

# 浅谈跨平台方案的适用场景

主讲人：\_\_\_ 郑鹏 \_\_\_



郑鹏

LeanCloud 开发工程师

目录

1. 跨平台开发简介
2. React Native
3. Flutter
4. 总结

# 什么是跨平台开发？

Write once, run anywhere

## 主流方案的标语

**React Native** - Learn once, write anywhere.

**Flutter** - Flutter allows you to build beautiful native apps on iOS and Android from a single codebase.

**Xamarin** - Deliver native Android, iOS, and Windows apps with a single shared .NET code base.

... ..

## 主流方案的实现原理

### 方式 1 - 嫁接 Native Tool Kit

用某种方式封装各平台的 Native API,  
然后再以另一种形式 API 提供给开发者。  
代表性方案：React Native。

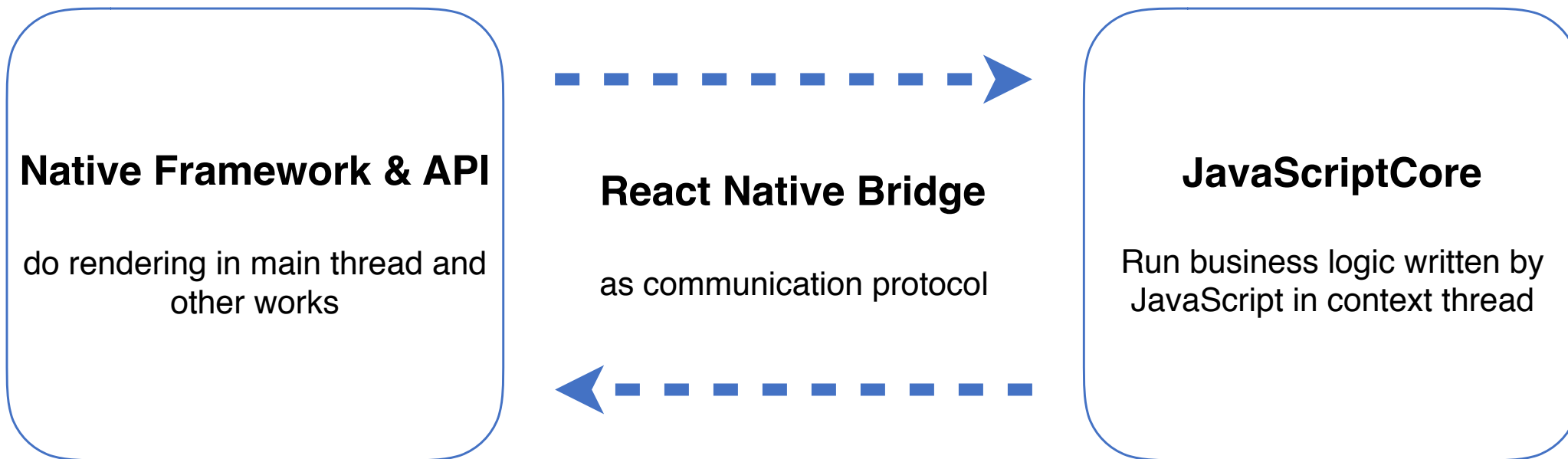
### 方式 2 - 构建一套可以跨平台的 Tool Kit

通过直接与系统底层接口交互,  
实现一套跨平台的类似 Native Kit 的工具。  
代表性方案：Flutter。

# React Native

Facebook 于 2015 年 3 月，推出的一套开源的跨平台方案。

## React Native 的实现原理



## React Native 的特点

- 能使用 JavaScript 和 React 在移动平台构建原生应用
- 支持热重载 (Hot Reloading)
- 同一套代码可以运行在 iOS, Android 以及其它平台



## React Native 的市场表现

- 成功案例

Facebook, Instagram, Pinterest,

Skype, Tesla, Uber, Walmart

... ..

