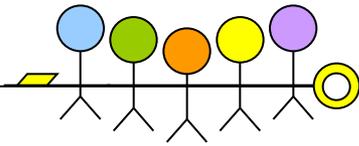


火龙果讲堂：

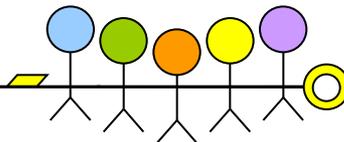
- 一线专家
- 案例回顾
- 经验分享



电子商务平台架构设计

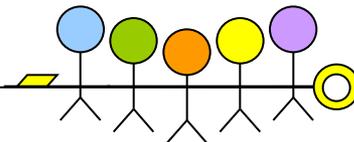
孟祥龙

本讲座语音实录：uml.org.cn 的讲座吧



- ❖ 电子商务平台面临的挑战
- ❖ 如何应对挑战
- ❖ 电子商务平台分类
- ❖ 电子商务平台业务架构设计
- ❖ 电子商务平台技术架构设计
- ❖ 交流与讨论

电子商务平台面临的挑战

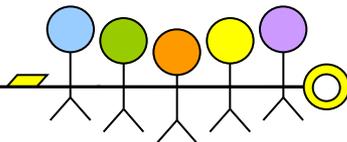


火龙果讲堂

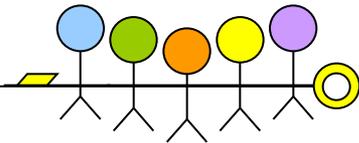
uml.org.cn

- 用户体验决定成败
- 业务类型快速增长
- 业务量急剧膨胀
- 实时可靠提供服务

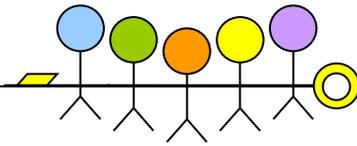
如何应对挑战？



- 掌握业务特点和用户特征，正确将平台归类
- 设计可灵活扩展的业务架构
- 构建实时高可用的技术架构
- 不断优化和完善架构，应对新挑战



- B2B——Business To Business
- B2C——Business To Customer
- C2C——Customer To Customer
- O2O——Online To Offline
- 其他——电商导购等



- **B2B平台(Business To Business):**

企业与企业之间通过网络进行产品、服务和信息的交换

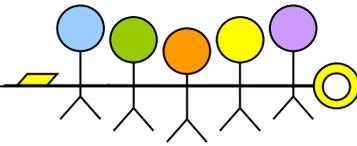
- **挑战:**

业务访问量高、成交金额大、业务灵活度高

- **代表站点:**

综合类——阿里巴巴、环球资源、慧聪网

垂直类——网盛生意宝、我的钢铁网、全球纺织网



- **B2C平台(Business To Customer):**

商家通过网站向个人消费者提供商品和服务的电商平台

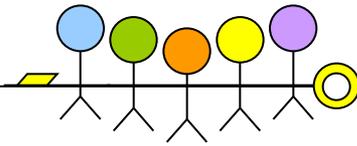
- **挑战:**

买家用户多，用户体验要求高，经常性的促销活动要求系统高可靠、高实时

- **代表站点:**

有形商品和服务：天猫、京东、亚马逊、当当等

无形商品和服务：搜索引擎广告平台、猪八戒威客网等



- **C2C平台(Customer To Customer):**

为消费者之间提供产品、服务交易的电子商务平台

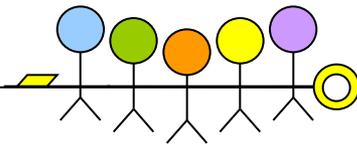
- **挑战:**

用户量巨大，商品种类繁多，交易次数多而金额小，系统访问强度较均匀，系统需要持续稳定地提供可靠服务

- **代表站点:**

淘宝、易趣、拍拍等

电子商务平台分类——O2O



火龙果讲堂

uml.org.cn

•O2O平台(Online To Offline):

指线上营销线上购买带动线下经营和线下消费的电商平台

•O2O特点:

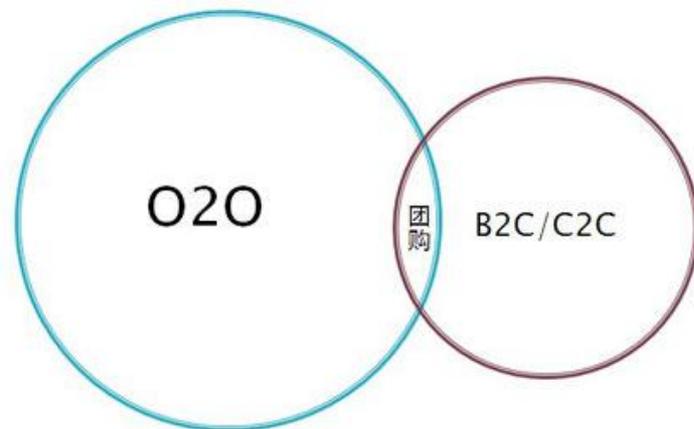
O2O通过打折、提供信息、服务预订等方式,把线下商店的消息推送给互联网用户,从而将他们转换为自己的线下客户,这就特别适合必须到店消费的商品和服务

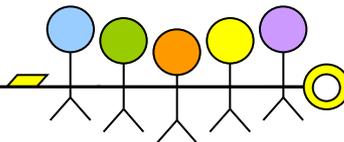
•挑战:

