景和千城  
平行世界

# 概述

## 项目背景和说明

平行世界实现了在互联网上完整呈现一个个城市的超大场景，将城市中的街道、公共设施、交通、商业等有价值的构件，按1:1对应复制到互联网上，用户可以在3D城市中建立个人家园，进行购物、社交、旅游、娱乐等持续性活动，真正实现足不出户逛遍全城！

互联网和线下传统行业深度融合是行业趋势，用户已经不再满足于单纯地分类查询，浏览图片，而更希望获得目的丰富有特色的多样化、个性化体验，线上与线下的资源整合是行业必然的走向。

在互联网高速发展20年后的今天，互联网正面临重大转折。可视化互联网技术、虚拟技术、增加现实、人工智能、物联网等等新技术正以排山倒海之势铺面而来，强烈冲击着传统互联网的旧有格局。平行世界这种线上线下无缝衔接的新型模式已经是互联网时代发展的大势所趋。

## 项目范围

电子商务：

城市中的每一个店铺、商城、商业街，都能真实呈现在平台上。每位商家都有一个属于自己的3D门户网站，可以发布货品、招聘信息。网民可以根据搜索分类精准定位，培养国人全新消费习惯。与淘宝的虚拟店铺不同，平台的店铺全部真实存在，大力保护和发展实体经济，实现线上线下无缝对接。

虚拟社交：

用户注册之后，有一个自己的虚拟形象。可以选择自己的发型、身材、服饰。用户白天可以在3D城市的星巴克与客户谈生意，夜晚可以在酒吧交友或者与好友聚会。平台中配有各种虚拟道具，用户可以买豪车、豪华别墅、衣服、宠物、礼物等等，彰显身份和实力。利用实际社交圈的传播性，引爆用户群，使亿万用户进入平行世界。

## 项目要求

将电子商务和社交相结合，并以3D形式在移动端（android，iOS）和PC端呈现，用户能够以3D形象在3D城市愉悦的浏览、方便的购物、真实的社交。

# 项目复杂度和难度

平行世界将电商平台和虚拟社交平台以3D模式呈现，电商和社交单独每一个平台都有很多的功能点需要开发，再加上3D，就将整个项目的复杂度提高到一个更高的高度。

电商平台包括如下几个大功能点：

* 用户管理
* 商户管理
* 用户服务
* 商品管理
* 支付系统
* 财务系统
* 统计报表

以上几个功能点看起来不多，但是用户管理、商户管理、用户服务加起来就类似于一个CRM系统了，CRM系统的复杂性是显而易见的；商品管理至少需要一整套商品分类体系，和淘宝、京东类似，淘宝和京东在这方面大概积累了10年，才基本完善，学习和效果他们的成果和经验还需要时间。支付和财务方面数据一致性问题和安全问题也是两个比较大的难点。

电商通常面临高并发交易量的问题，交易时间比较集中，比如双十一、618。平行世界预计的用户量至少是千万，高并发的问题是必须解决的问题之一。

虚拟社交包括一下功能点：

* 用户虚拟形象，多种可选
* 真实的3D城市
* 文字、图形、声音等交流手段
* 多种关系的沟通（陌生人，好友等）

虚拟社交中人物的皮肤就有6套，还有各种发型，人物形象的制作就费时费力。真实的城市以1:1对应复制到互联网上，本身就很具挑战性，再加上需要展现到手机上，并流畅的在3D城市中浏览，就极具挑战性。原因是平行世界的3D模型资源需要实时下载，并渲染到手机上，这点和游戏不一样，游戏的场景是固定的，可以预知，游戏的资源可以打包的app中，但平行世界不能这样做，从而增加了实现的难度。多种关系，多种交流手段也增加了虚拟社区的复杂性。

# 项目情况

## 系统总体架构

平台采用微服务设计思想设计总体架构，docker部署。电商部分采用Maven做工程管理，采用php和java开发，数据库采用mysql，geoinfo部分采用golang语言开发，场景服务部分采用erlang语言开发。

下图为电商平台4月份前的一个架构图。



## 系统设计的不足

* API网关部分的负载均衡在聚合层之后，没起到作用。
* zookeeper服务没有做负载。
* java微服务没有安全措施和熔断机制。
* 对数据一致性没有做处理。
* 可通过聚合层直接访问微服务，也可通过API网关访问微服务，调用方法不统一，访问控制不统一。
* 数据持久层采用JdbcTemplate，数据映射不太方便，开发效率低。安全性不好控制，容易出现sql注入的安全问题。
* 日志收集不统一，不能追溯问题。
* 对服务划分不够合理，如：支付服务和订单没能分开。

## 系统运行时的一些问题

* 系统不太稳定。
* 汇聚层采用PHP开发，服务端采用Java开发，日志收集不统一，不利于排错。
* REST API调用没有统一的返回值，没有错误表示，汇聚层出错不能明确问题。
* 没有分库，一些服务之间耦合度增大。

## 系统维护部署复杂

* 场景服务采用erlang开发，由于技术问题不能实现采用docker部署，增加了部署的复杂度。
* geoinfo的数据库采用了postgresql和mongodb，与电商的mysql加起来，平行世界一共有3种数据库，数据库的使用和维护的难度很大。
* 平行世界涉及的技术栈多而且广。移动端和PC端有android、iOS和unity，后台服务的开发语言有php、java、golang、erlang，前端开发有javascript，其它项目还有c++，数据库有mysql、postgresql和mongodb，其它框架有redis，mq，zookeeper，vue.js和对象存储等。技术栈的复杂性使开发、管理、部署和维护的难度极大。
* 场景服务不能用docker部署，多种部署方式并存，造成运维工作复杂，工作量增大。

# 项目优化方向

分析当前系统的不足和运行时的问题，首先重新评估了Spring Cloud和Dubbo/DubboX这两个微服务框架。Spring Cloud是比较适合的一个框架，可以主要用于电商平台。

* 引入API网关作为服务器，是系统的唯一入口。API网关封装了系统内部架构，为每个客户端提供一个定制的API。它还具有其它职责，如身份验证、监控、缓存。API网关负责服务请求、汇聚（组合微服务api接口）及协议转换。
* 汇聚层取消，作为API网关的一部分。统一入口，减少层级。
* API网关根据不同的业务场景分别做定制。分如下几种：用户（APP）API网关、商家API网关、后台（eboss）API网关、联营API网关、众包API网关、用户（PC）网关、资源下载网关、城市渲染网关、IM网关、第三方接口API网关（第三方接口API网关后期在实现）。
* API网关实现数据格式转换和规整。
* API网关为日志跟踪提供traceId。
* 采用开源组件ELK来实现日志收集
* 资源文件采用青云对象存储进行保存和下载，需要加深与青云合作沟通，要求青云定制一些API，做到资源打包下载，减少握手和建链次数和时间，提高用户体验。
* 采用基于Spring Cloud和Netflix打造熔断器Hystrix。引入Hystrix可以避免服务出现瘫痪的问题。