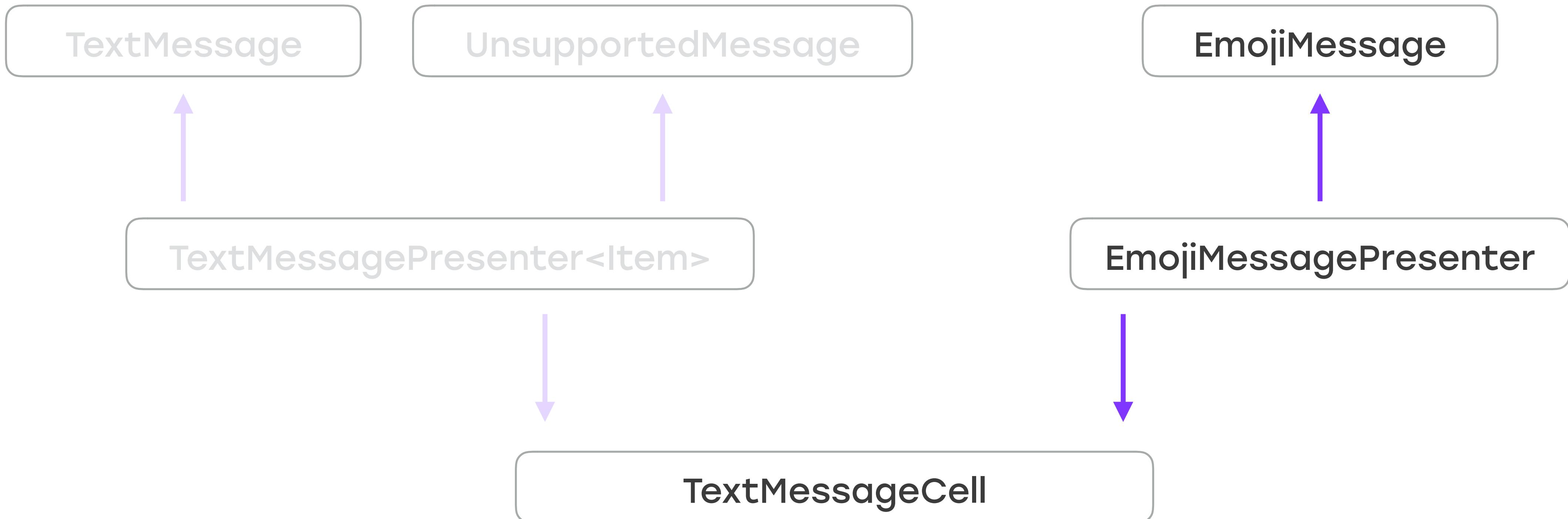
TextMessagePresenter<Item>



TextMessageCell

Привет, Мобиус!

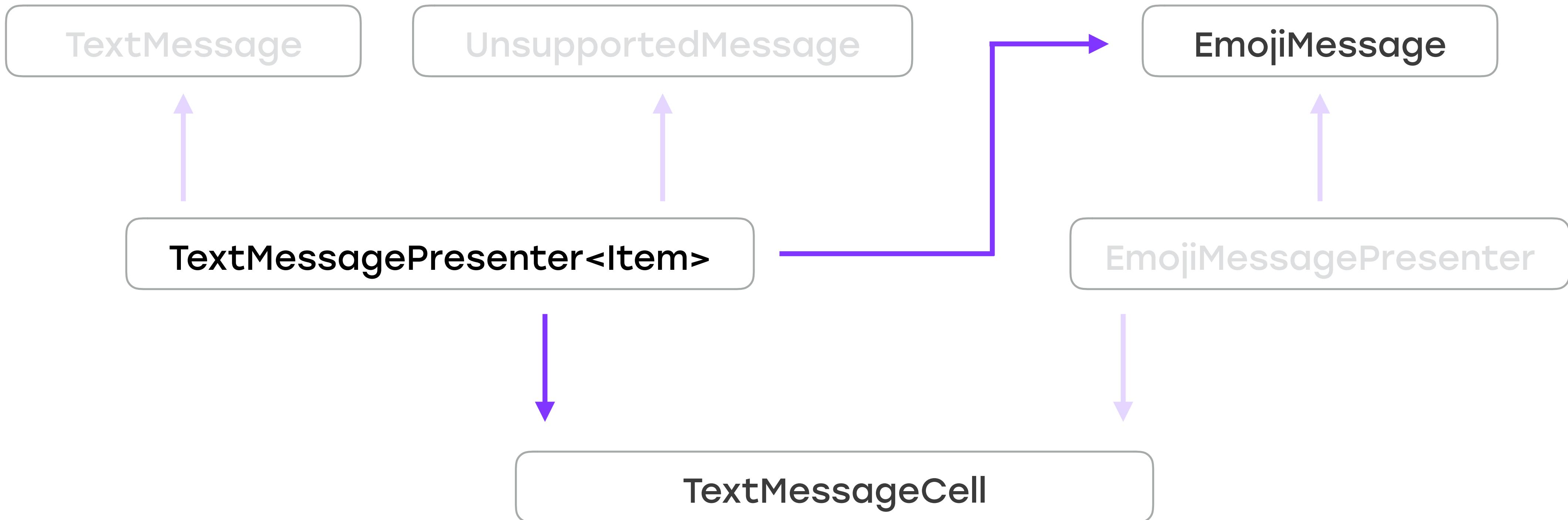




Привет, Мобиус!

Этот вид сообщения не поддерживается.
Пожалуйста, обновите приложение.





Привет, Мобиус!

Этот вид сообщения не поддерживается.
Пожалуйста, обновите приложение.



Как сконфигурировать Chatto?

[

```
TextMessage.type: TextMessagePresenterBuilder<TextMessage>,  
UnsupportedMessage.type: TextMessagePresenterBuilder<UnsupportedMessage>,  
EmojiMessage.type: EmojiMessagePresenterBuilder
```

]

Одна model - один presenter

TextMessage

TextMessage

UnsupportedMessage

EmojiMessage



TextMessagePresenter<TextMessage>

TextMessagePresenter<TextMessage>

TextMessagePresenter<UnsupportedMessage>

EmojiMessagePresenter

Одна model - один presenter

- Хранение состояния элемента
 - Посчитанный layout
 - Состояние загрузки фото
 - Секунда, на которой остановилось воспроизведение

Одна model - один presenter

- Хранение состояния элемента
 - Посчитанный layout
 - Состояние загрузки фото
 - Секунда, на которой остановилось воспроизведение
- Изменение состояния элемента
 - Останавливать загрузку фото при скрытии ячейки
 - Загрузка link preview при показе ячейки

