## **选型问题**：

开发使用的游戏引擎

## 主流的开发微信小程序的游戏引擎：

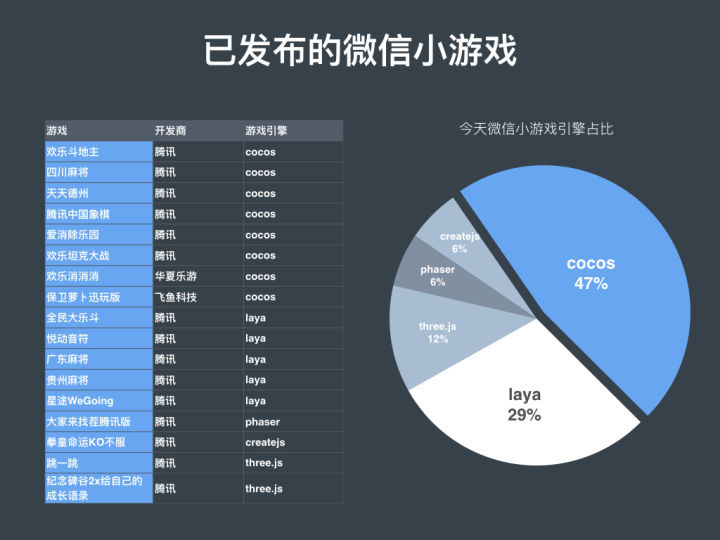
考虑以下三个主流的微信小游戏游戏引擎：

1. cocos creater
   1. 优点：市场份额大，技术成熟，有大量资料可以借鉴；使用了组件化开发的设计思想，2D编辑器比较成熟
   2. 3D编辑功能较弱
2. Laya
   1. 优点：3D编辑功能强
   2. 缺点：资料少
3. 白鹭（Egret）
   1. 优点：开始做引擎的时间很早，技术成熟
   2. 缺点：资料少；使用TypeScript语言，上手有一定难度

最终选择cocos，这是考虑到

1. 本次我们开发的圈地大作战是一个2D游戏，而cocos在2D方面比较成熟
2. 市场份额大，开发的人多，遇到问题容易查找资料，适合新手
3. 3.cocos使用c++， lua，js语言（c++文档多些），不需要学习新的语言，可以很快上手
4. 底层使用c++实现，效率高。

下图为第一批小游戏游戏引擎使用情况



## 选型问题

游戏服务器框架

## 主流游戏服务器框架

1. KBEngine

KBEngine 是一款开源 mmog 服务端引擎， 使用统一协议能够轻松与前端对接，能轻松使用 unity3d、 ogre、 cocos2d、 html5 等作为前端表现。

底层框架由 C++编写， 逻辑层使用 Python(支持热更新)。

1. Skynet

Skynet 是一个基于C跟lua的开源服务端并发框架，结构精简，C层没有多余的一堆三方库，只有紧凑的核心结构，提供最核心的消息处理框架；Lua层用来写游戏逻辑，降低了开发门槛。

1. Matchvs

<https://blog.csdn.net/sinat_22631433/article/details/98494960?depth_1-utm_source=distribute.pc_relevant.none-task-blog-BlogCommendFromMachineLearnPai2-2&utm_source=distribute.pc_relevant.none-task-blog-BlogCommendFromMachineLearnPai2-2>

最终选择Matchvs，原因

1. Matchvs是国内的服务器引擎，有大量中文文档和资料，上手快
2. Matchvs直接提供了cocos和微信小游戏的解决方案，在这方面比较成熟。

## 最终方案：

开发环境：windows 10

前端：cocos + JS

微信小游戏开发工具 + JS

后端：Matchvs

服务器： 腾讯云/Matchvs云