

**Rubius / DP Labs**

**Кросс-платформенная  
мобильная разработка**



# Артур Дробинский

Ведущий архитектор  
команды DP Labs

[arturdr.ru](http://arturdr.ru)

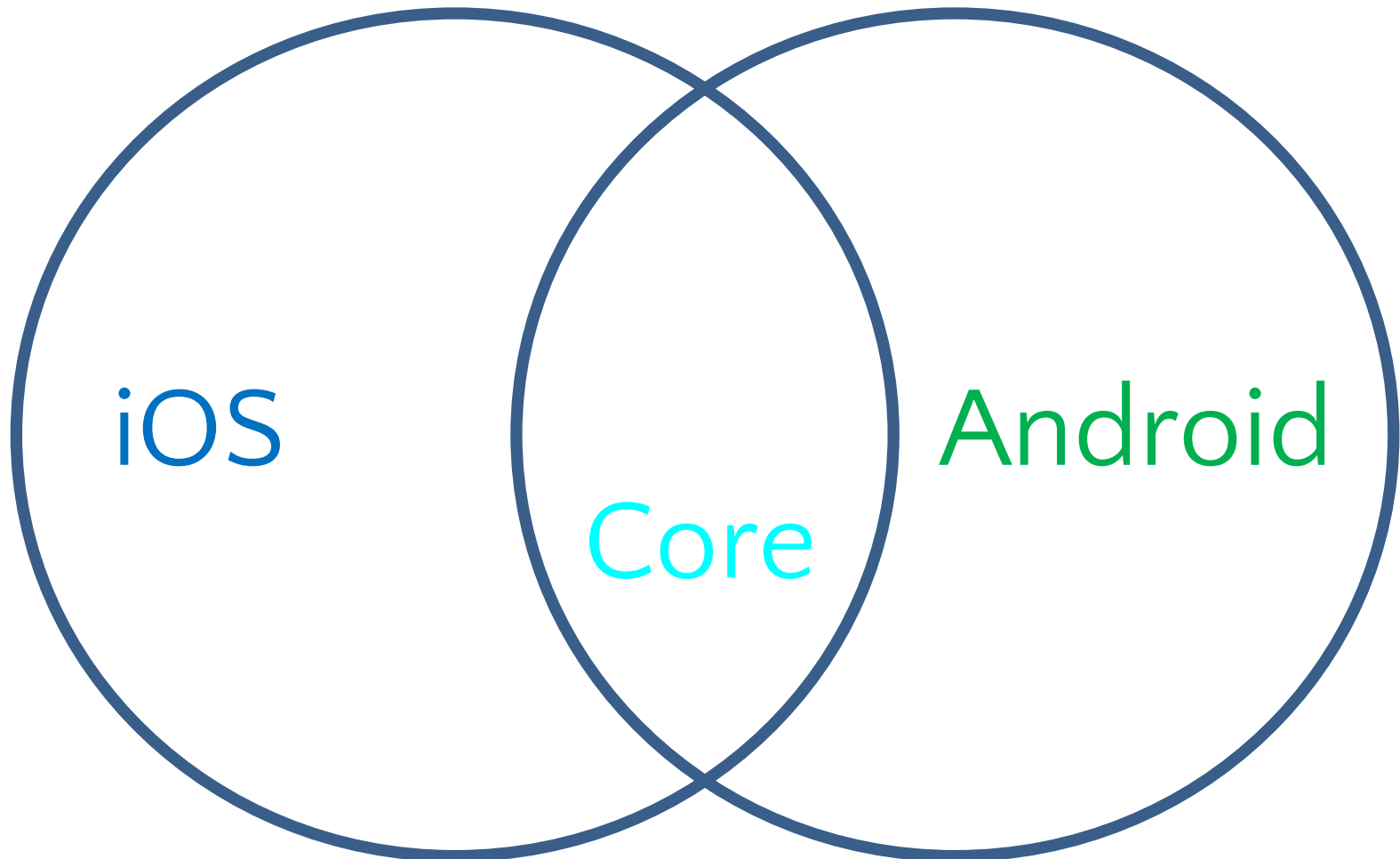
[artur.drobinskiy@arturdr.ru](mailto:artur.drobinskiy@arturdr.ru)

Зачем кросс-  
платформенность?

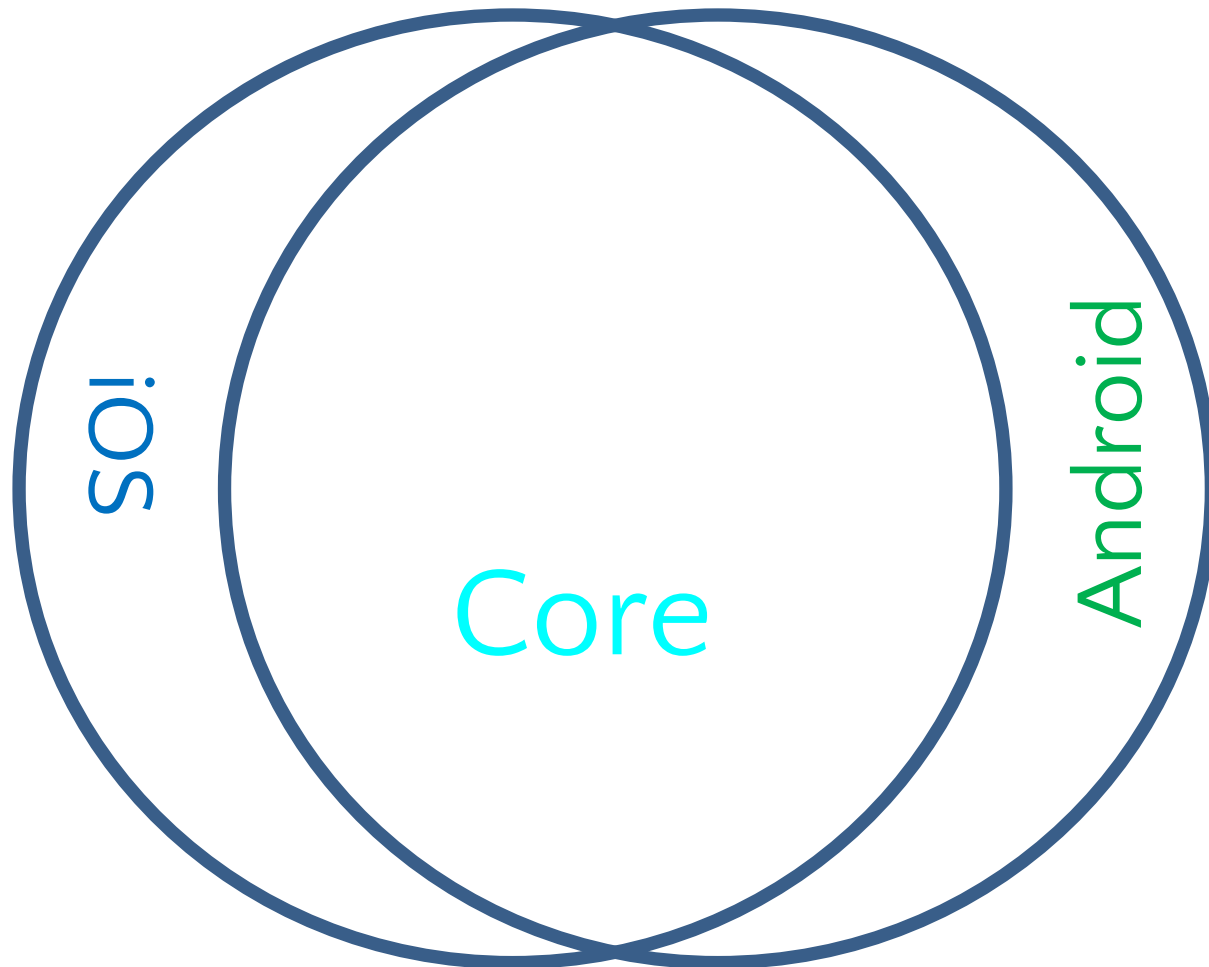
# Везде используем универсальный инструмент