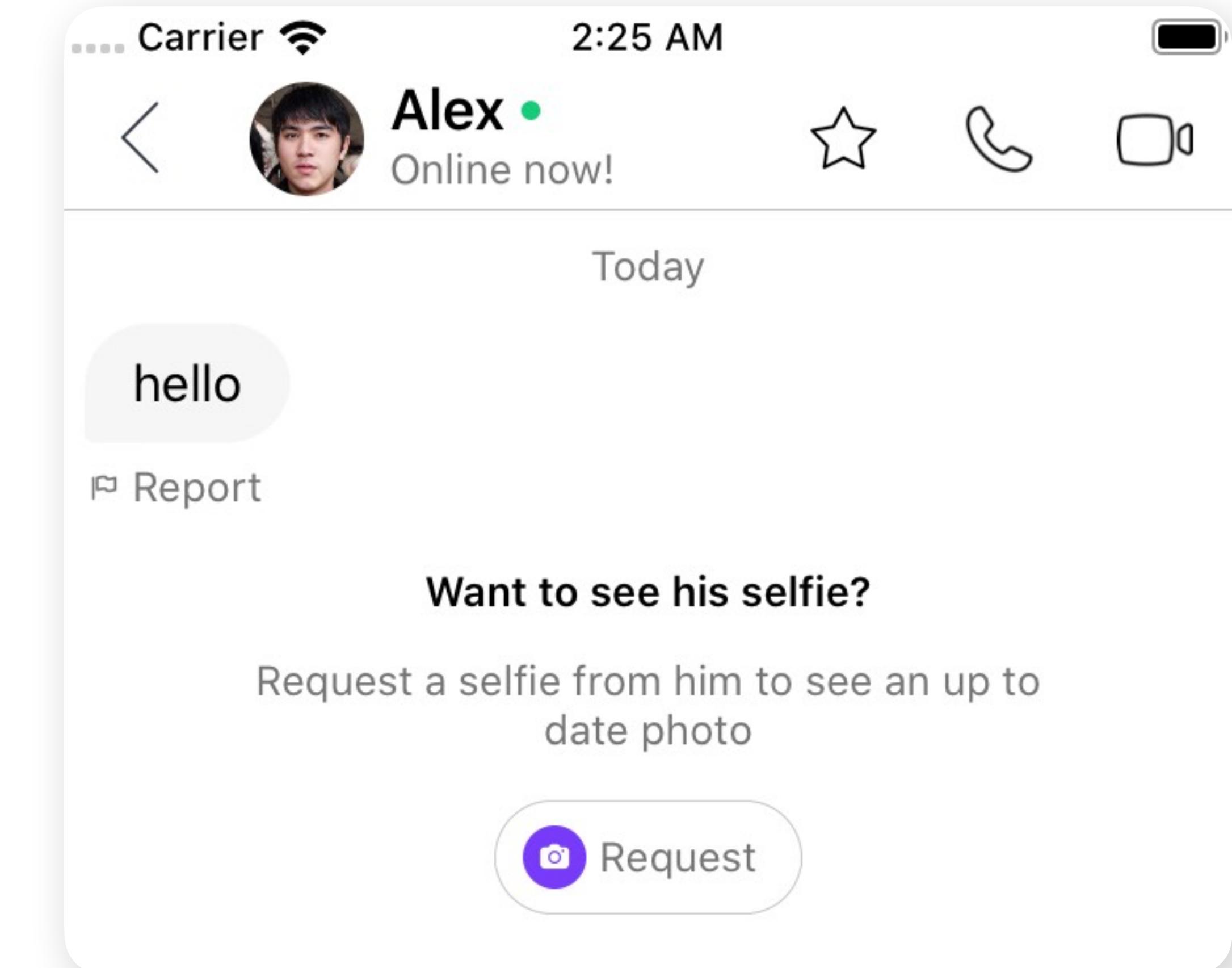
Data-Driven UICollectionView

TimeSeparatorItem

ChatMessage

ReportItem

PromoBlockItem



UICollectionView

UICollectionView

UICollectionViewDataSource

- Количество элементов
- View для элементов

UICollectionView

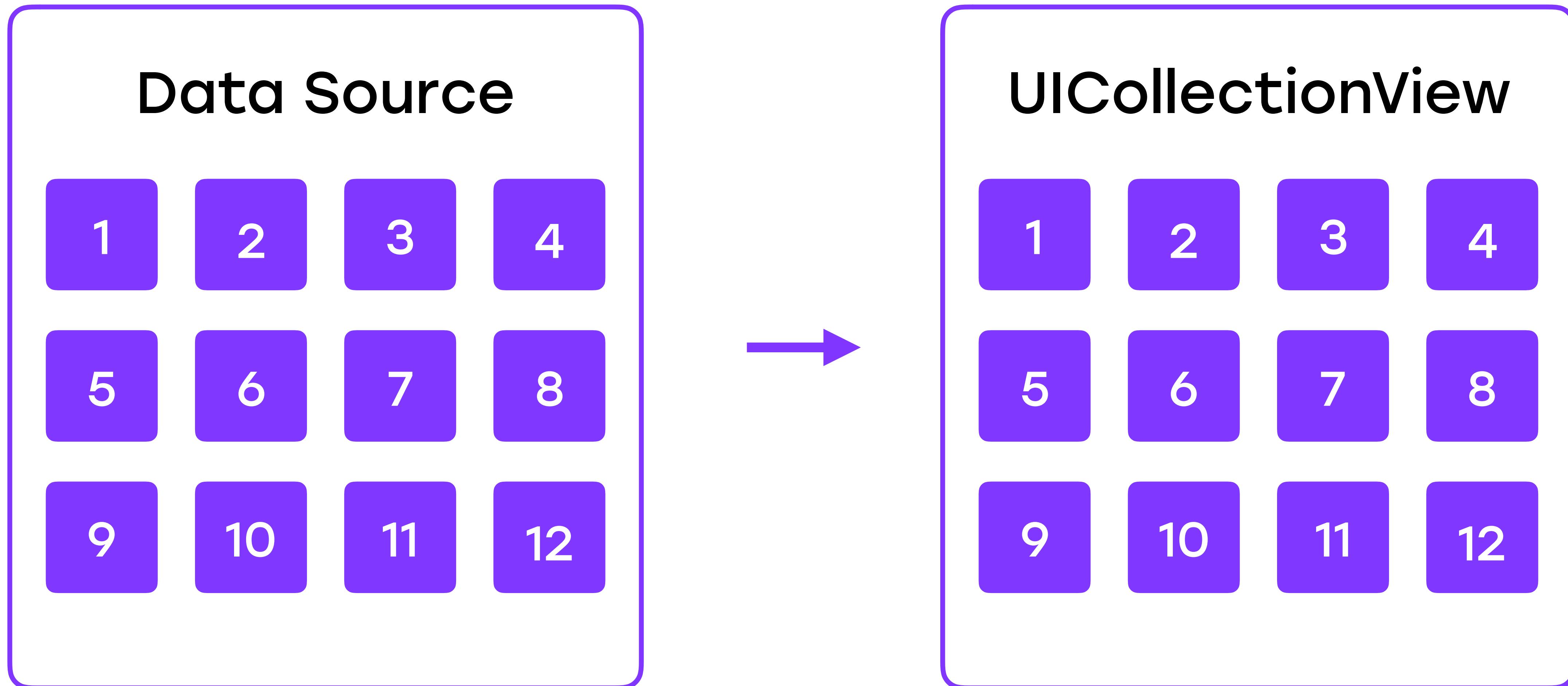
UICollectionViewDataSource

- Количество элементов
- View для элементов

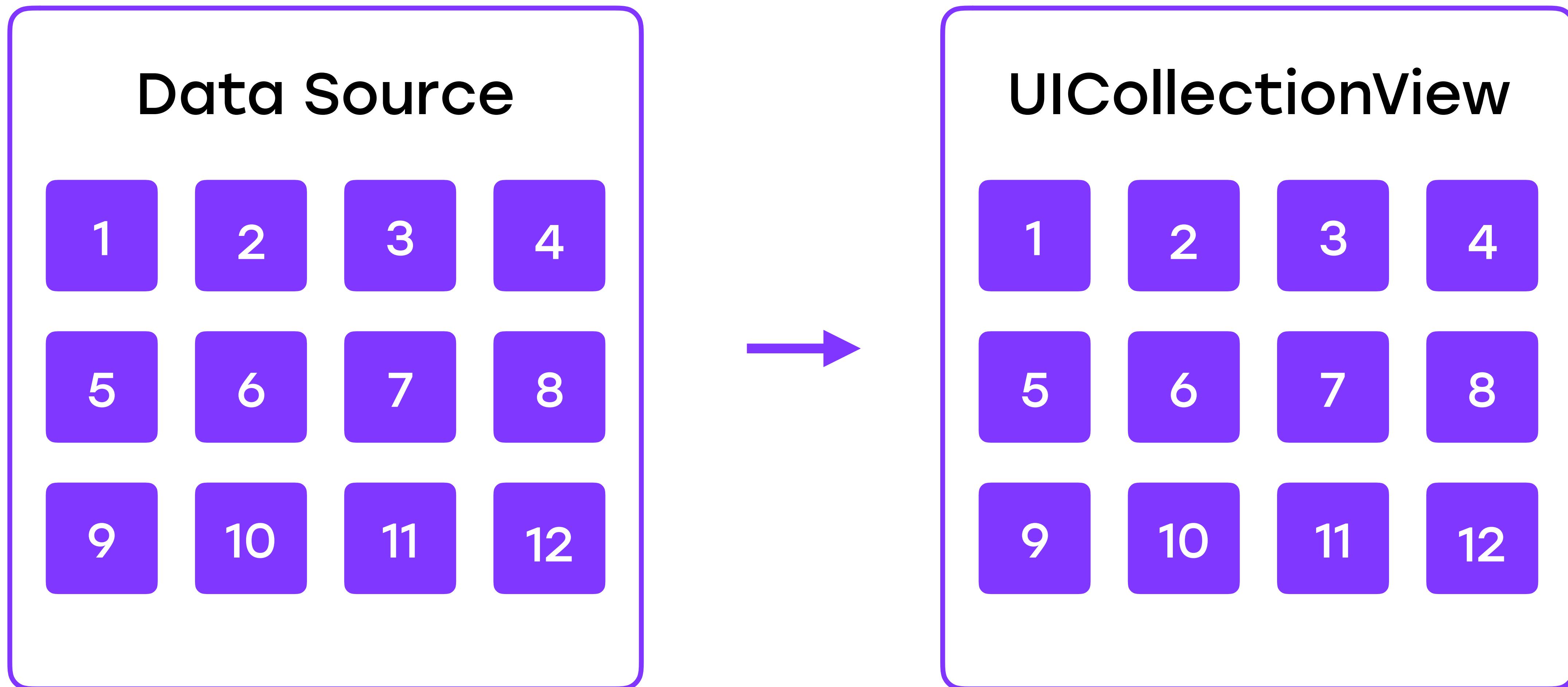
UICollectionViewLayout

- Расположение элементов
- Видимые элементы

Обновление UICollectionView



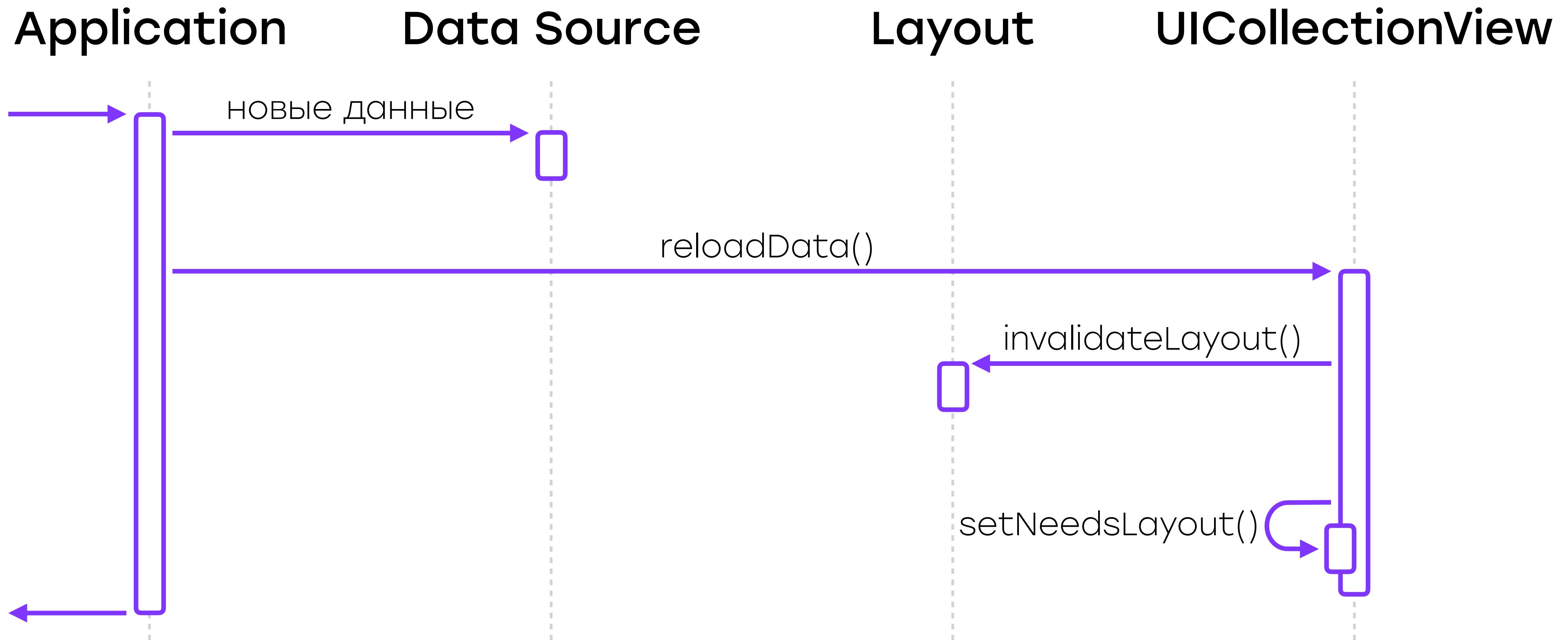
Обновление UICollectionView



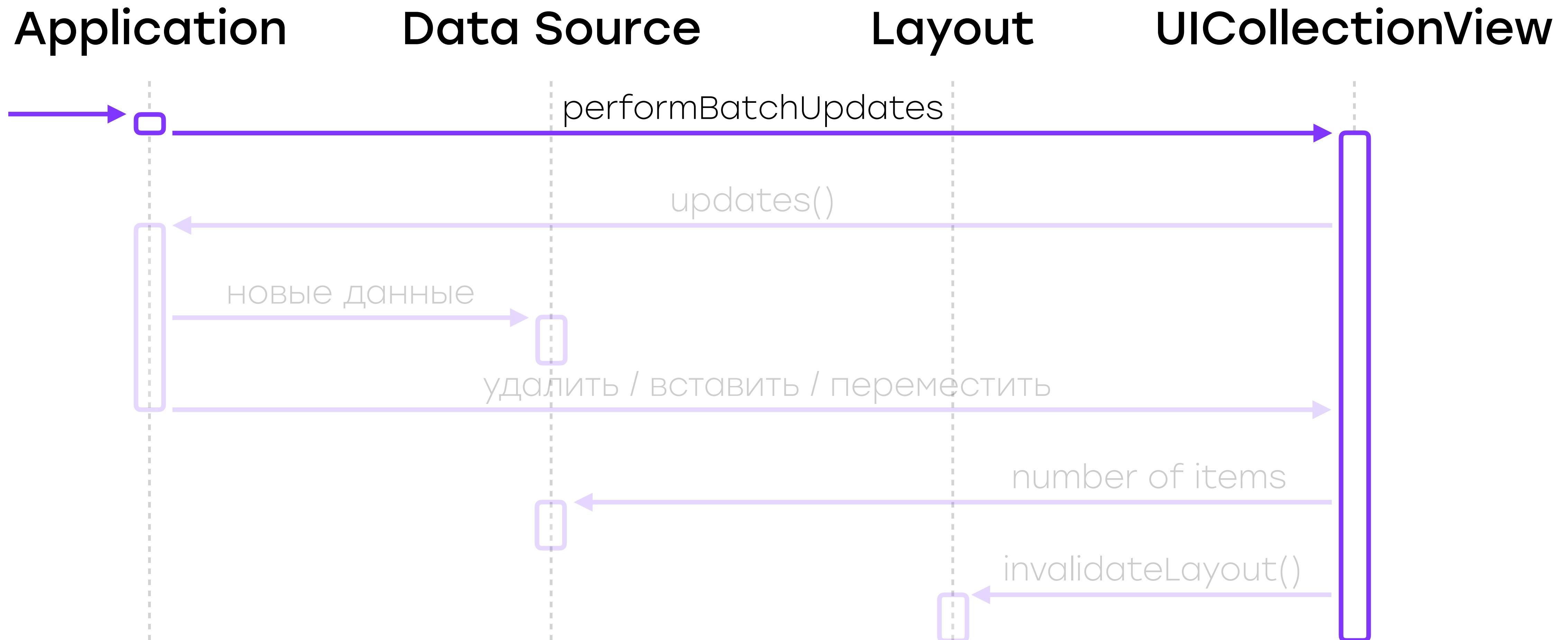
Обновление UICollectionView

- `reloadData()`
- `performBatchUpdates(_:completion:)`

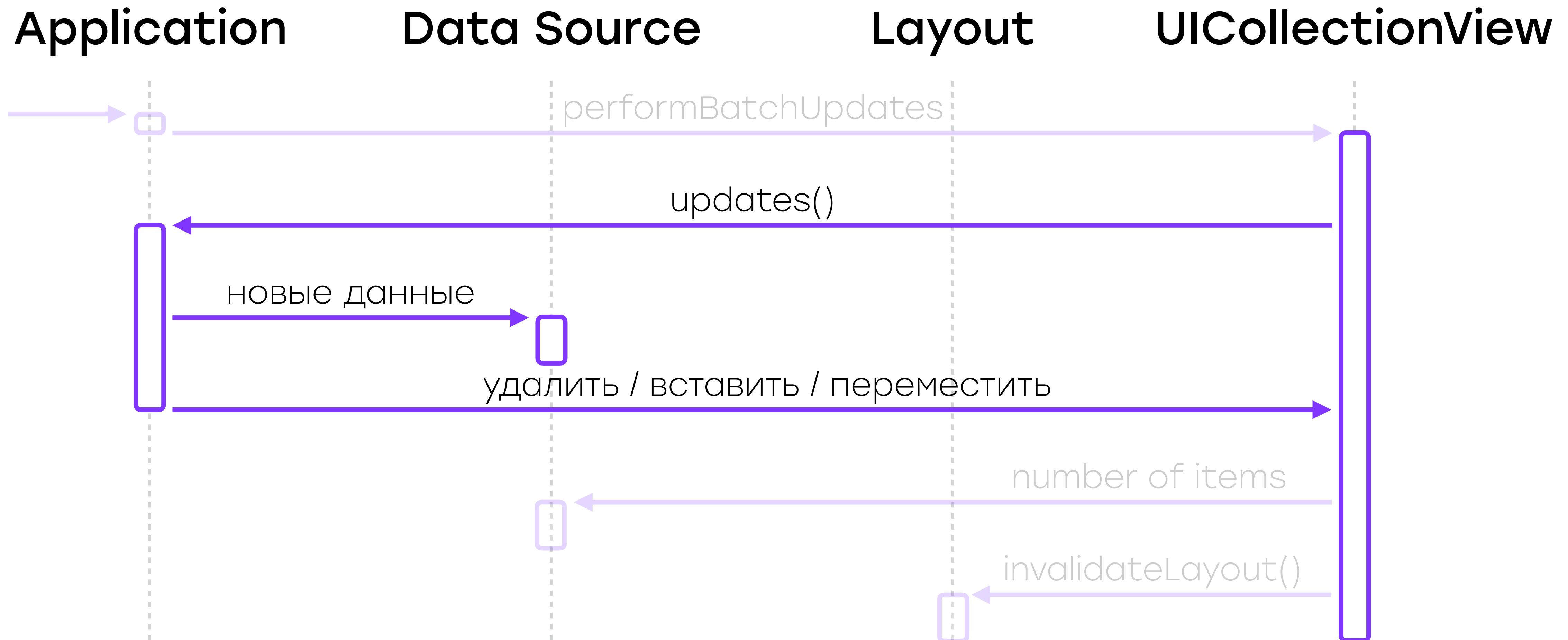
reloadData()



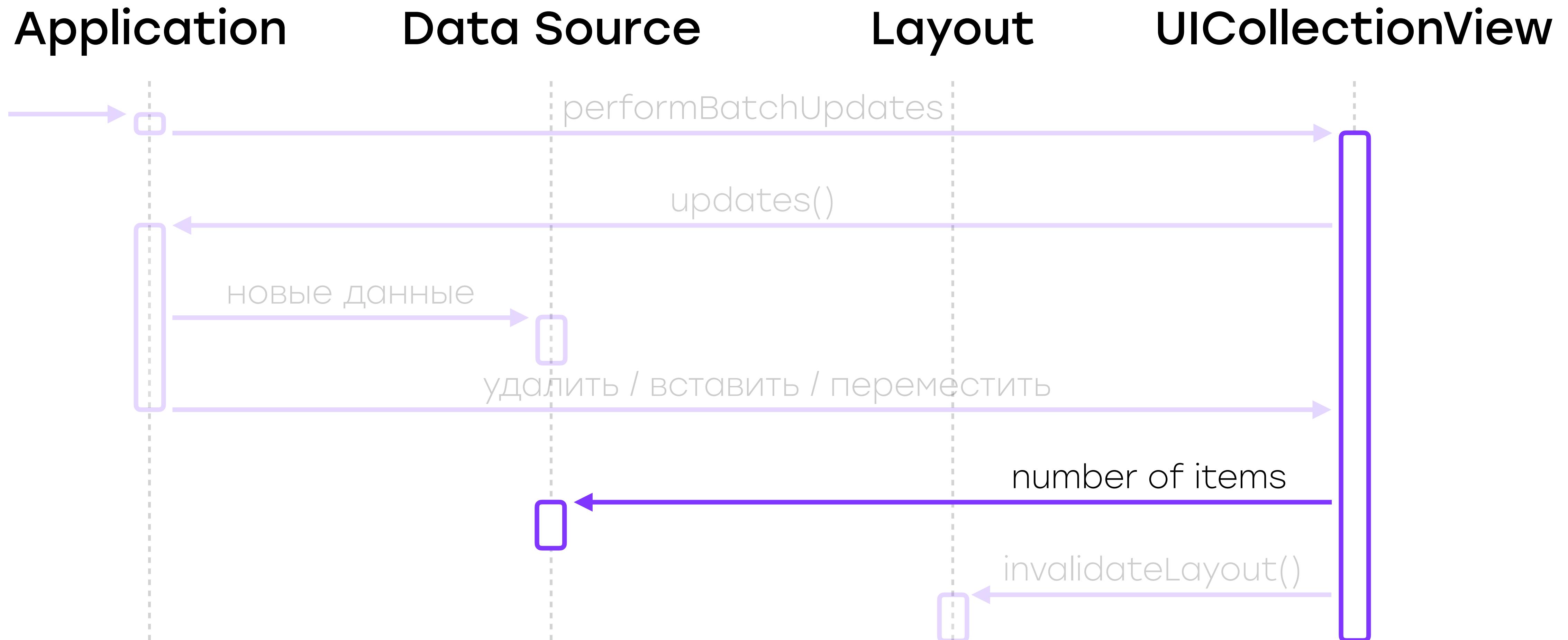
performBatchUpdates(_:completion:)



performBatchUpdates(_:completion:)