- 失败案例

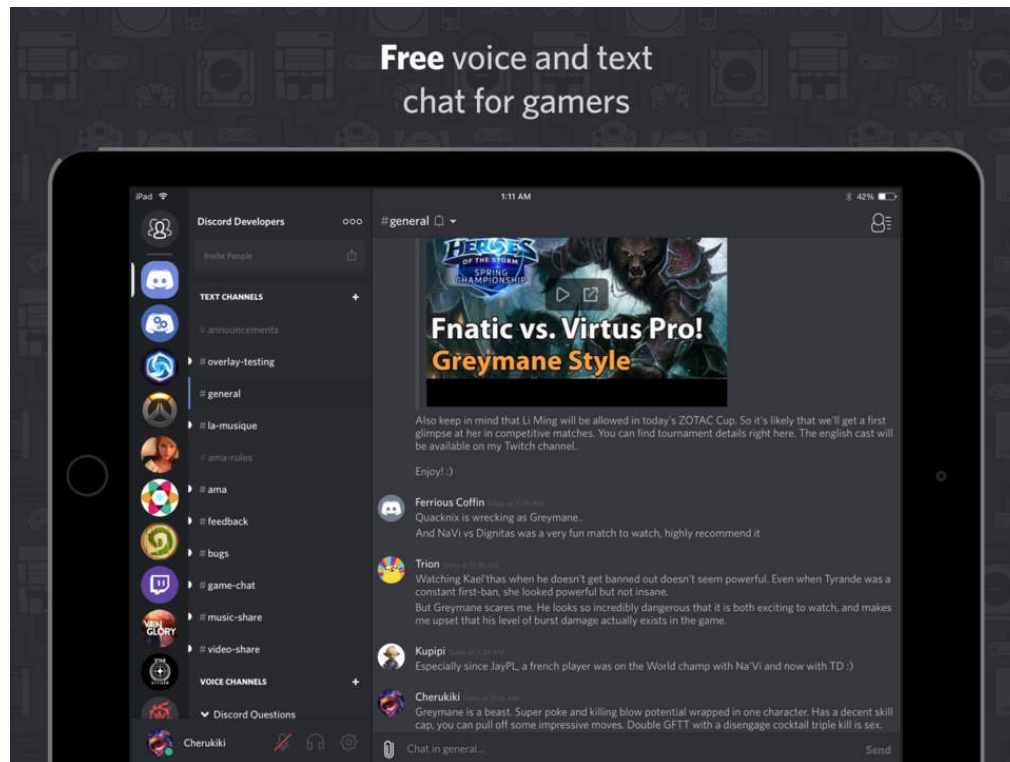
Airbnb, Udacity

**适合 React Native 的场景是什么？**

## 分析 Discord

Discord 是一个针对游戏玩家的 IM 应用

- 交互复杂度上，IM 应用比一般的信息流应用要高
- Discord 的 UI 构成很简明，主体就是 Chat View
- Discord 大量的使用了 React Native
- 所以 React Native 能够支撑 IM 这种类型的应用？



## 分析 Discord

Discord 团队对 React Native 的看法

### 优点：

- 技术栈契合
- 可以迅速的迭代
- 部分代码可以和 Web 前端共用
- 升级方便
- 支持 OTA
- 大部分组件支持 60 fps

### 缺点：

- Android 无法使用 React Native
- Chat View 使用原生的 UITableView 实现的
- 依旧需要大量的 Native 平台的知识
- App 启动慢
- 不够透明