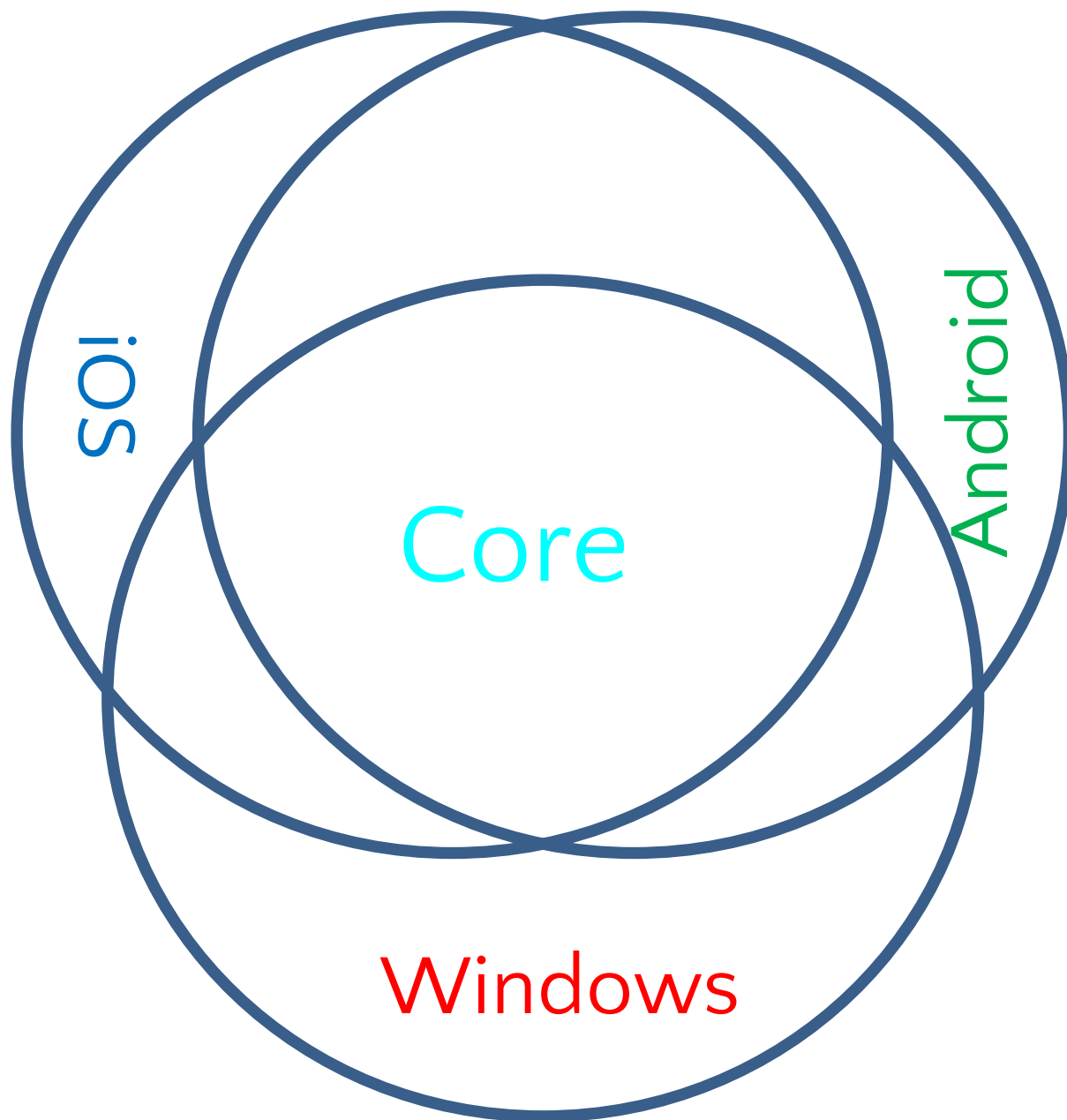


# Повторное использование кода



# Повторное использование кода





Проще поддержка – исправляем баги  
один раз



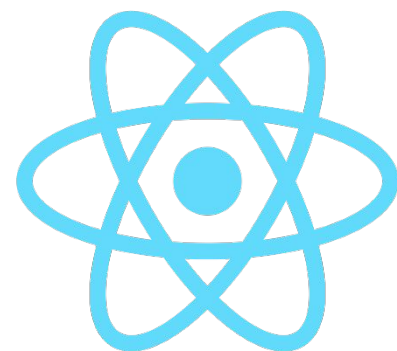
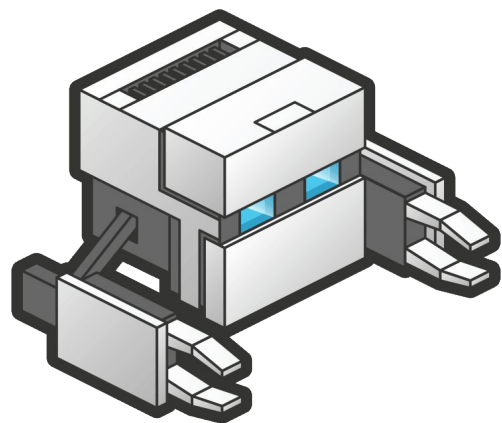


Дешевле

!