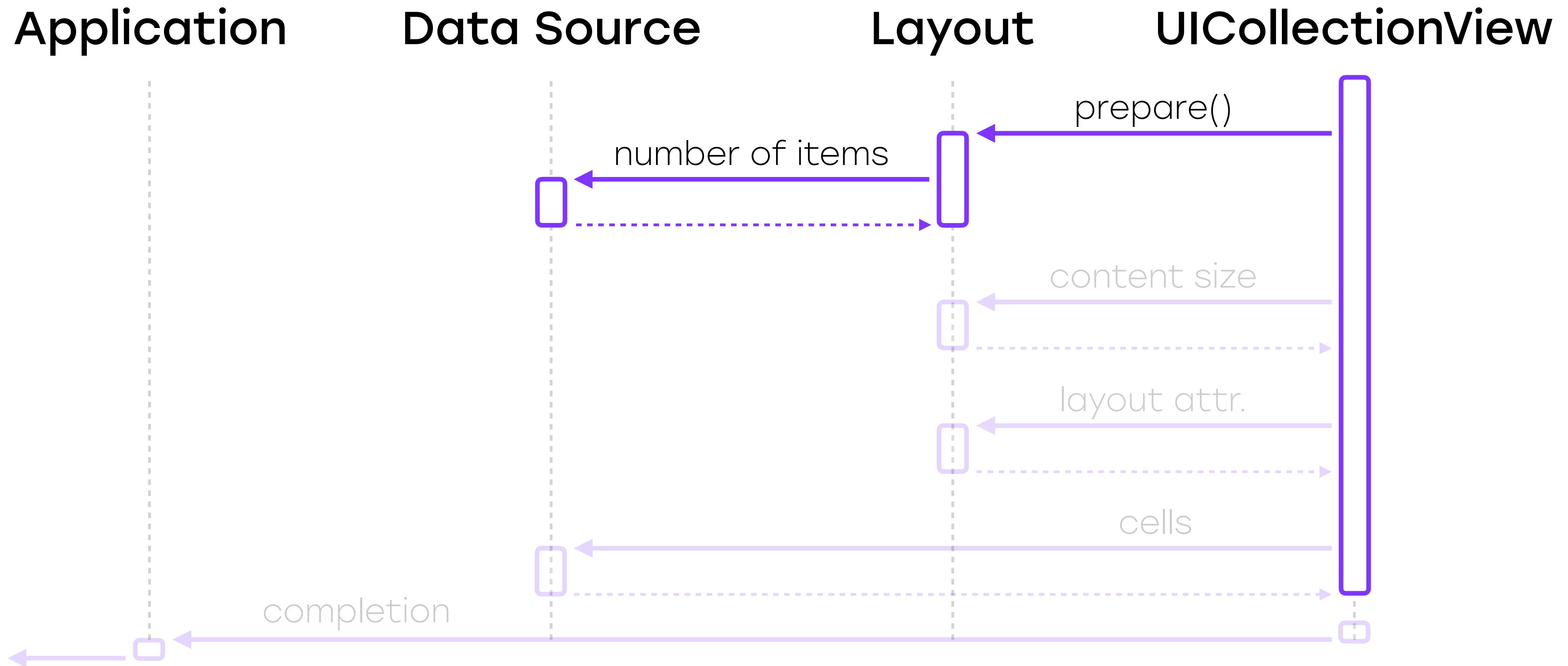
performBatchUpdates(_:completion:)



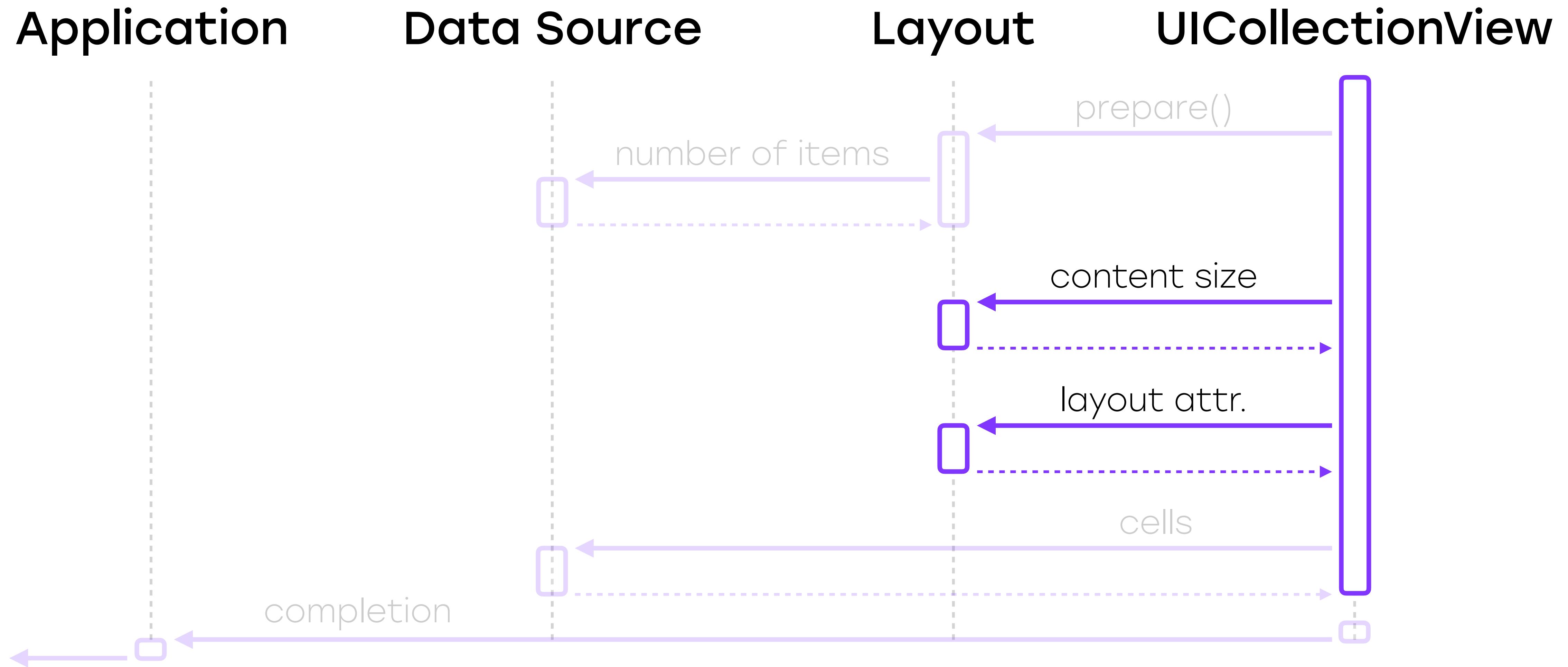
performBatchUpdates(_:completion:)



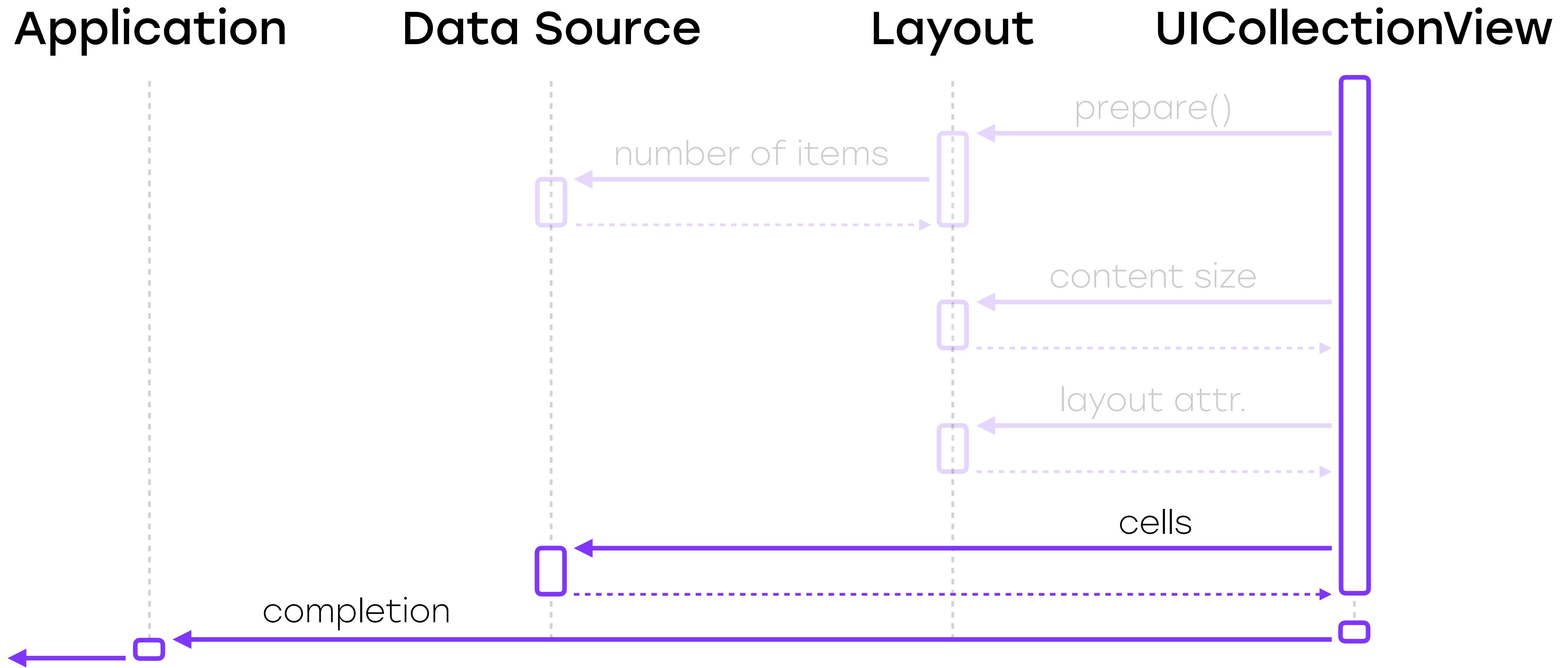
performBatchUpdates(_:completion:)



performBatchUpdates(_:completion:)



performBatchUpdates(_:completion:)



Какой тип обновления использовать?

`reloadData()`

- **первое обновление**
- **постраничная загрузка (без анимаций)**

Какой тип обновления использовать?

`reloadData()`

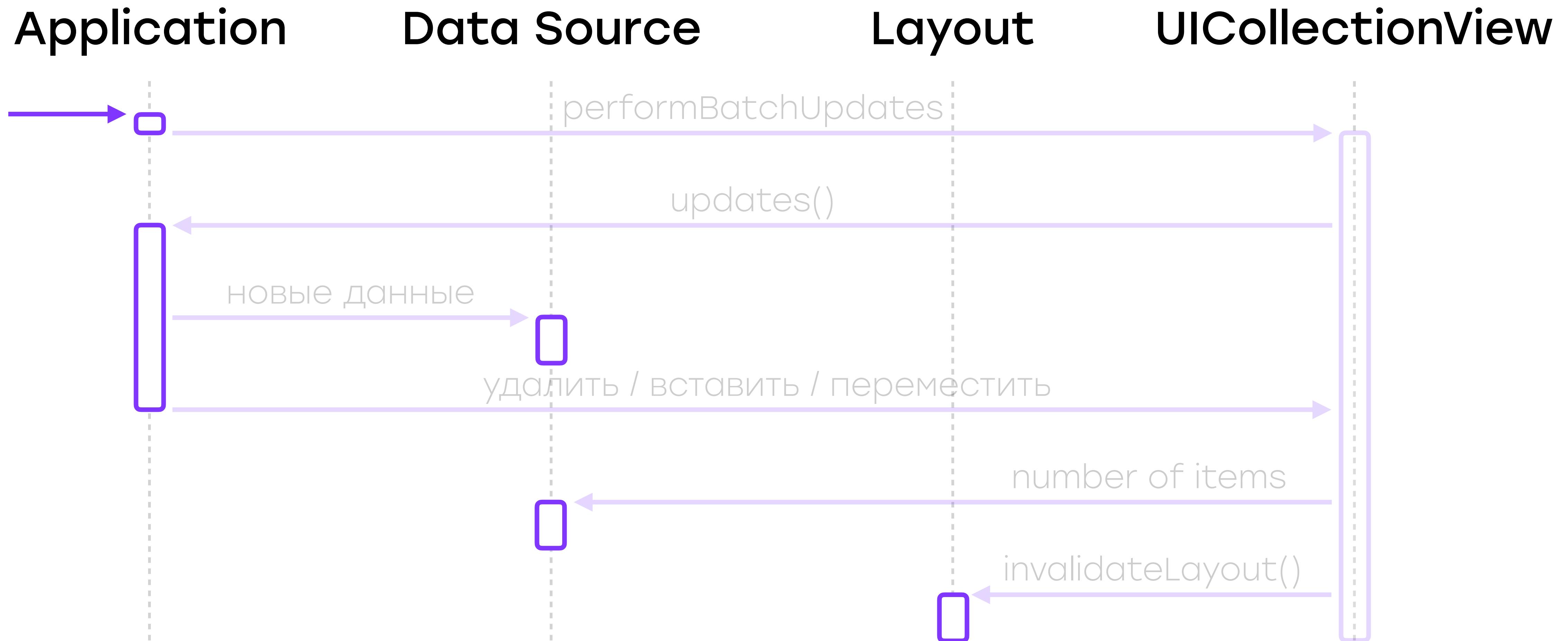
- **первое обновление**
- **постраничная загрузка (без анимаций)**

`performBatchUpdates(_:completion:)`

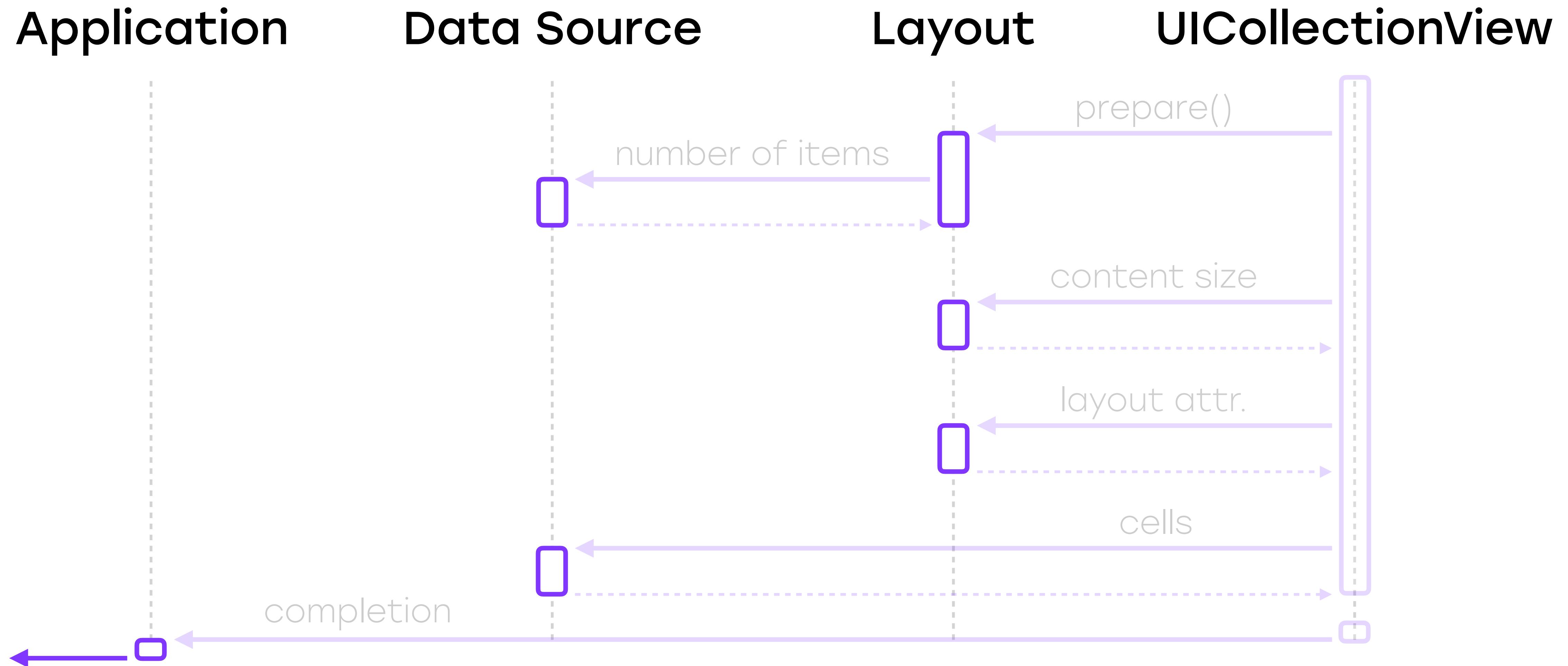
- **одиночные обновления**
- **групповые обновления**

Оптимизация UICollectionView

performBatchUpdates(_:completion:)



performBatchUpdates(_:completion:)