Discord 团队会继续使用 React Native

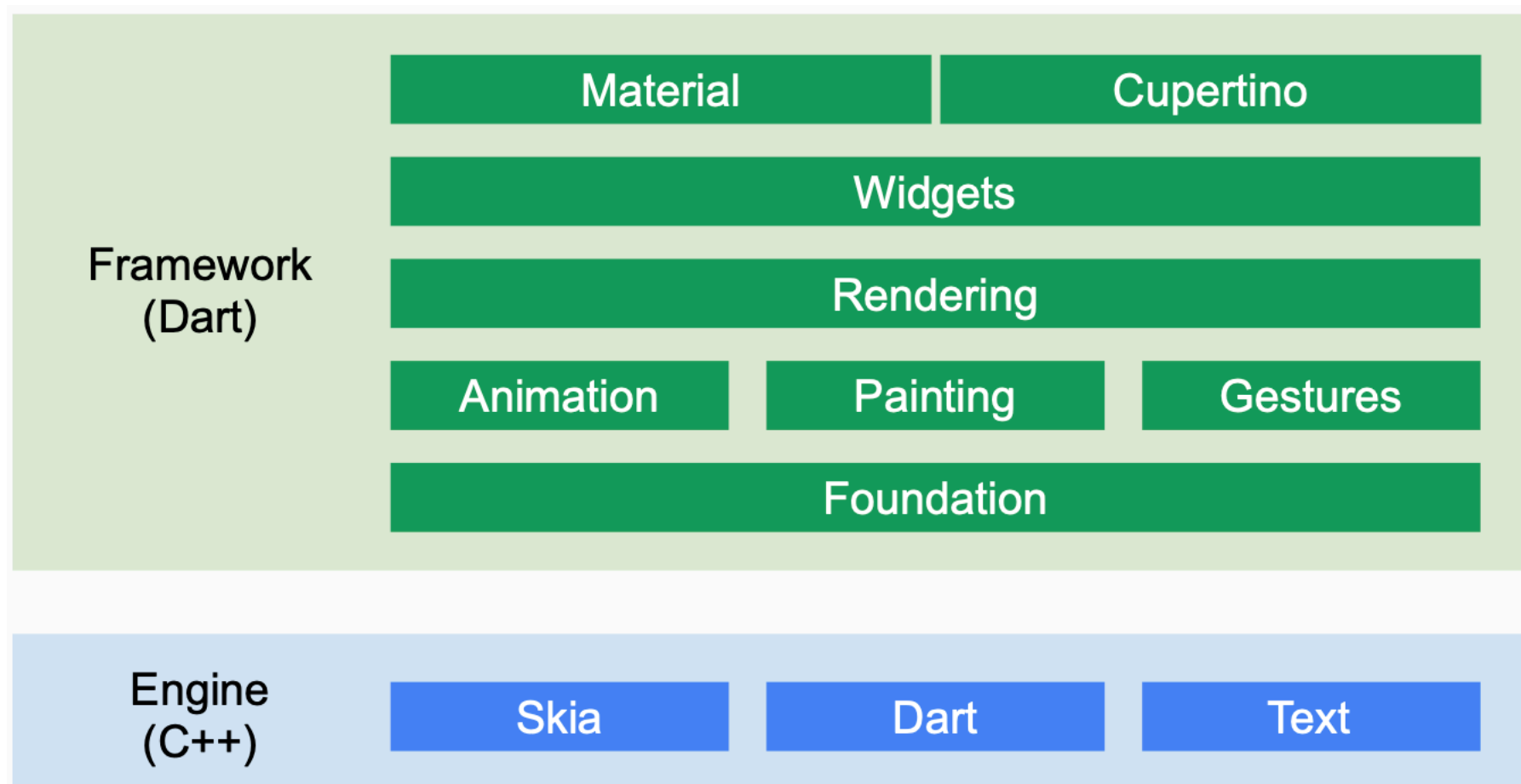
## React Native 现状

- 在功能比较单一的场景下，React Native 不会出现性能问题，可以较快的构建 UI 或原型。
- 如果交互复杂度过高，或者需要大量的计算资源，React Native 难以支撑；在跨平台的兼容性上也不尽人意。
- Facebook 计划对 React Native 的底层进行改造，来提升性能，没有时间表。

# Flutter

Google 推出一个更具「野心」的跨平台方案。

## Flutter 的系统架构



## Flutter 的特点

- 能使用 Dart 以及 React 风格的组件在移动平台构建原生应用
- 支持热重载 (Hot Reloading)
- 同一套代码可以运行在 iOS, Android 以及其它平台



## Flutter vs React Native

- 性能上，Flutter 比 React Native 好
- 平台兼容性上，目前 React Native 优于 Flutter
- Dart vs JavaScript
- 市场方面，React Native 比 Flutter 更成熟

## Flutter 的问题

**应用层** - 如何兼容各平台应用层的特性?

**底层** - 底层引擎何时从 OpenGL 迁移到 Metal?

**产品层** - 谷歌坟墓?

## 总结

- 市场依旧选择了 Native 的开发方式
- 跨平台方案提供了更多的选择
- 跨平台方案的使用场景更多的取决于环境

声网 | agora.io

RTC Dev Meetup

# THANK YOU