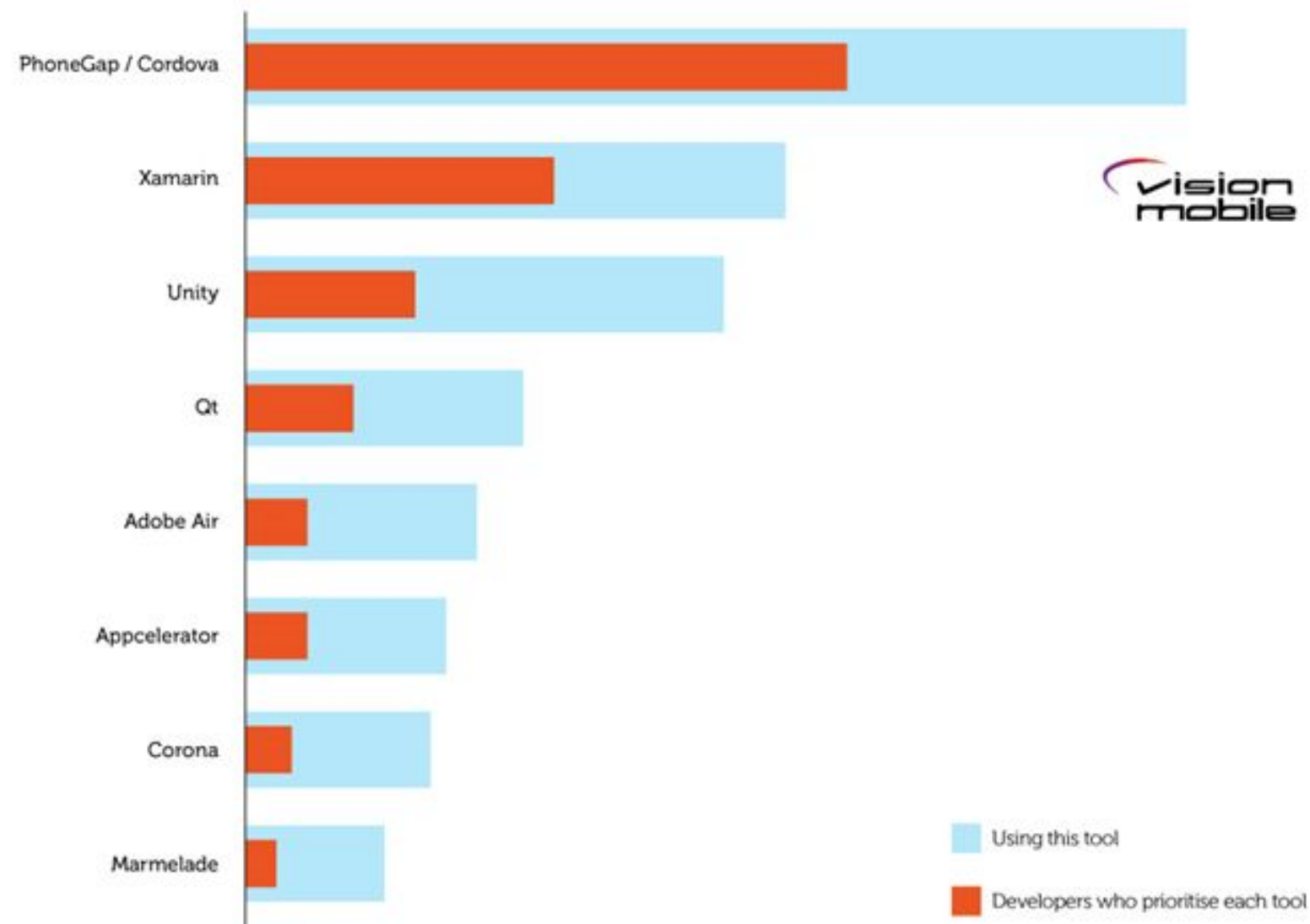


# Кросс-платформенные решения



# THE TOP 3 CROSS PLATFORM TOOLS ACCOUNT FOR 70% OF PRIMARY USE

% of developers using cross-platform tools by primary tool and tools used (n=1,664)



How does it work?

Cross-platform code

Native .apk / .ipa / .exe (application package)

Cross-platform code

Native .apk / .ipa / .exe (application package)

UI, Native controls

Native API

Cross-platform code

Native .apk / .ipa / .exe (application package)

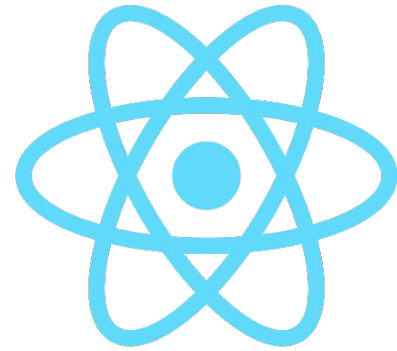
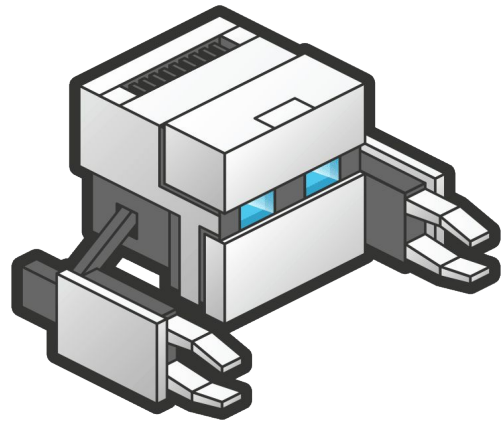
UI, Native controls

Native API

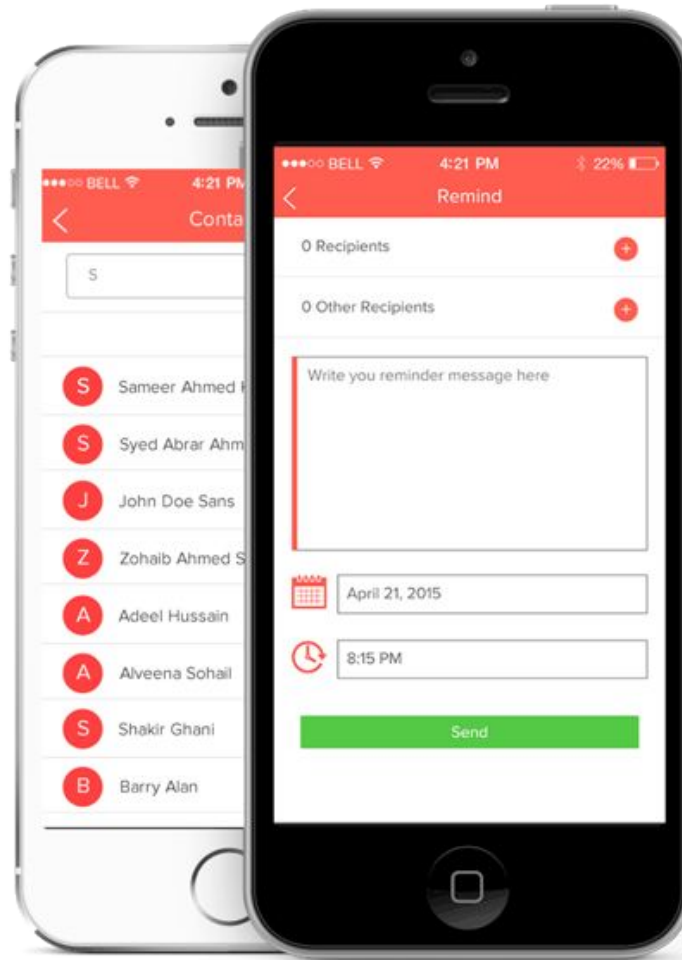
API Bridge

Cross-platform code





# PhoneGap



# PhoneGap

Native .apk / .ipa / .exe (application package)