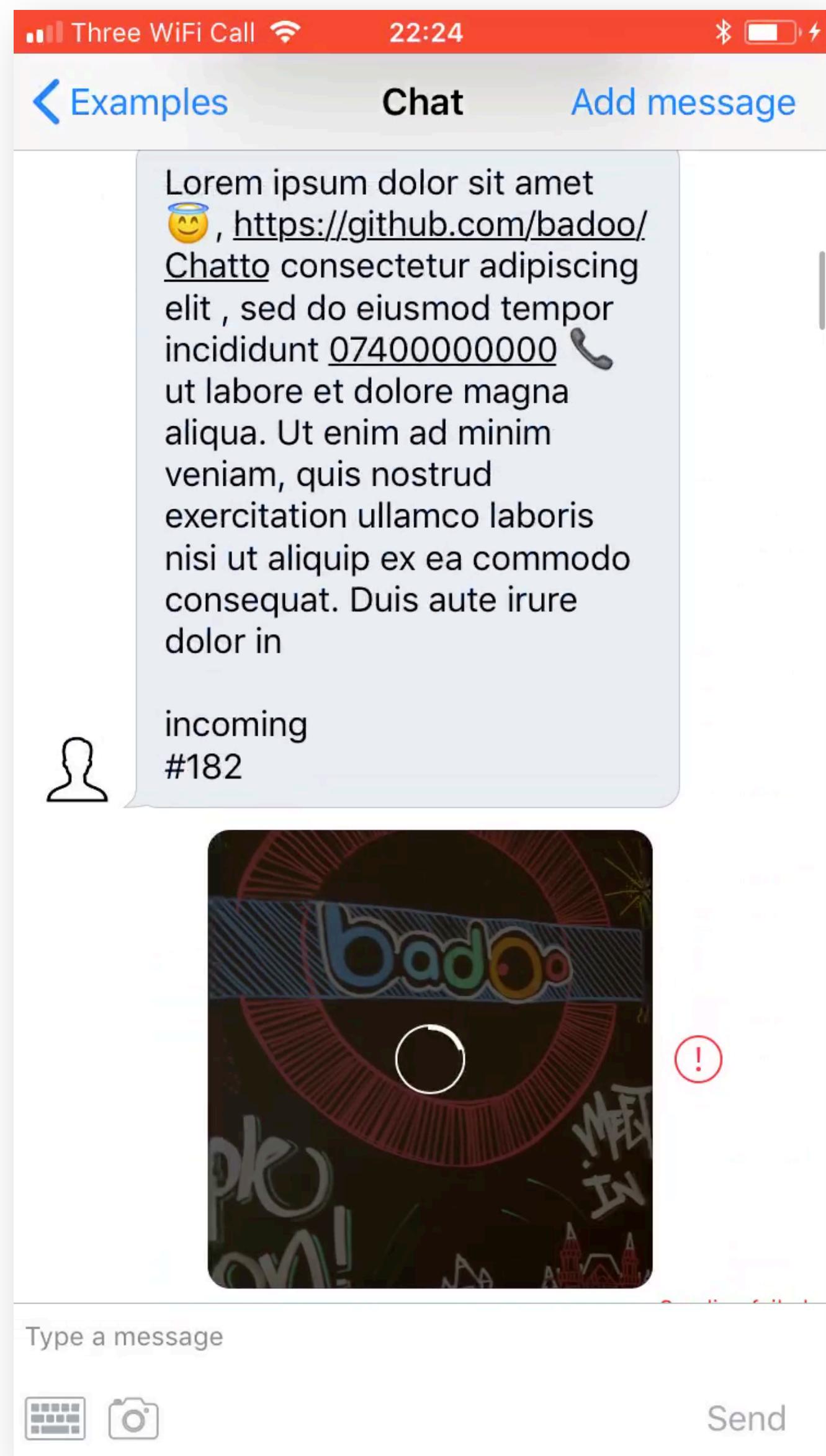
Основные шаги обновления

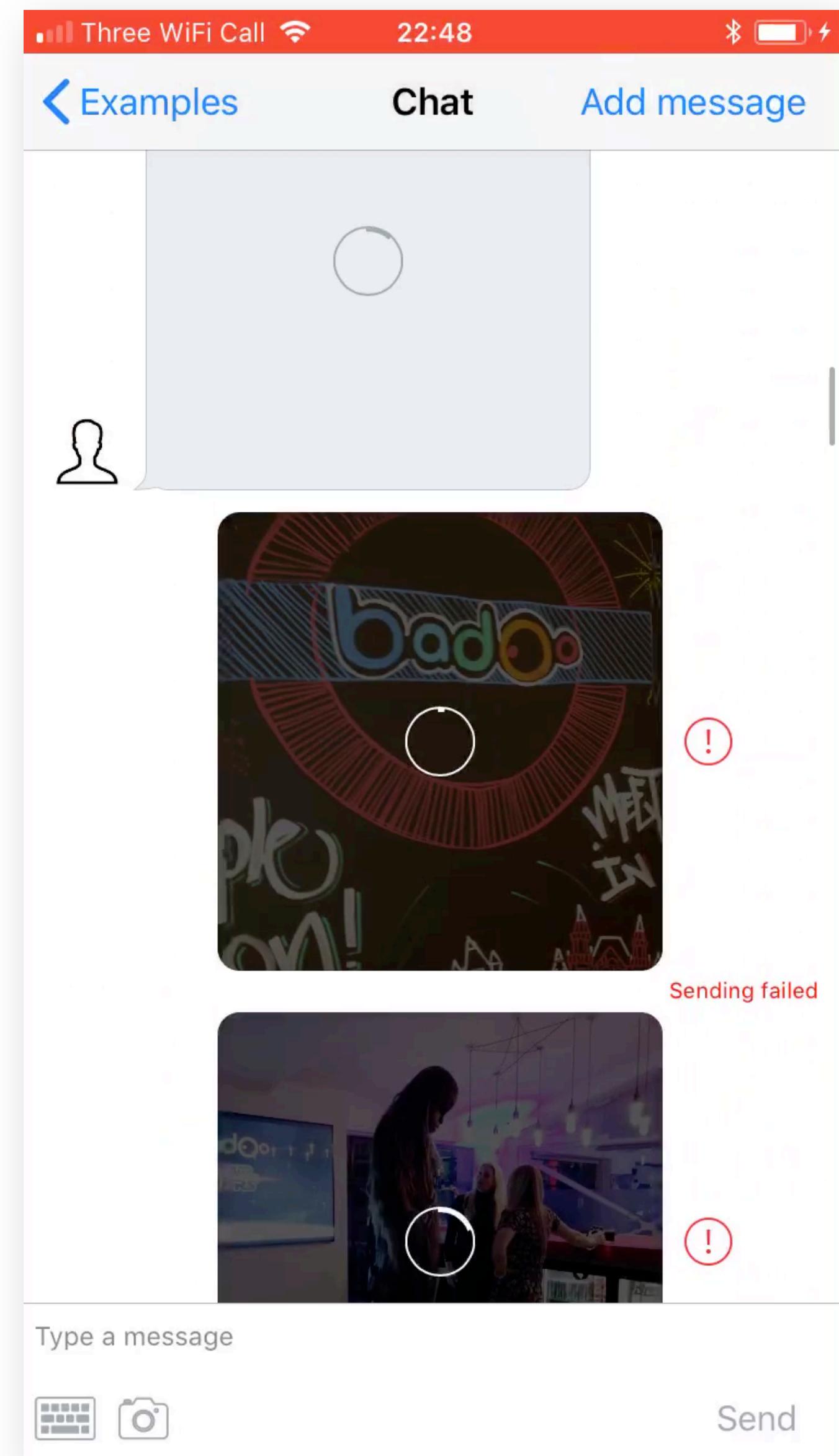
- Подготовка списка элементов
- Обновление коллекции
- Обновление layout'a
- Обновление ячеек

Основные шаги обновления

- Подготовка списка элементов
- Обновление коллекции
- Обновление layout'a
- Обновление ячеек

< 1/60s





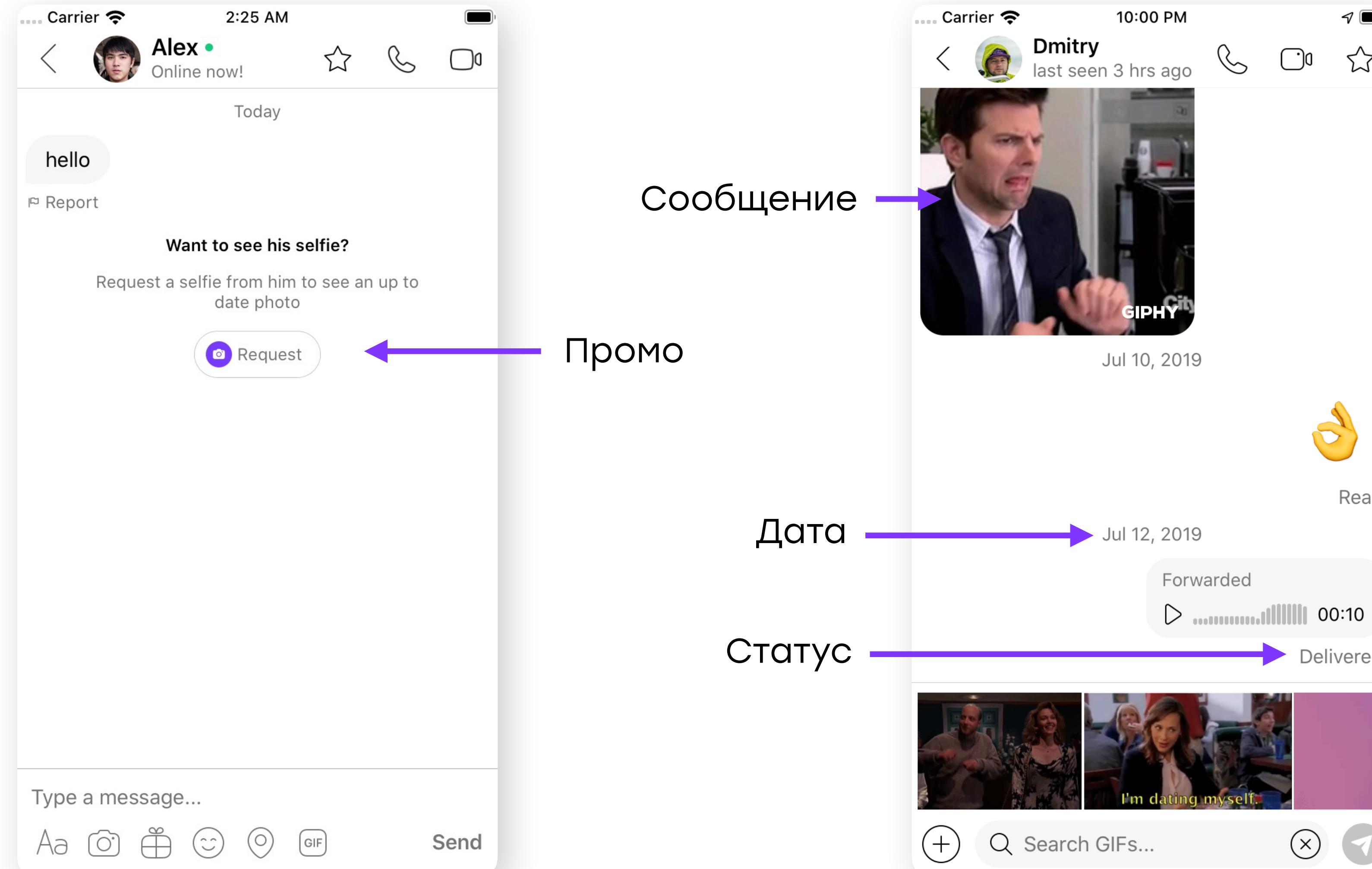
Оптимизация обновлений

- Освободить главным потоком по максимуму
- Посчитать все заранее

Основные шаги обновления

- **Подготовка списка элементов**
 - Обновление коллекции
 - Обновление layout'a
 - Обновление ячеек

Подготовка списка элементов

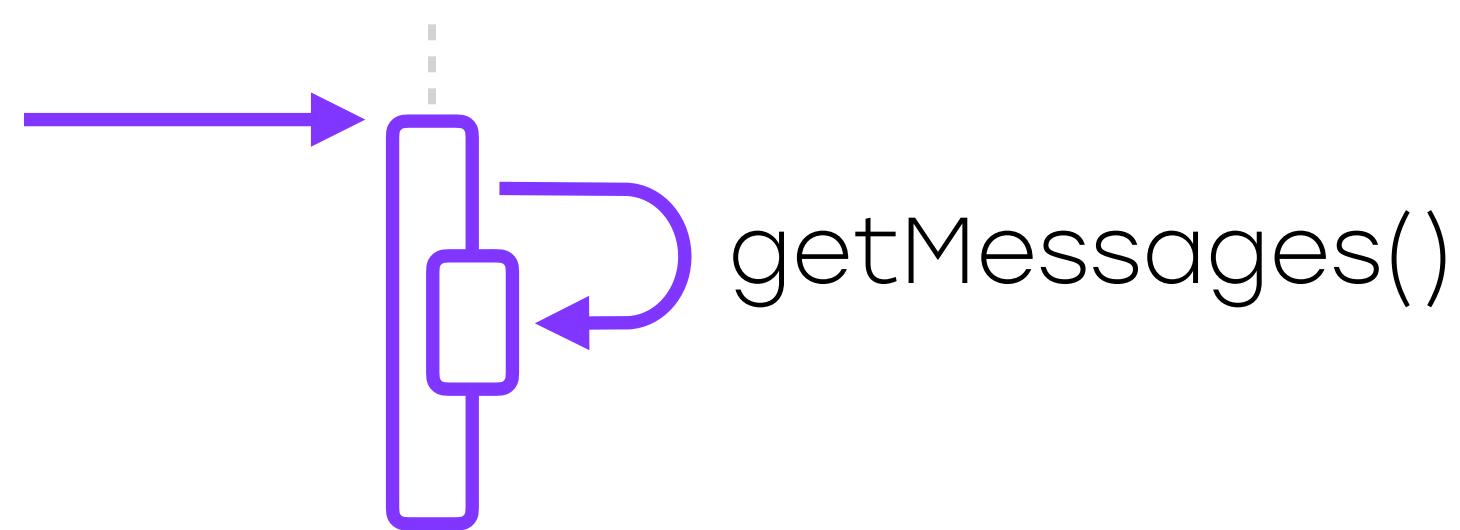


Подготовка списка элементов

- загрузка сообщений
- добавление UI элементов

Загрузка сообщений

Application



Chatto



UICollectionView

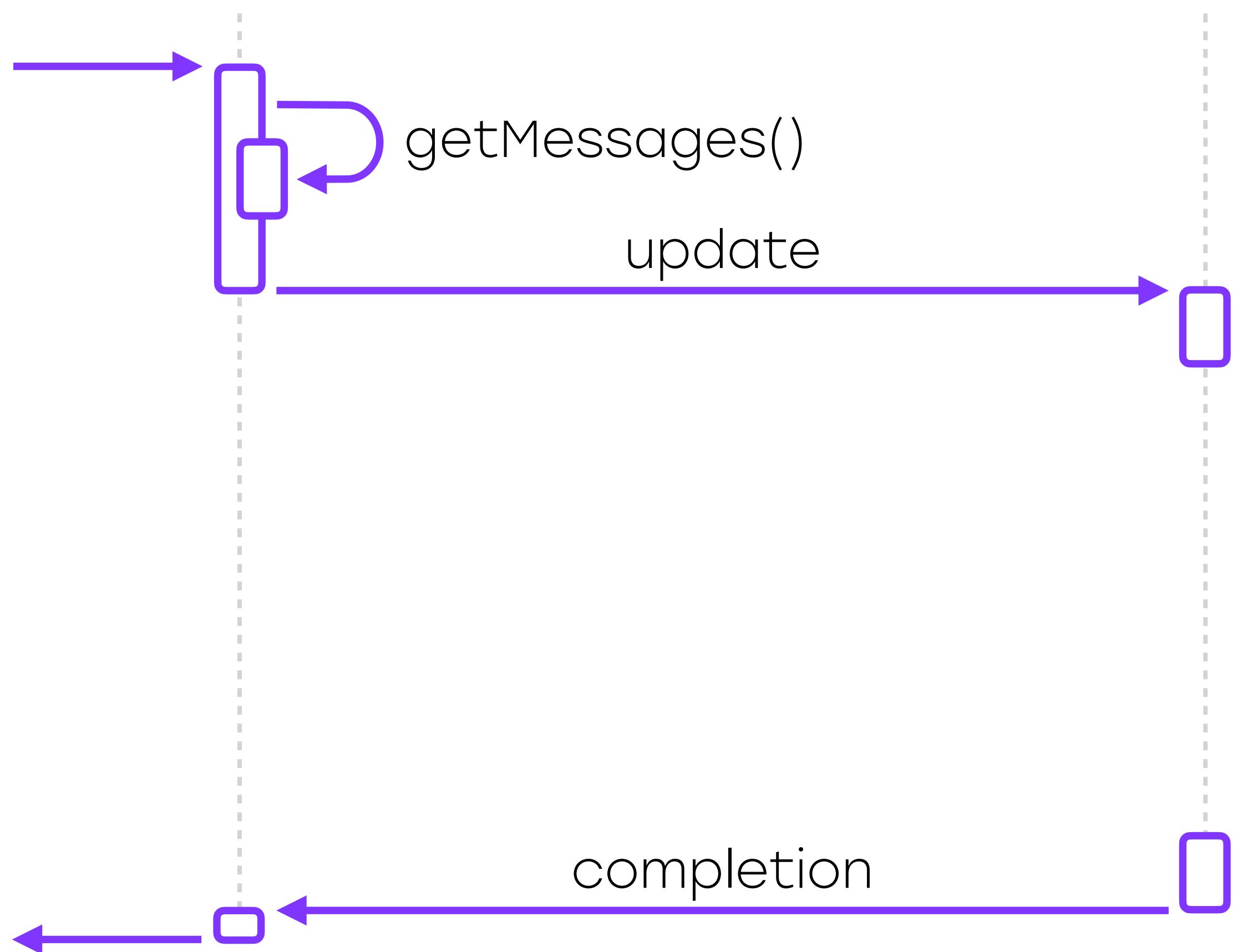


Начало обновления

Application

Chatto

UICollectionView

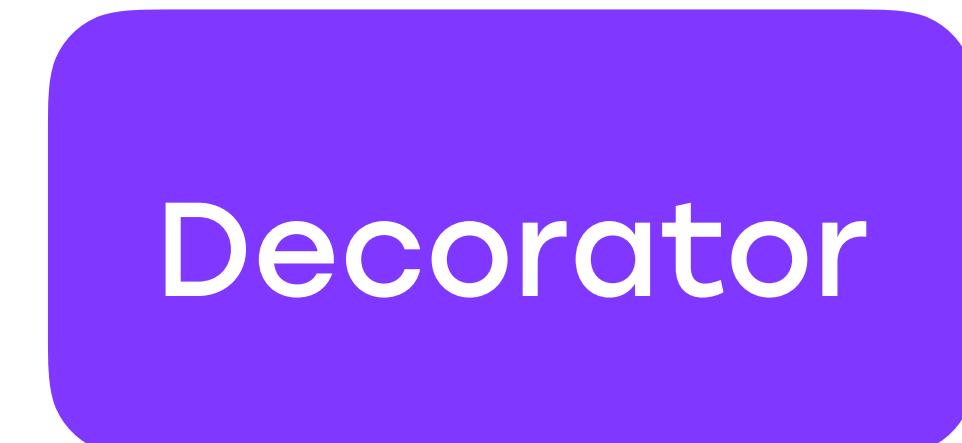


Добавление UI элементов

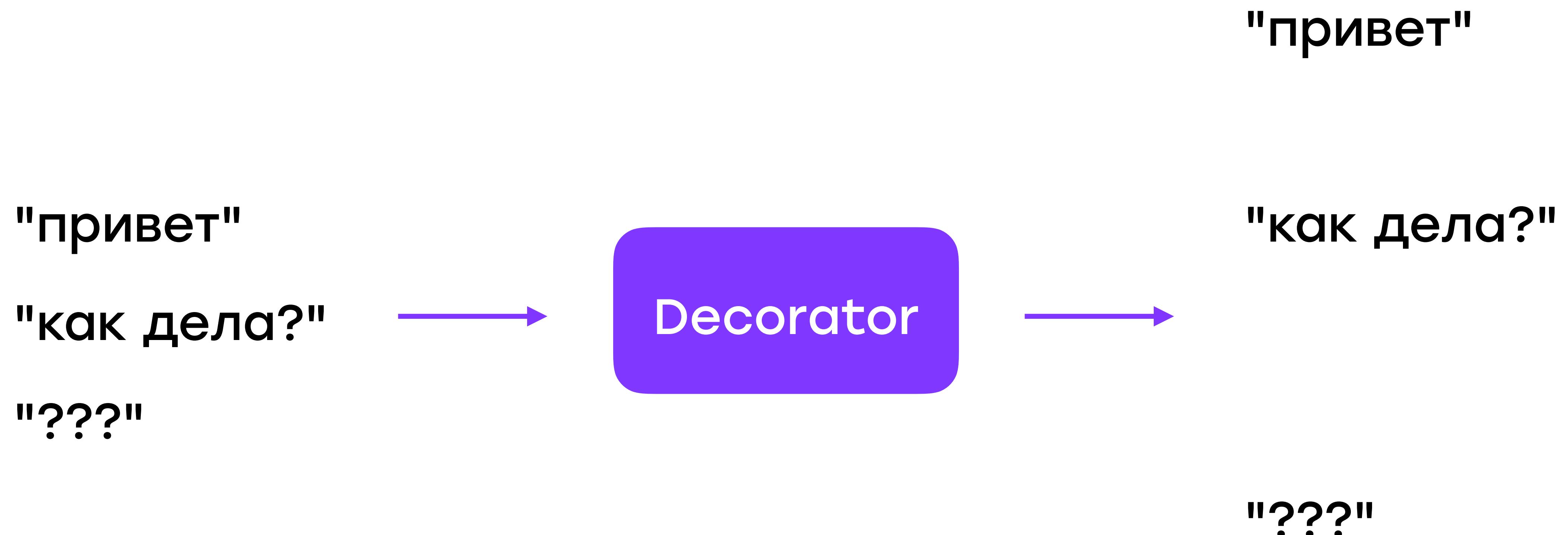
"привет"

"как дела?"

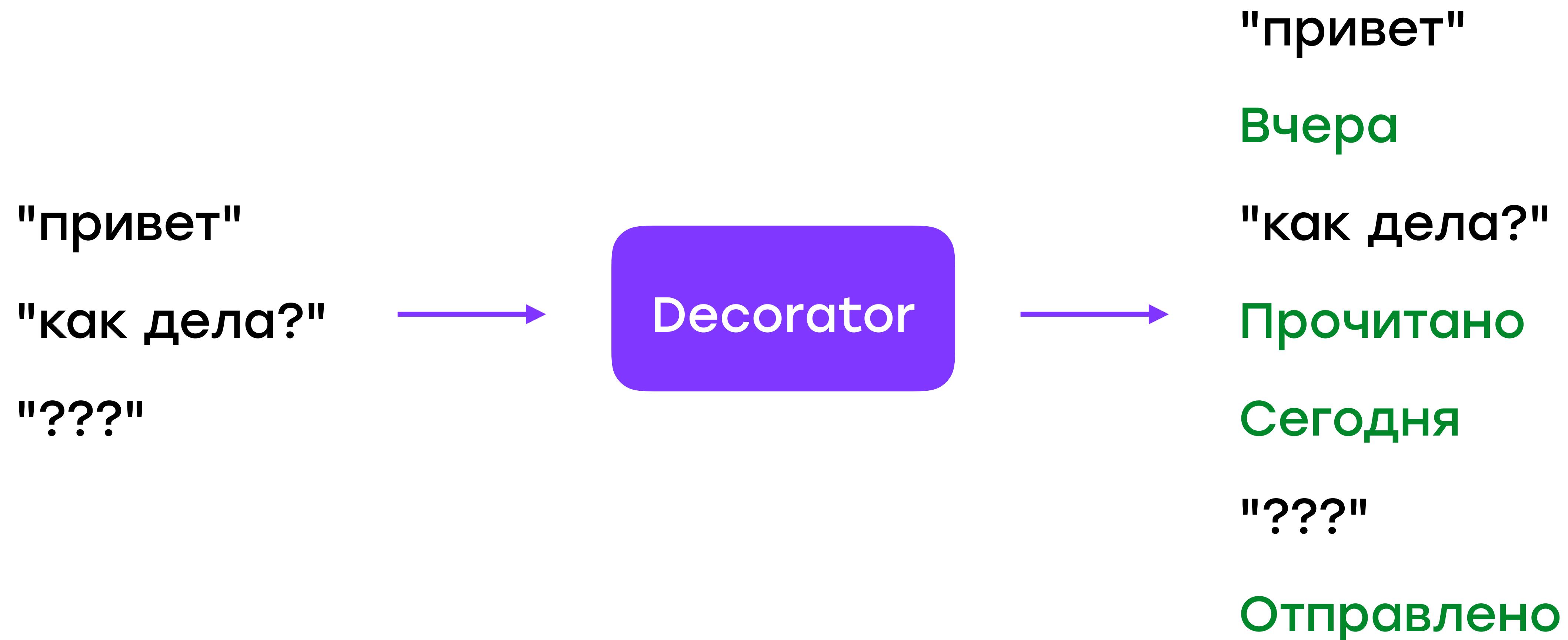
"???"



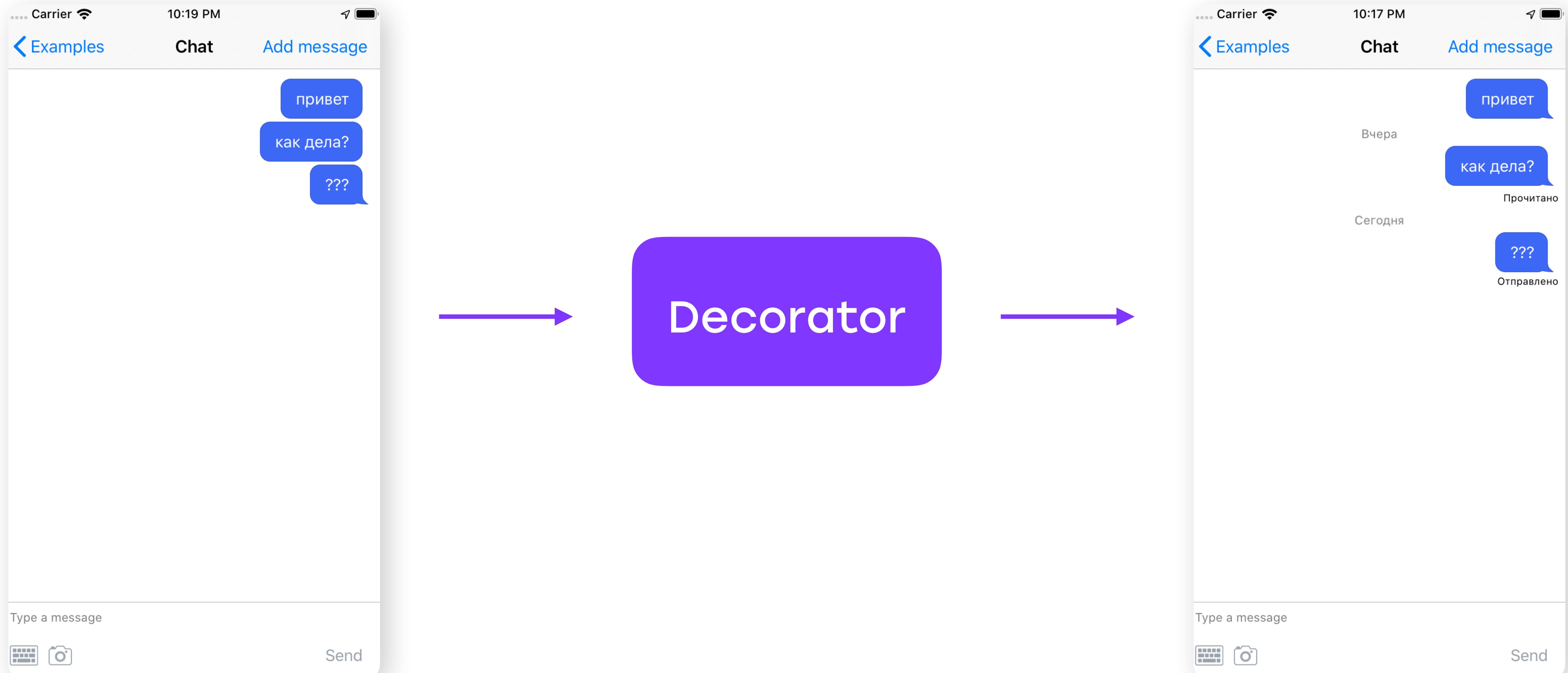
Добавление UI элементов



Добавление UI элементов



Добавление UI элементов



Декоратор

- Добавляет UI элементы
 - Выполняется в фоновом потоке
 - Сложность - $O(n)$

Декоратор

- Добавляет UI элементы
 - Выполняется в фоновом потоке
 - Сложность - $O(n)$
- Считает дополнительные UI атрибуты для сообщений
 - отступы между соседними сообщениями
 - наличие хвостиков у облачков
 - показ аватара пользователя

Подготовка списка элементов

- Какую структуру данных выбрать?

Подготовка списка элементов

- Какую структуру данных выбрать?
 - доступ по индексу - константа

Подготовка списка элементов

- Какую структуру данных выбрать?
 - доступ по индексу - константа
 - доступ по ключу - константа

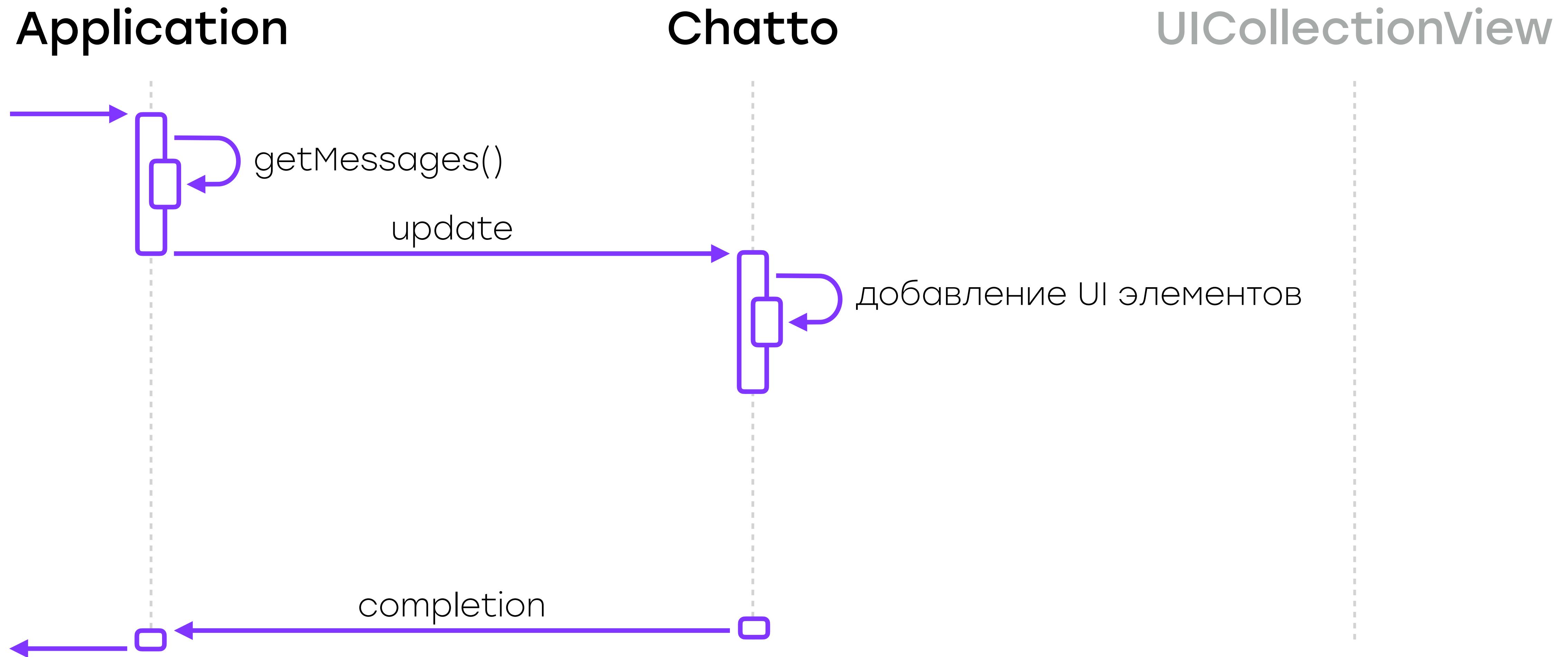
Подготовка списка элементов

- Какую структуру данных выбрать?
 - доступ по индексу - константа
 - доступ по ключу - константа
- `OrderedDictionary`

Подготовка списка элементов

- Какую структуру данных выбрать?
 - доступ по индексу - константа
 - доступ по ключу - константа
- `OrderedDictionary`
 - `Array` для доступа по индексу
 - `Dictionary` для доступа по ключу

Подготовка списка элементов



Основные шаги обновления

- Подготовка списка элементов
- **Обновление коллекции**
- Обновление layout'a
- Обновление ячеек

Подготовка обновления коллекции

- Как посчитать diff?