WebView

UI: HTML

Code: JS

Native API

Limited API Bridge

# Qt



# Qt/Unity

Native .apk / .ipa / .exe (application package)

View (Canvas)

UI: QML/Game Engine

Code: C++/C#

Native API

Limited API Bridge

# Xamari

n

Native .apk / .ipa / .exe (application package)

Native Controls

Native API

API Bridge

Code: C#

# React Native

Native .apk / .ipa / .exe (application package)

Native Controls

API Bridge

JSX

Code: V8, JS

Native API

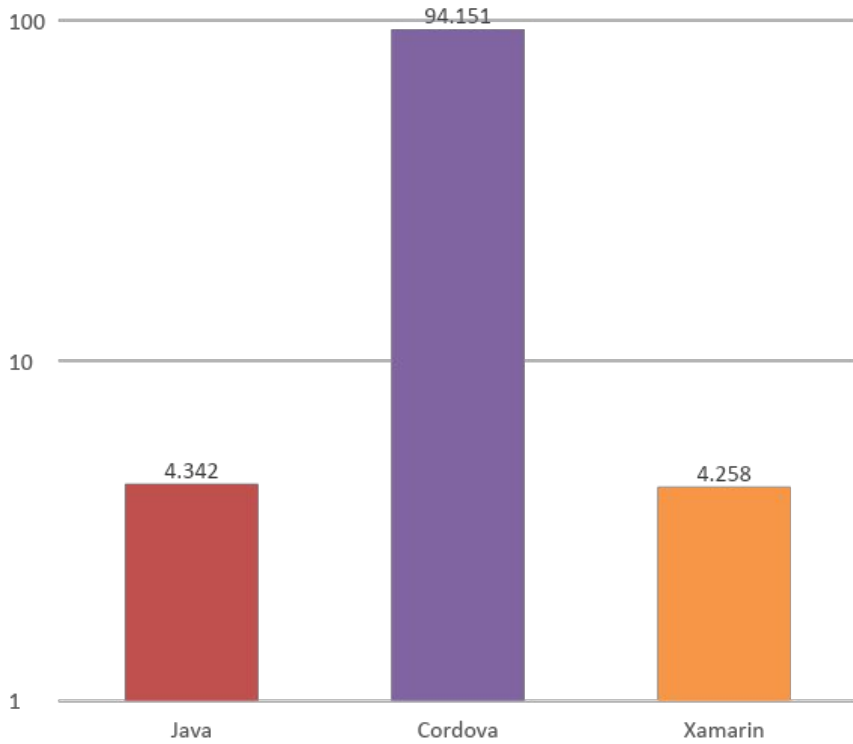
Limited API Bridge

Performance