Подготовка обновления коллекции

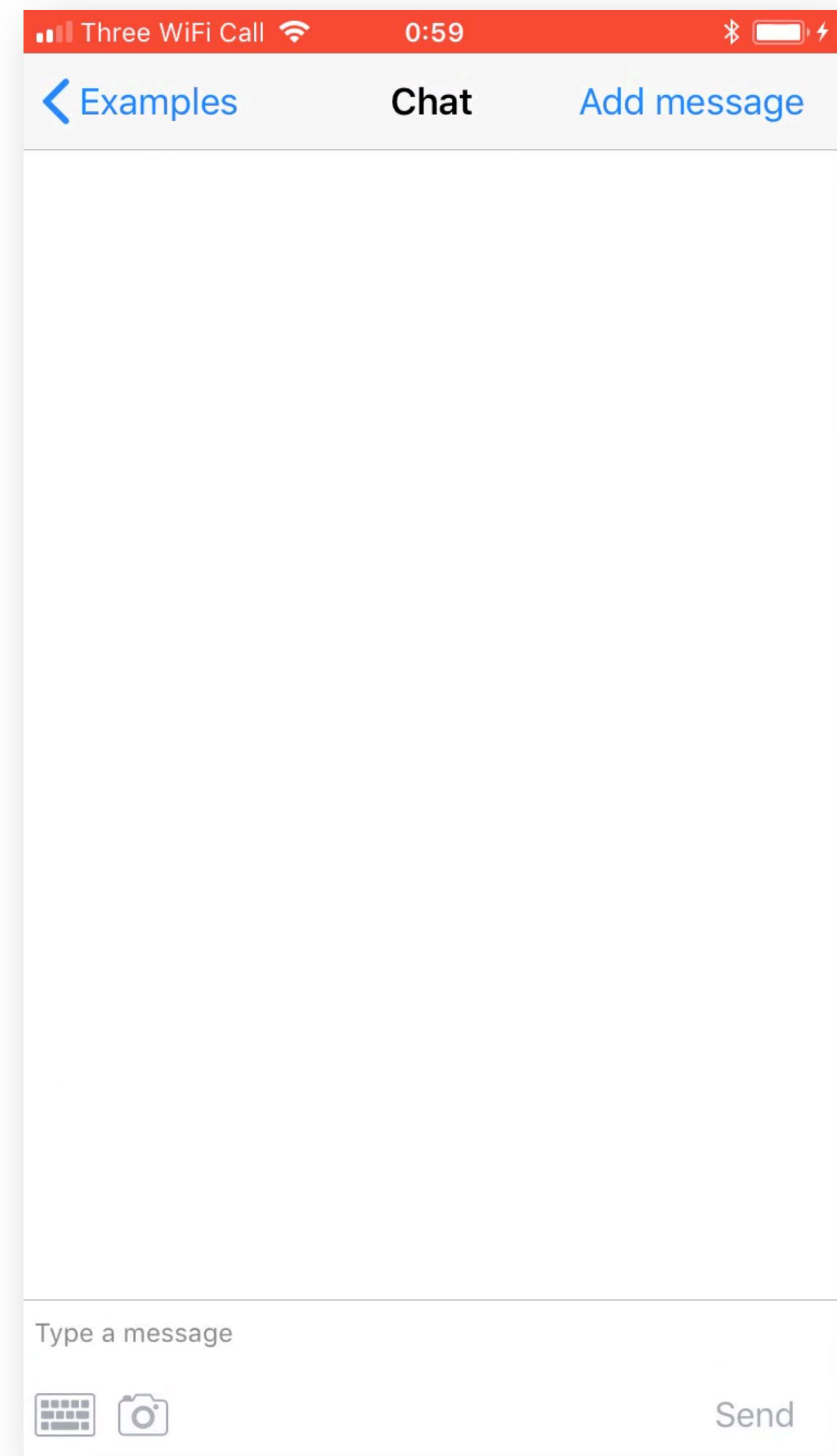
- Как посчитать diff?
 - Chatto - самописный алгоритм
 - не поддерживает дубликаты

Подготовка обновления коллекции

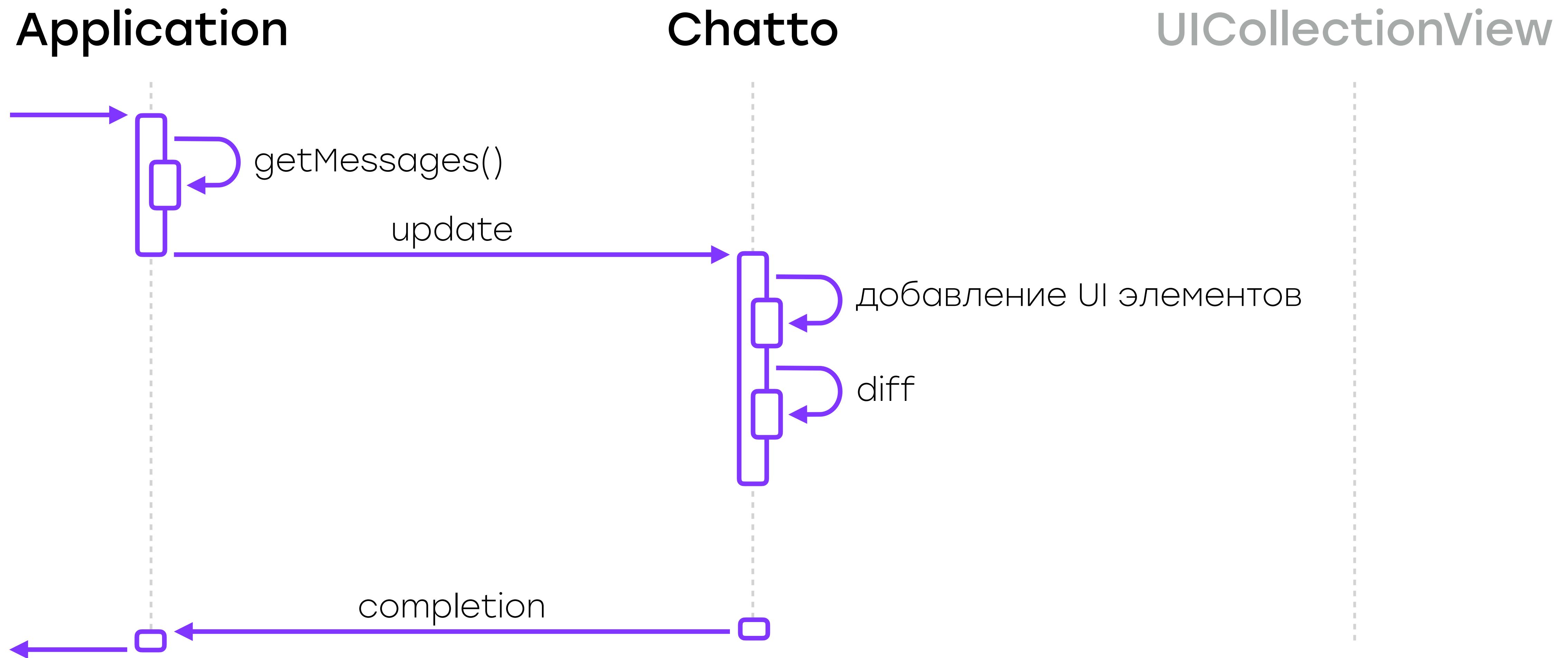
- Как посчитать diff?
 - Chatto - самописный алгоритм
 - не поддерживает дубликаты
 - Алгоритм Paul Heckel'a (используется в IGListKit)
 - <https://gist.github.com/ndarville/3166060>

Подготовка обновления коллекции

- Как посчитать diff?
 - Chatto - самописный алгоритм
 - не поддерживает дубликаты
 - Алгоритм Paul Heckel'a (используется в IGListKit)
 - <https://gist.github.com/ndarville/3166060>
 - Алгоритм Myers'a (GNU diff, git, Swift)
 - <https://www.fivestars.blog/code/swift-5-1-collection-diffing.html>