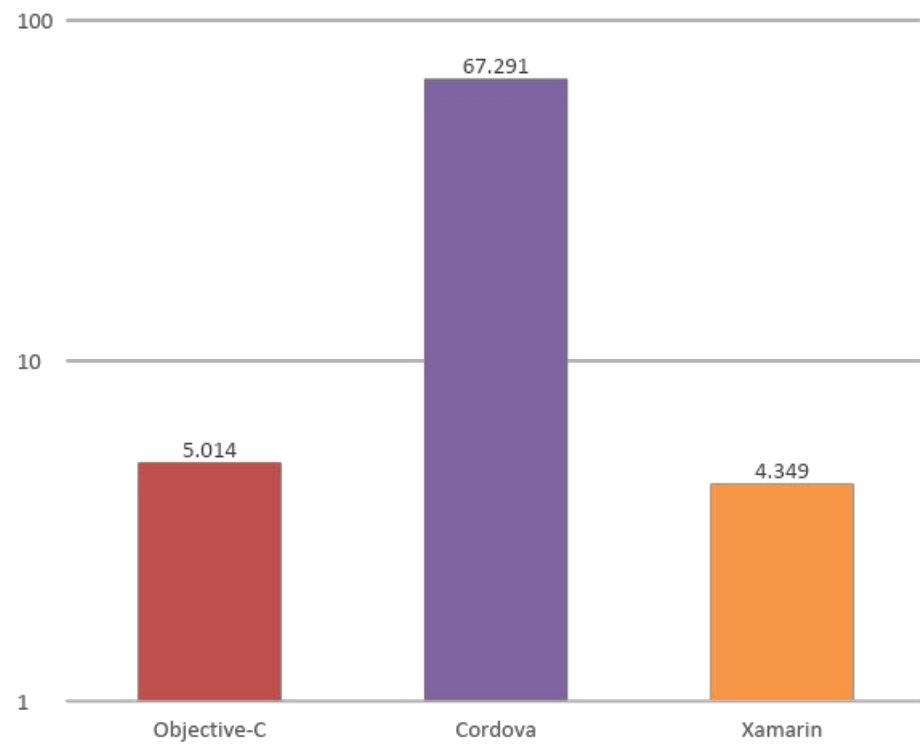


# Математические ВЫЧИСЛЕНИЯ

Android

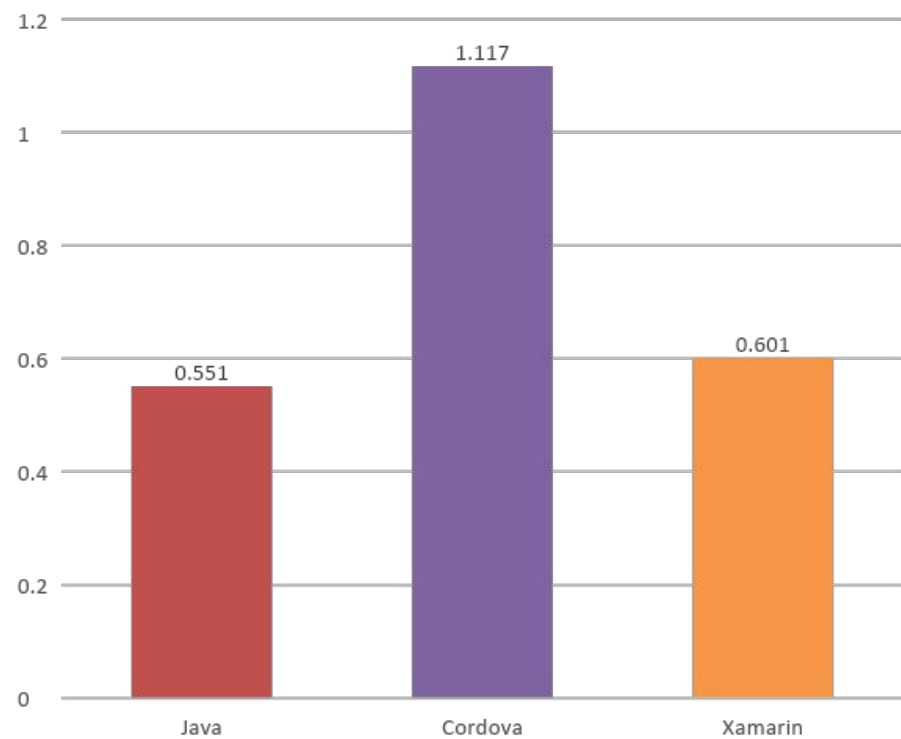


iOS

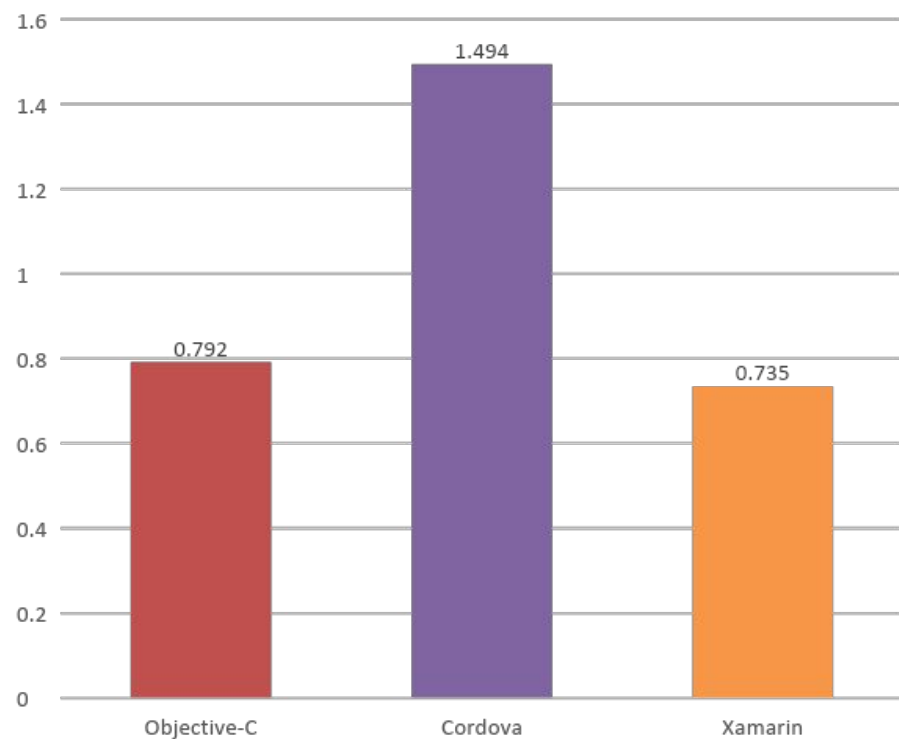


# SQLite: чтение данных

Android

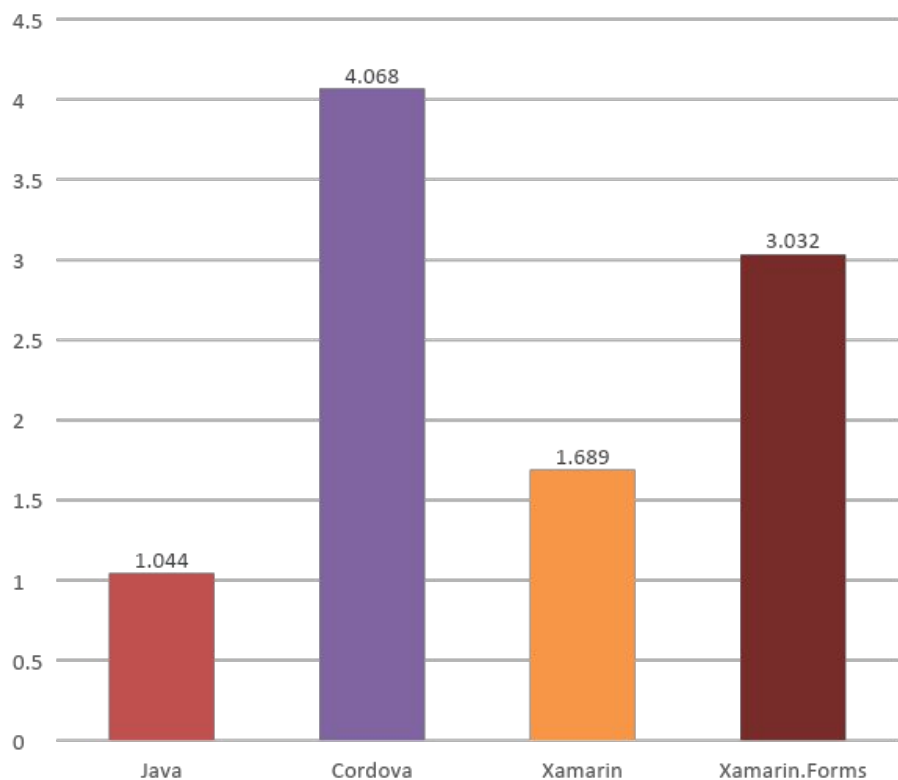


iOS

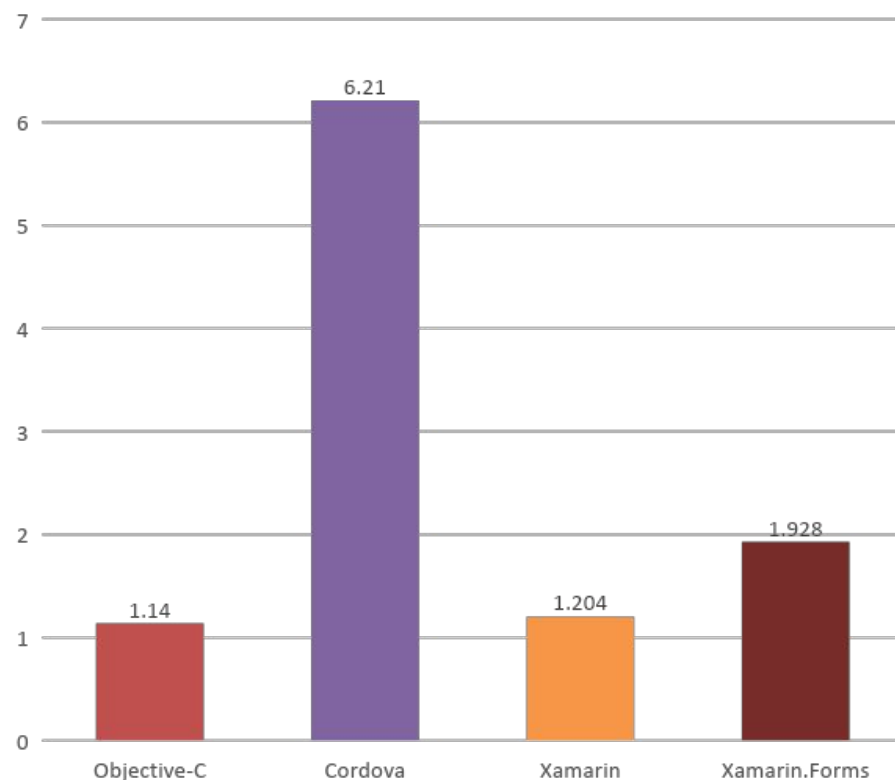


# Запуск приложения

Android

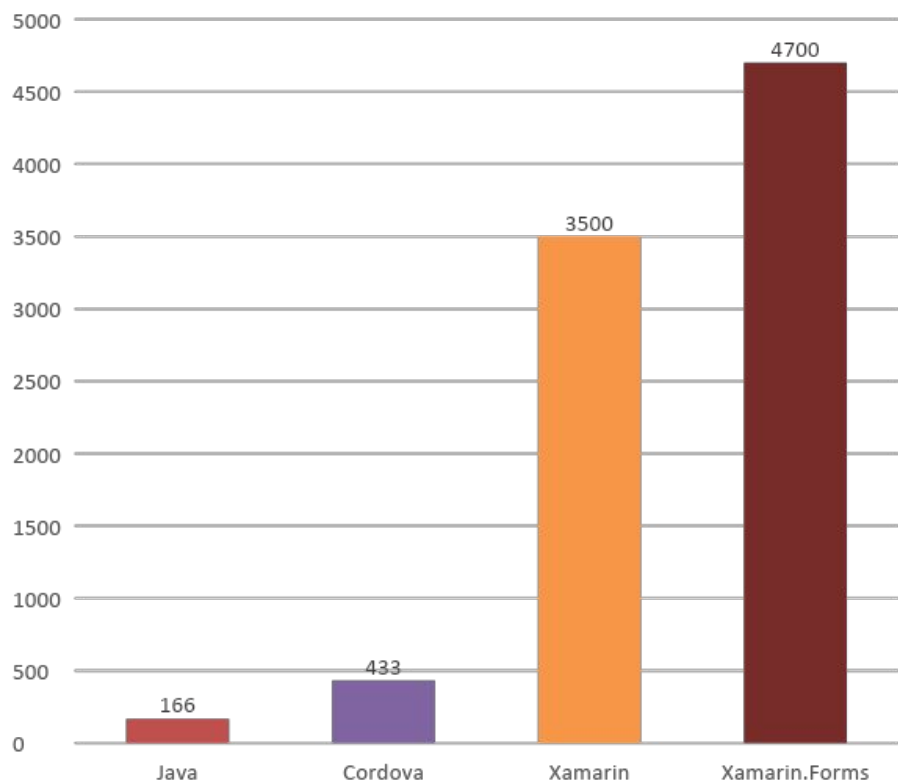


iOS

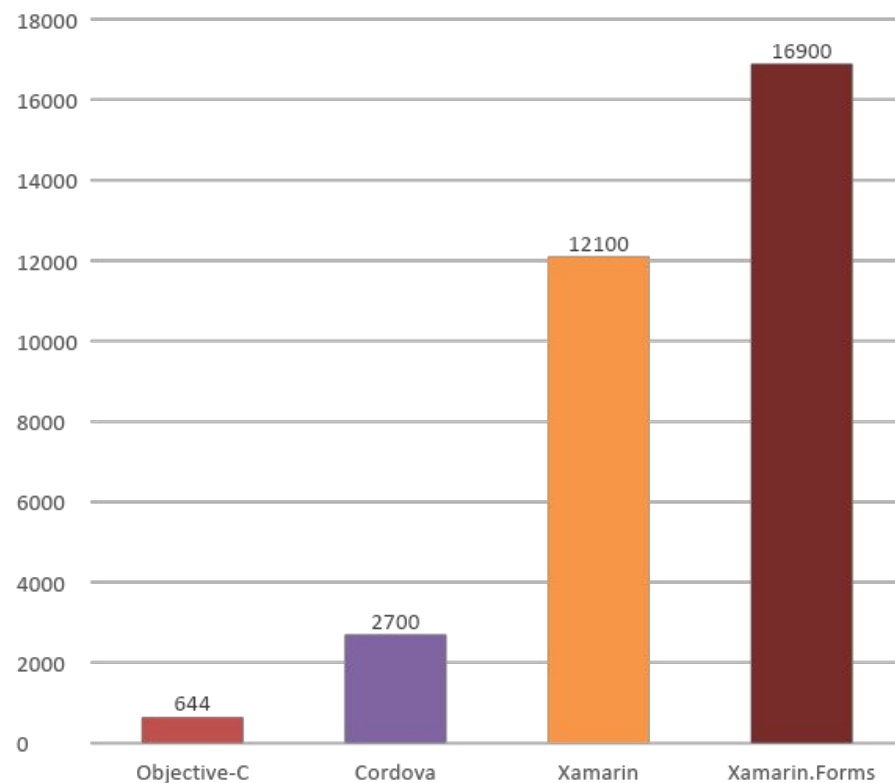


# Размер приложения

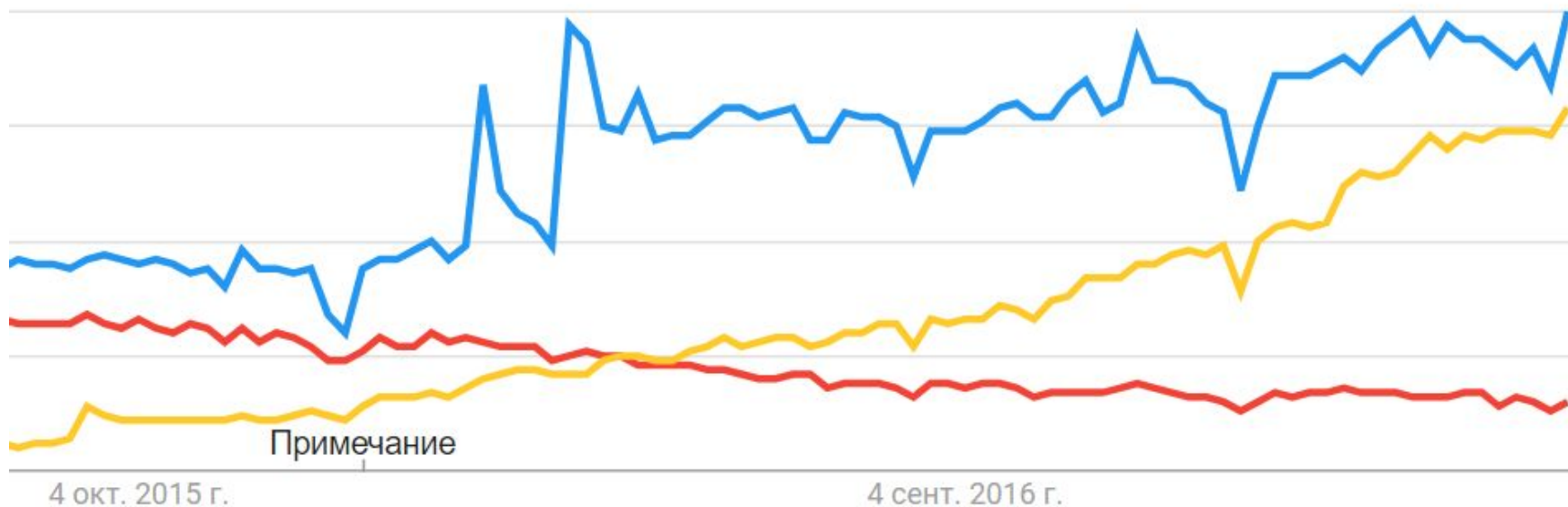
Android



iOS

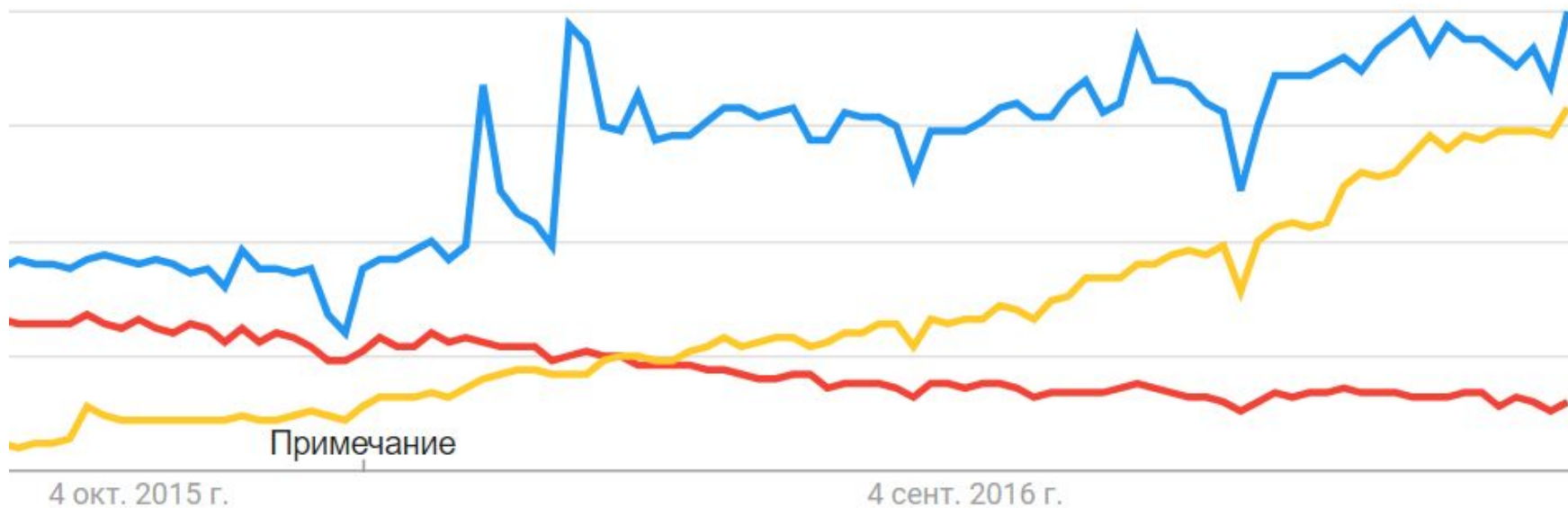