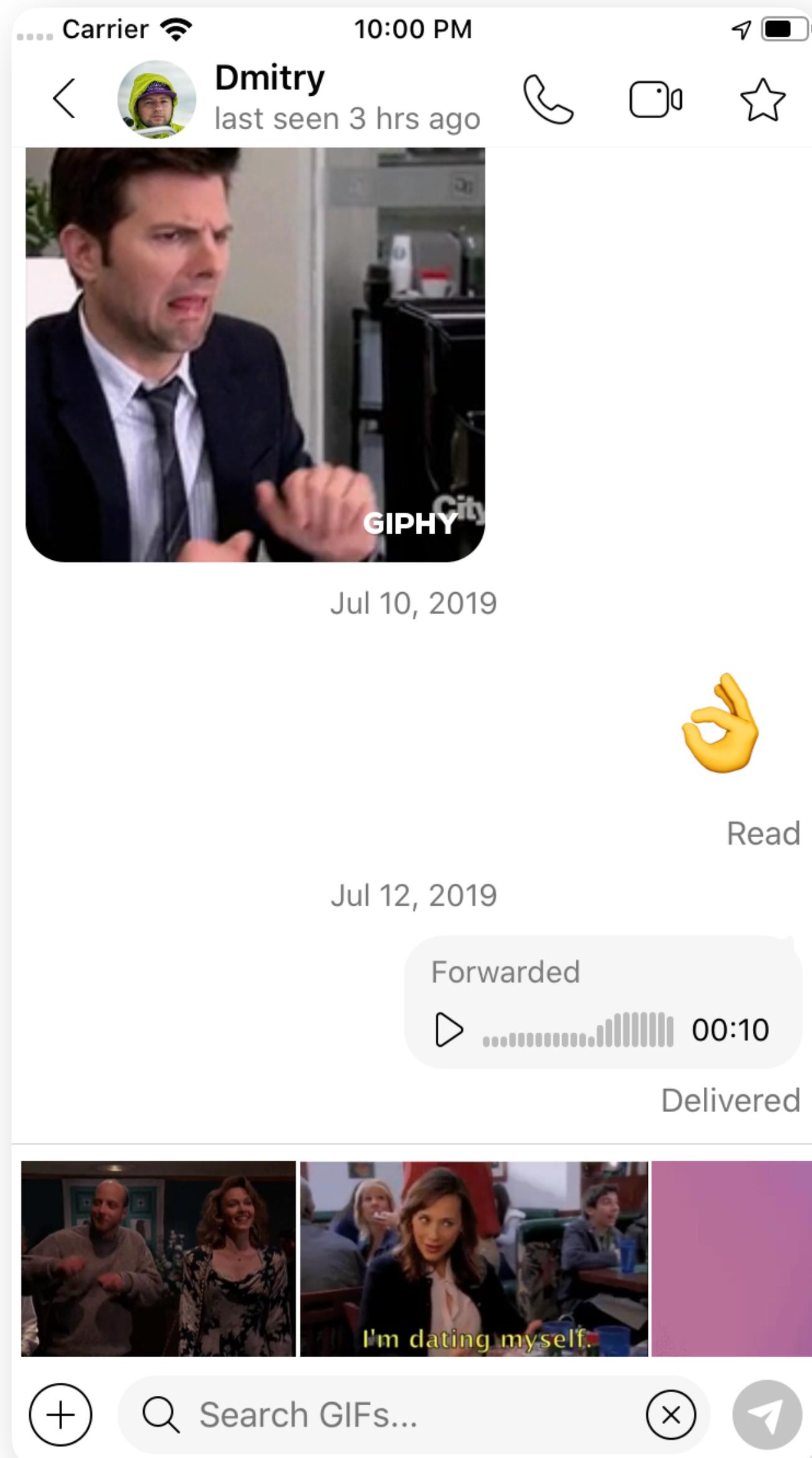


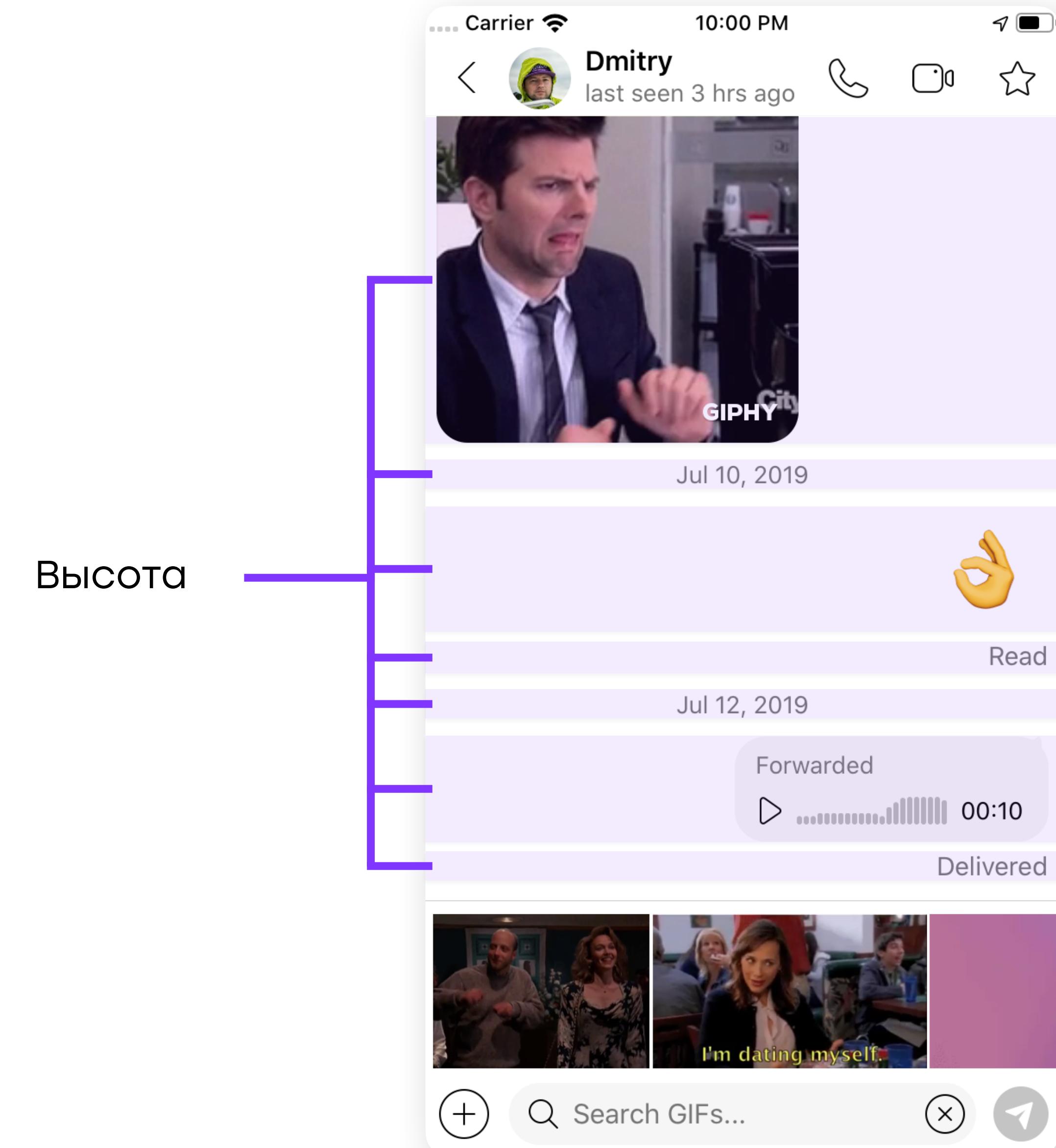
Подготовка обновления коллекции

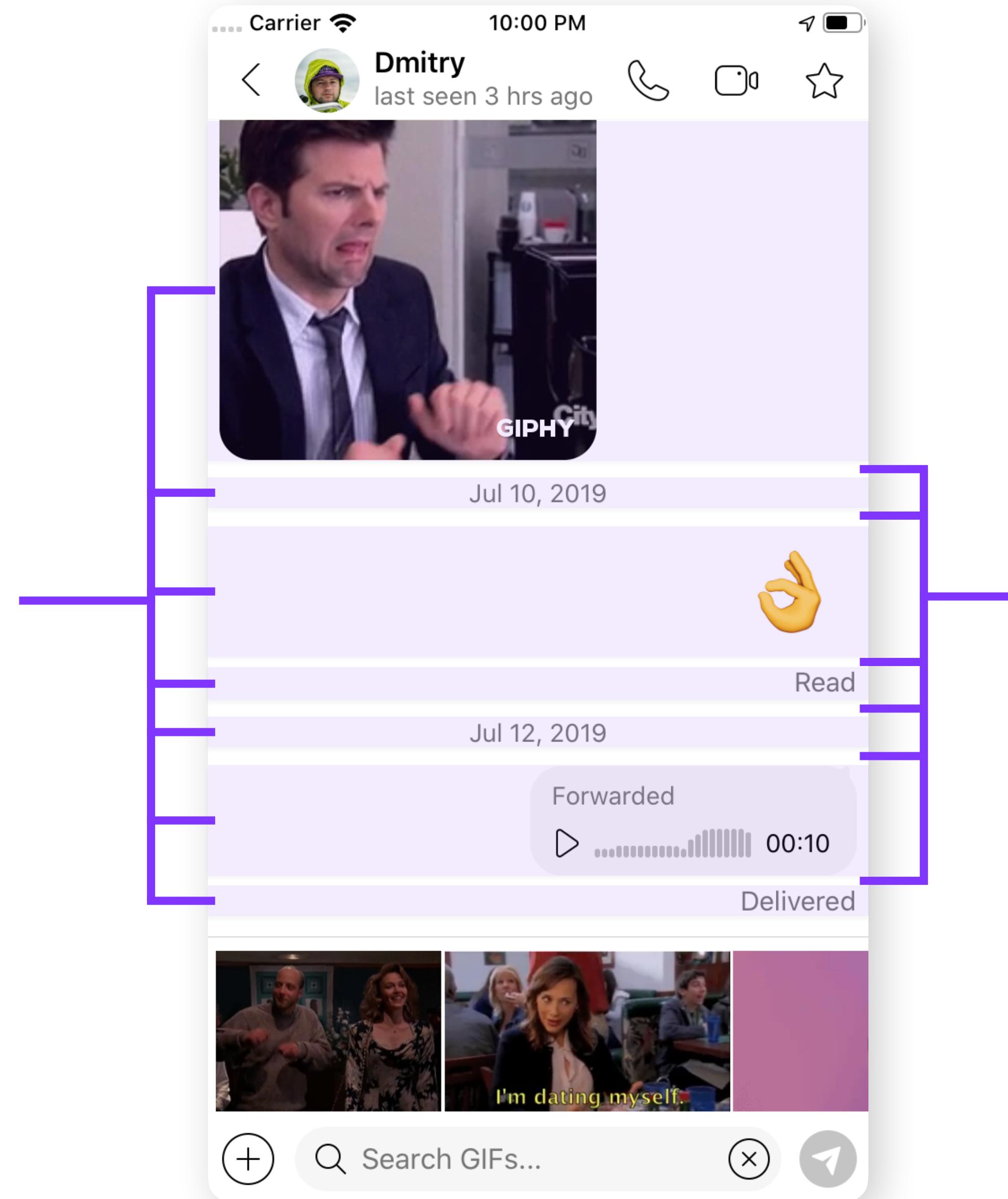


Основные шаги обновления

- Подготовка списка элементов
- Обновление коллекции
- **Обновление layout'a**
- Обновление ячеек







Подготовка layout'a

- Расчет высоты элементов
- Расчет отступов между элементами
- Расчет фрейма для каждого элемента

Расчет высоты элементов

- Расчет в фоновом потоке
 - view не считает layout
 - presenter и view используют один и тот же layout объект

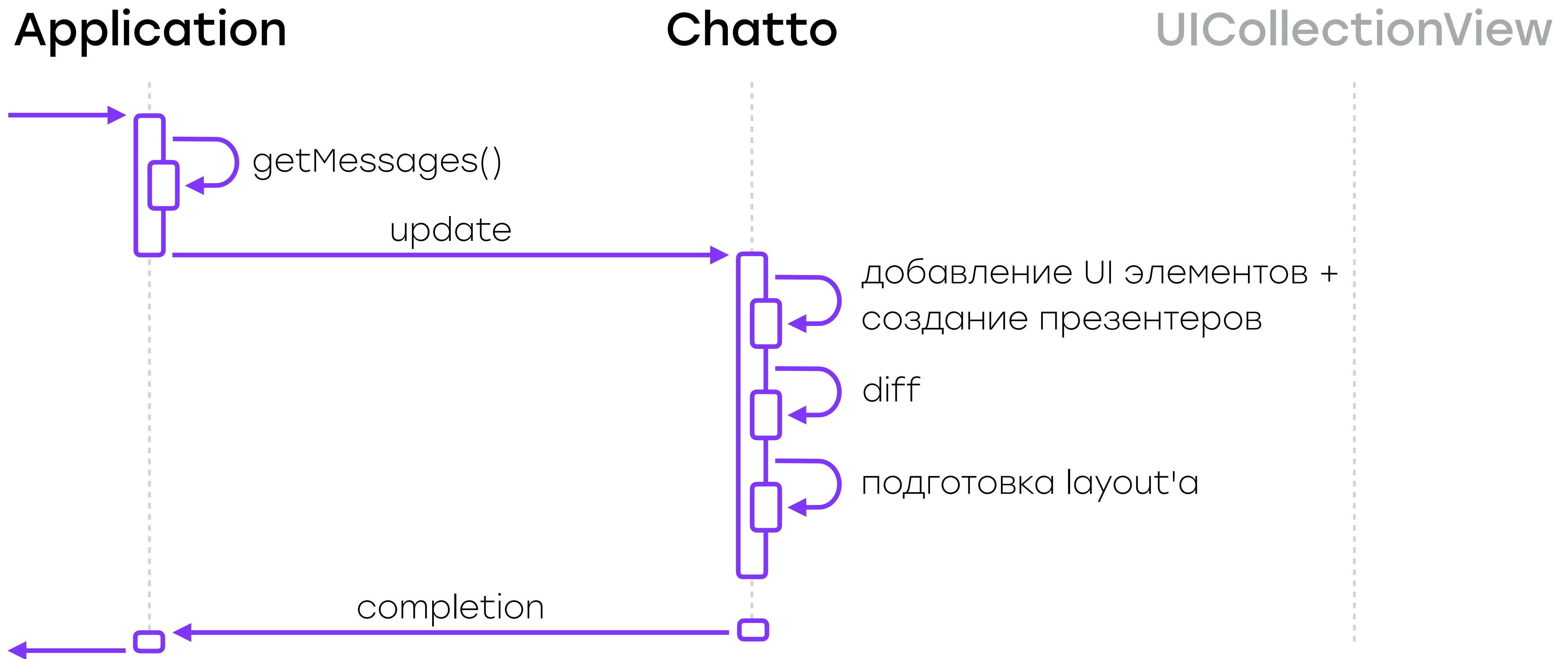
Расчет высоты элементов

- Расчет в фоновом потоке
 - view не считает layout
 - presenter и view используют один и тот же layout объект
- Расчет на главном потоке
 - Auto Layout + sizing cells

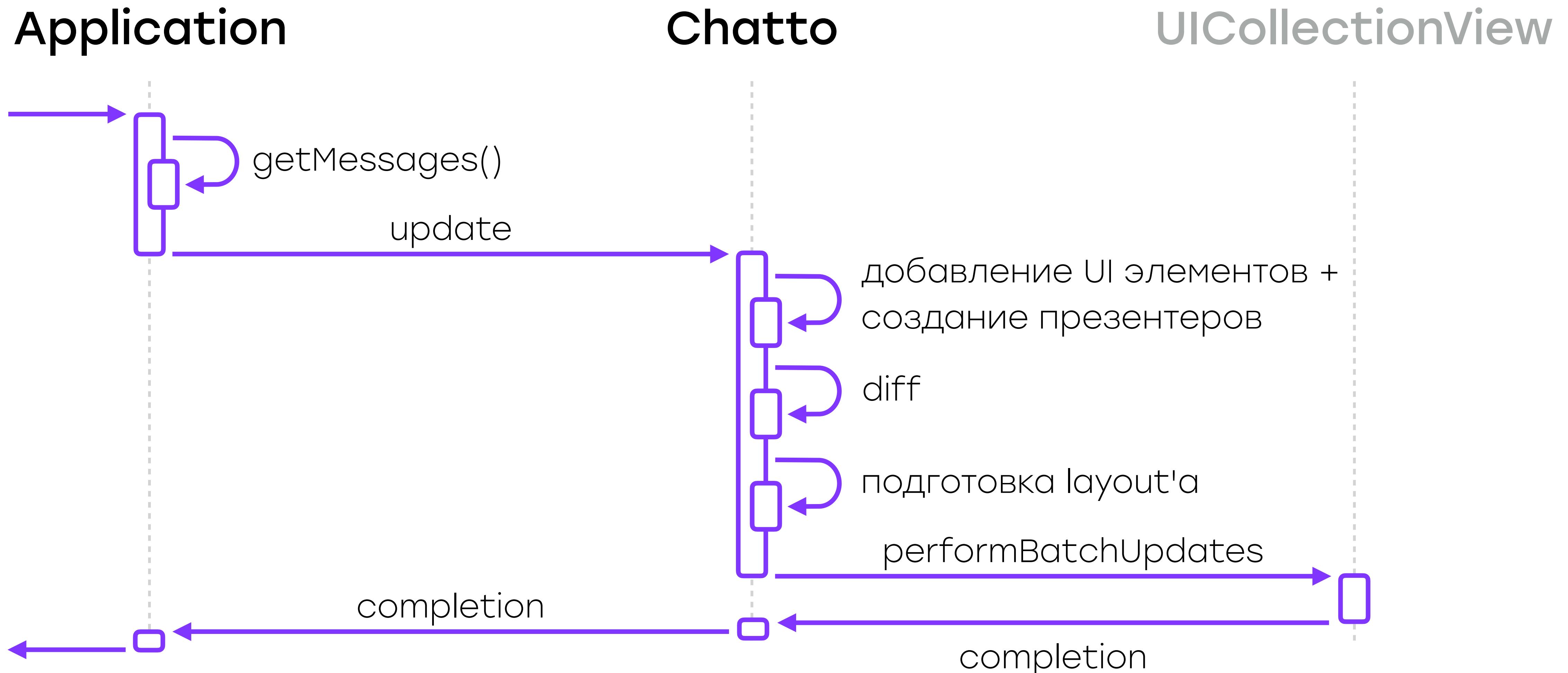
Подготовка layout'a

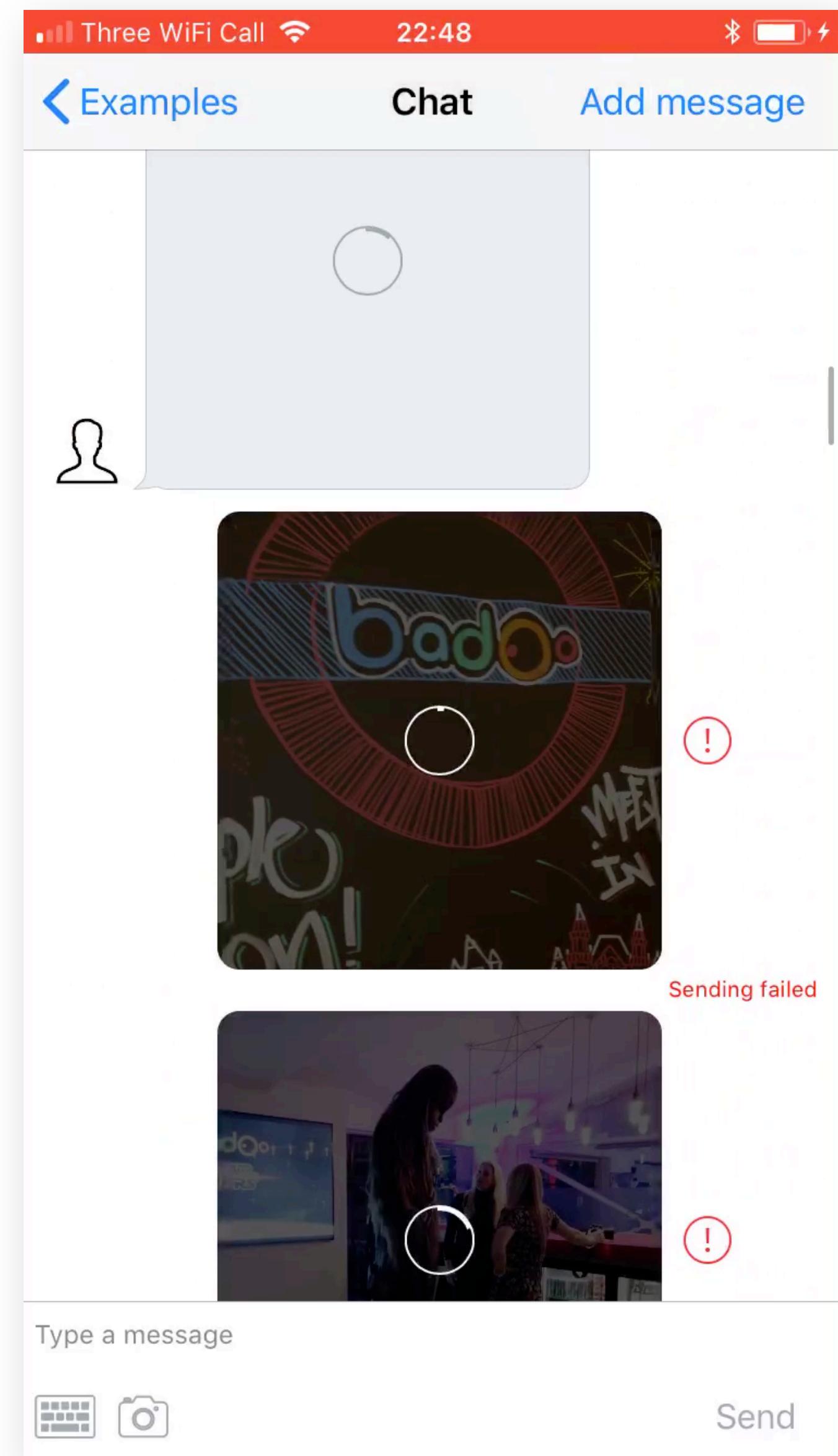
- Советы по оптимизации
 - кэшируйте layout для отдельных элементов
 - не используйте auto layout

Подготовка layout'a



Обновление UICollectionView





Оптимизация UI для сообщений

The quick brown
fox jumps over
the lazy dog\nThe
quick brown fox jumps over the
lazy dogThe quick brown fox
jumps over the lazy
dog\nThe quick brown fox
jumps over the lazy
dog\nThe quick brown fox
jumps over the lazy dog\nThe
quick brown fox jumps over
the lazy dog\nThe
quick brown fox jumps over
the lazy dog\n(t)(h)(e)
(q)(u)(i)(c)(k)(b)(r)(o)(w)(n)(f)(o)(x)
(j)(u)(m)(p)(s)(o)(v)(e)(r)(t)(h)(e)
(l)(a)(z)(y)(d)(o)(g)

"اِيَا اِوَّرِيمَح"

Hi



email@example.com
+7 (999) 999-99-99



Что использовать для отрисовки текста?