# Xamarin React Native Phonegap

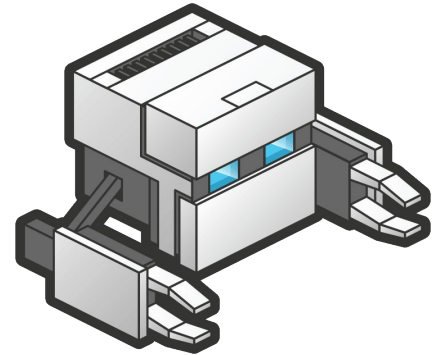


Примечание

# Xamarin   React Native   Phonegap



Примечание



DEMO



# React Native

# Плюс ы

- Hype
- Hot reloading + простое обновление приложений
- Javascript
- Встраивание в нативные приложения

# Минусы?

- Breaking Changes
- Second class android support
- Javascript & TypeSafety
- Animations are slow (bridge)
- Неполная поддержка нативных контролов, ненативная разметка

# Xamarin



Native iOS App

Platform-specific C#



Native Android app

Platform-specific C#



Native Windows app

Platform-specific C#

Shared C# App Logic

# Xamarin Forms React Native



Native iOS App



Native Android app



Native Windows Phone app

Shared C# User Interface Code

Shared C# App Logic

# Хamarin

## Мифы или реальность?

Утечки  
памяти



и особенности Garbage Collection

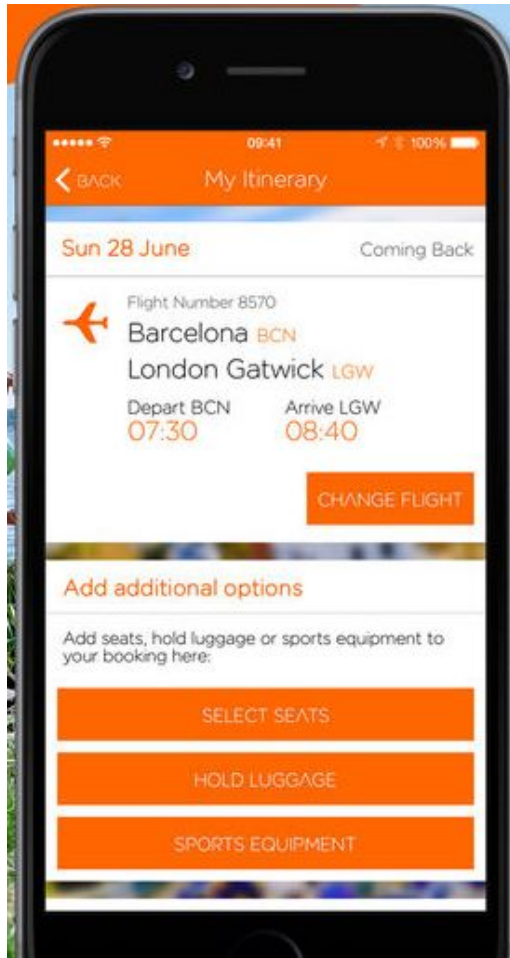
# Использование open-source КОНТРОЛОВ



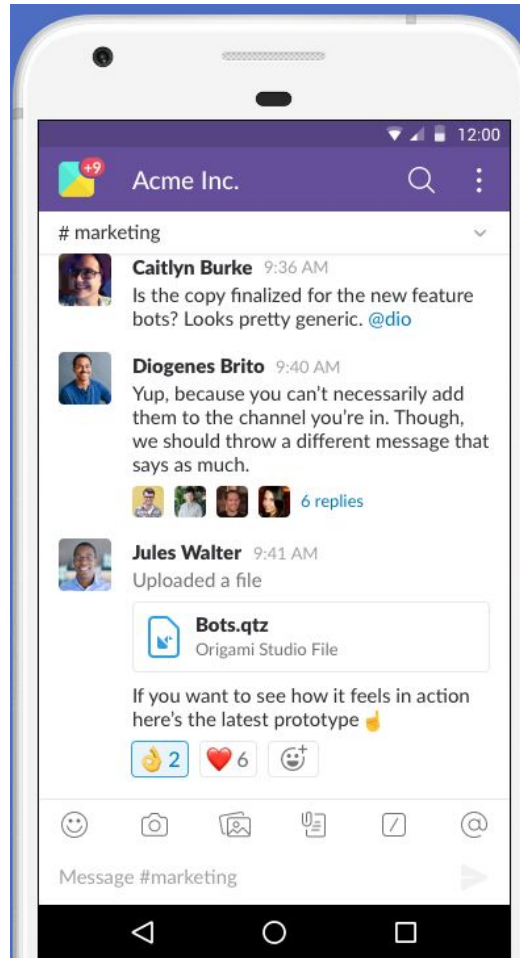
Необходимо написание собственных API  
Bridge