UILabel

UITextView

Рисует текст

+

+

Находит данные
в тексте

-

+

Считает размер
в фоне

-

-

Что использовать для отрисовки текста?

	UILabel	UITextView	TextKit + NSDataDetector
Рисует текст	+	+	+
Находит данные в тексте	-	+	+
Считает размер в фоне	-	-	+

Что использовать для отрисовки текста?

	UILabel	UITextView	TextKit + NSDataDetector
Рисует текст	+	+	+
Находит данные в тексте	-	+	+
Считает размер в фоне	-	-	+

Нахождение данных

- **isSelectable** выключает выделение текста И! нахождение данных

This is a TextMessageCollectionViewCell. It uses UITextView with data detectors so you can interact with urls: <https://github.com/badoo/Chatto>, phone numbers: 0740000000, dates: 3 jan 2016 and others

Along with Chatto there's ChattoAdditions, with bubbles and the input component

It calculates sizes in the background for smooth pagination and rotation, and it can deal with thousands of messages with a sliding data source



Нахождение данных

- **isSelectable** выключает выделение текста **И!** нахождение данных
- Решение
 - фильтровать gesture recognizers
 - ломалось каждый год

This is a TextMessageCollectionViewCell. It uses UITextView with data detectors so you can interact with urls: <https://github.com/badoo/Chatto>, phone numbers: 0740000000, dates: 3 jan 2016 and others

Along with Chatto there's ChattoAdditions, with bubbles and the input component

It calculates sizes in the background for smooth pagination and rotation, and it can deal with thousands of messages with a sliding data source



Расчет размеров текста

- `UITextView.sizeThatFits(_:)`
 - использует TextKit
 - работает только на главном потоке
 - требует наличие экземпляра UITextView

Расчет размеров текста

- Использовать TextKit
 - Настройки должны совпадать с UITextView
 - `NSTextContainer.lineFragmentPadding = 0`

Расчет размеров текста

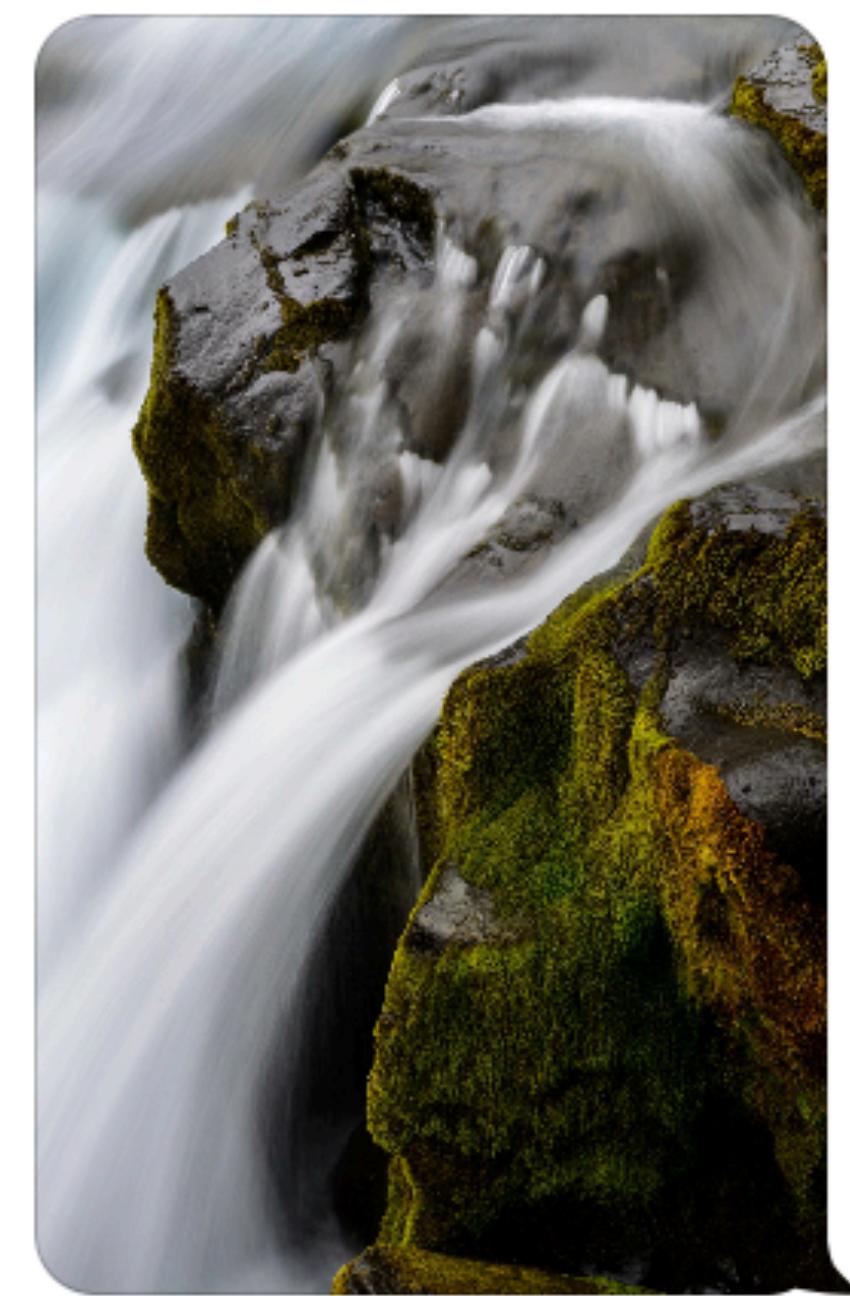
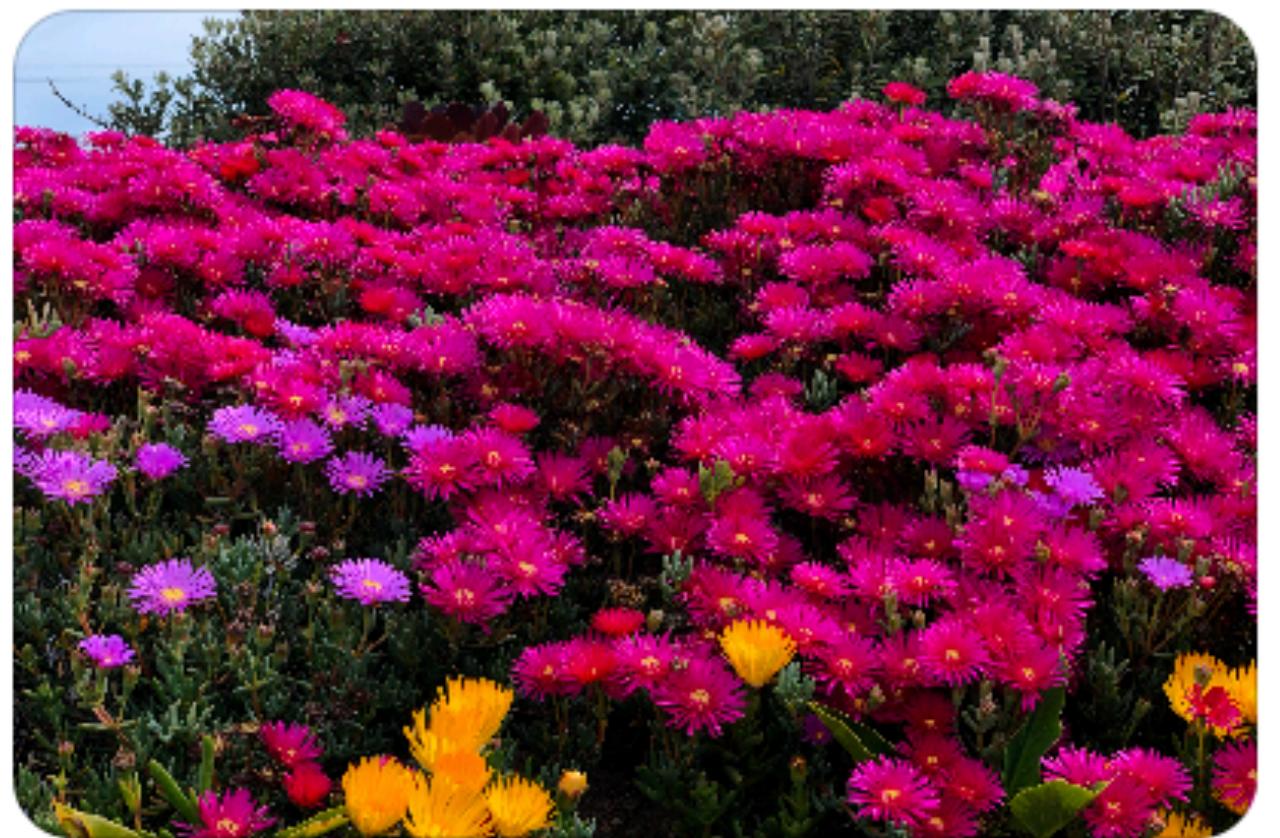
- Использовать TextKit
 - Настройки должны совпадать с UITextView
 - `NSTextContainer.lineFragmentPadding = 0`
 - Поддержка отсутствующих символов в своих шрифтах
 - `NSAttributedString.Key(rawValue: "NSOriginalFont")`: <ваш шрифт>

Оптимизации отрисовки UITextView

- `contentOffset` - не нужен
- `selectedRange` - не нужен

Какие есть альтернативы?

- Считать **и** рисовать текст в фоне
 - **CGContext + TextKit**
- Примеры
 - **CKTextKitRenderer (ComponentKit)**
 - **ASTextKitRenderer (AsyncDisplayKit)**



Как посчитать размер фото?

Проблема:

- Сообщение содержит только URL
- Фото в кэше отсутствует

Как посчитать размер фото?

Проблема:

- Сообщение содержит только URL
- Фото в кэше отсутствует

Решение:

- Сообщение должно содержать размер фото

Как посчитать размер фото?

Проблема:

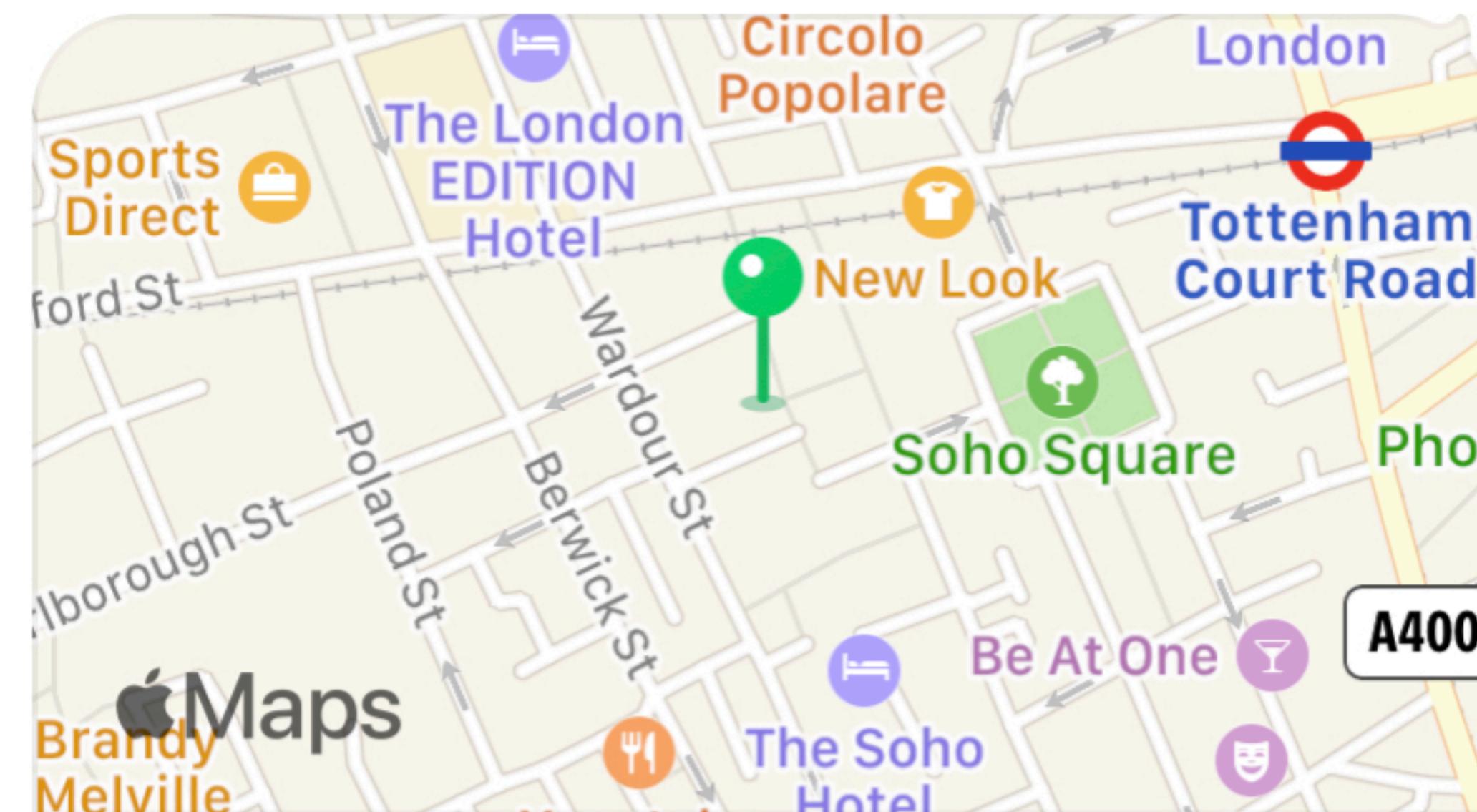
- Сообщение содержит только URL
- Фото в кэше отсутствует

Решение:

- Сообщение должно содержать размер фото
- Все фото одной высоты

Оптимизация отрисовки фото

- Загружайте изображение минимального размера
 - Сервер присыпает URL с шаблоном внутри:
 - `.../photo?id=xxx&size=__size__`
 - Клиент подставляет размер в реальных пикселях
 - Делайте downsampling и decoding в фоне



Great Chapel Street, London
England, W1F 8FL, United Kingdom

Как показать карту?

- MKMapView
 - потребляет много памяти
 - лаги при скролле

Как показать карту?

- MKMapView
 - потребляет много памяти
 - лаги при скролле
- MKMapSnapshotter
 - рисует карту в фоне
 - не поддерживает overlays и annotations

Заключение

Производительный чат

- Используйте UICollectionView
- Разгружайте главный поток
- Оптимизируйте отрисовку UI

Спасибо!

- Chatto
- A Tour of UICollectionView
- Image and Graphics Best Practices