На Xamarin нет приложений

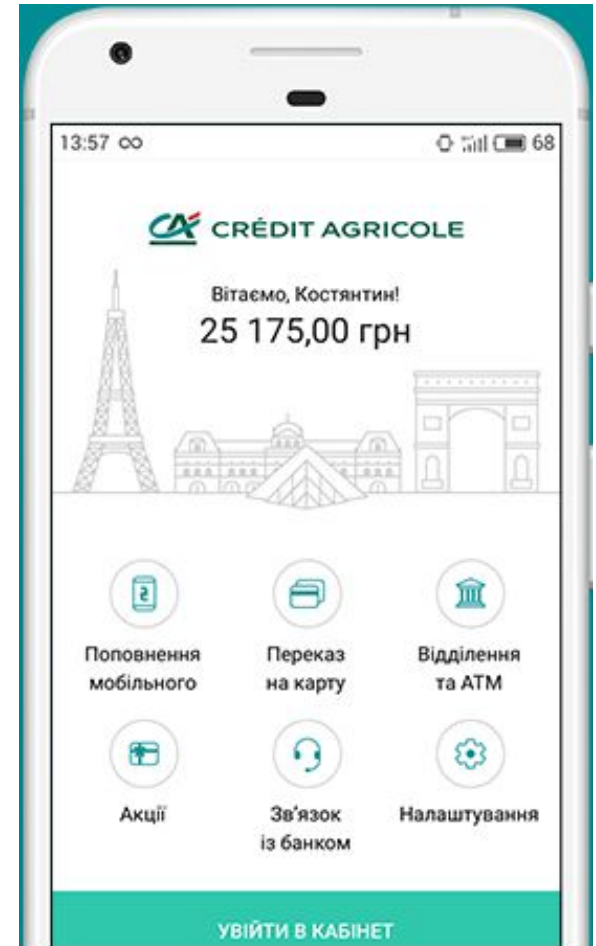




easyJet



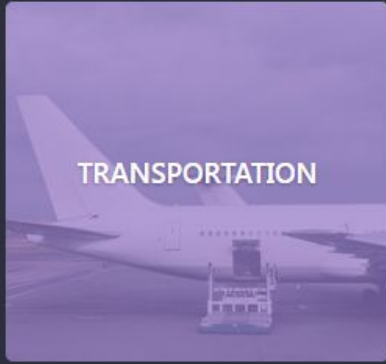
Slack



Credit Agricole



React Native 🤗



<https://www.xamarin.com/customers/>

# Минусы Xamarin (и ReactNative)

- Размер приложения (6MB+)
- Больше время запуска приложения
- Использование open-sourced controls требует поиска/написания обёртки

# Xamarin vs Native

# Xamarin

- Сложные Enterprise приложения
- Существующая команда C# разработчиков
- Бэкенд на .NET
- Похожие экраны/workflow на разных устройствах

# Native

- Дублирование бизнес-логики – не проблема (её мало)
- Команда с опытом на Obj-C и Java
- Бэкенд на Java 😊
- Workflow на Android/iOS радикально различный

# Xamarin vs React Native

# Xamarin

- Сложная бизнес логика
- C#-команда
- Использование всех нативных возможностей
- Компиляция, производительность
- UWP + Mac (+ Web)
- XAML + Native rendering + Flexbox

# React Native

- UI-heavy apps
- JS-команда
- Встраивание в нативные приложения
- Быстрое обновление (AppStore, Google Play)
- Задержка с новыми фичами на платформах (+ Web)
- Flexbox (but only flexbox)



# Вопросы?

<http://arturdr.ru>

[artur.drobinskiy@arturdr.ru](mailto:artur.drobinskiy@arturdr.ru)

```

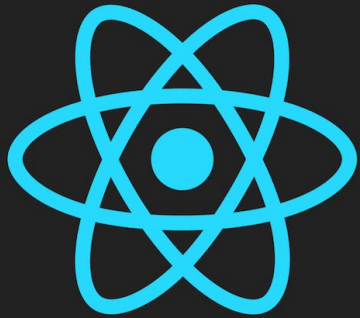
<ListView ItemsSource="{Binding MonkeysGrouped}"
ItemTapped="Handle_ItemTapped" ItemSelected="Handle_ItemSelected"
    HasUnevenRows="true" GroupShortNameBinding = "{Binding Key}"
    IsGroupingEnabled = "true" GroupDisplayBinding = "{Binding Key}">
<ListView.ItemTemplate>
    <DataTemplate>
        <ViewCell>
            <Grid Padding="10" RowSpacing="10" ColumnSpacing="10">
                <Grid.RowDefinitions>
                    <RowDefinition Height="Auto"/>
                    <RowDefinition Height="*/>
                </Grid.RowDefinitions>
                <Grid.ColumnDefinitions>
                    <ColumnDefinition Width="Auto"/>
                    <ColumnDefinition Width="*/>
                </Grid.ColumnDefinitions>

                <controls:CircleImage BorderColor="Aqua" BorderThickness="3"
HeightRequest="66"
HorizontalOptions="CenterAndExpand" Vertical`Options="CenterAndExpand"
Aspect="AspectFill" WidthRequest="66" Grid.RowSpan="2" Source="{Binding Image}"/>
                <Label Grid.Column="1" Text="{Binding Name}"
VerticalOptions="End"/>

                <Label Grid.Column="1"
                    Grid.Row="1"
                    VerticalOptions="Start"
                    Text="{Binding Location}"/>
            </Grid>
        </ViewCell>
    </DataTemplate>
</ListView.ItemTemplate>
</ListView>

```

```
<View style={[styles.container, {backgroundColor:
this.props.backgroundColor}]}>
  <View style={styles.labelContainer}>
    <Text style={labelStyles}> {this.state.maxValue} </Text>
  </View>
  <View style={styles.itemsContainer}>
    <View style={[styles.polygonContainer, {borderColor:
this.props.borderColor}]}>
      {this.renderBars()}
    </View>
    <View style={styles.itemsLabelContainer}>
      {this.renderLabels()}
    </View>
  </View>
</View>
```

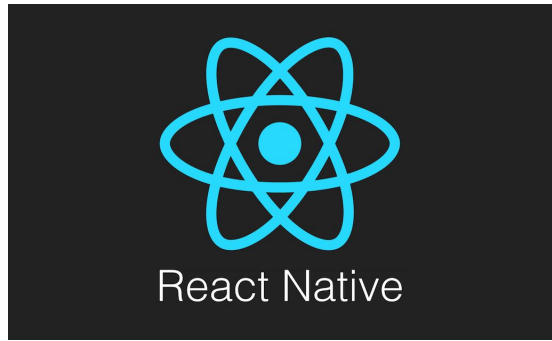


React Native

=



Xamarin.Forms



=



Xamarin.Forms  
s

>