平台模式尚不成熟,业务扩展能力要求高,线上与线下信息要求实时同步

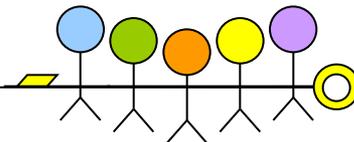
•代表站点:

美团、拉手等团购网站,
但团购只是O2O的一种类型





- ❖ 电子商务平台面临的挑战
- ❖ 如何应对挑战
- ❖ 电子商务平台分类
- ❖ 电子商务平台业务架构设计**
- ❖ 电子商务平台技术架构设计
- ❖ 交流与讨论



➤ 业务流程分析与设计

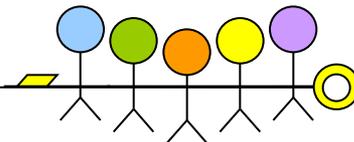
如：订购流程、支付流程、订单处理流程等

➤ 业务子系统划分

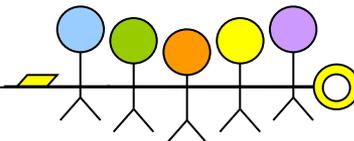
如：平台前端系统、后端管理系统、会员系统等

➤ 业务系统接口需求分析与设计

如：会员系统如何与财务系统交互等

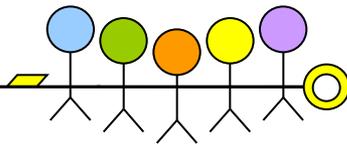


- 业务流程清晰，能解决平台的核心需求
- 系统划分能支撑业务流程的灵活扩展需要
- 业务系统间接口明确，系统内部调整不影响接口调用



- 分析平台的业务流程，确定流程主要环节
- 根据流程各个环节确定相关业务功能
- 业务功能汇总分类，映射成各个业务系统
- 梳理业务子系统之间的依赖关系，设计系统接口需求

示例：电商平台业务架构设计

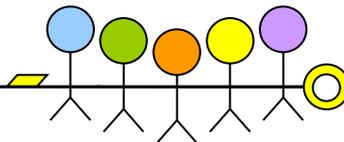


火龙果讲堂

uml.org.cn

- 普通用户的订购流程分析
- 商家的产品销售流程分析
- 平台方的管理流程分析
- 服务各个过程的业务系统

业务架构设计：用户购物流程



火龙果讲堂

uml.org.cn

会员系统



客户注册会员

商品推荐
站内搜索
商品展示



商品搜索选购

订购系统
CallCenter



下订单（放进购物车）



收银台



订单查询



购物完成



在线支付（或汇款）



选择送货方式



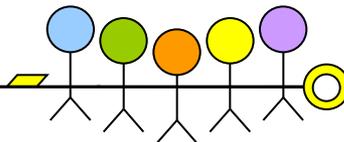
评价

评价系统
CallCenter

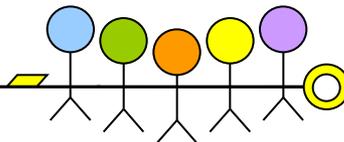
支付系统

配送系统

业务架构设计：商家处理流程



业务架构设计：平台方管理流程



管理后台

购物流程管理

- 审核会员与商品
- 订单系统维护
- 客户关系维护
- 用户投诉处理
- 配送体系维护
- 支付体系维护
- 用户评价维护

广告业务管理

- 商品竞价管理
- 平台广告管理
- 联盟站点合作与推广

数据管理

- 业务数据挖掘
- 个性化商品推荐
- 相关统计报告
- 财务结算

审核系统

订单管理系统

CRM系统

CallCenter

物流管理系统

支付管理系统

评价管理系统

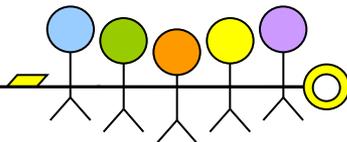
广告管理系统

联盟合作管理系统

个性化推荐系统

数据统计系统

财务结算系统



电子商务平台

购物流程相关

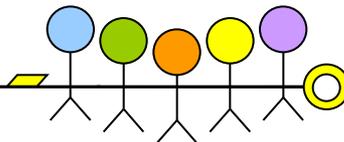
- 会员系统
- 商品展示系统
- 商品发布系统
- 商品订购系统
- 物流配送系统
- 订单管理系统
- 支付系统
- 评价系统
- CRM系统
- 搜索系统
- CallCenter

非购物流程

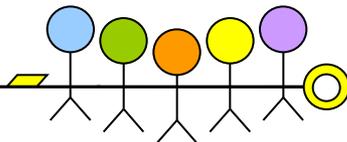
- 商品推荐系统
- 广告商业系统
- 联盟合作系统
- 数据统计系统
- 财务结算系统

其他

- 运维监控等系统



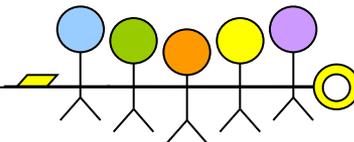
- ❖ 电子商务平台面临的挑战
- ❖ 如何应对挑战
- ❖ 电子商务平台分类
- ❖ 电子商务平台业务架构设计
- ❖ 电子商务平台技术架构设计**
- ❖ 交流与讨论



- 技术架构设计是什么
- 平台架构设计目标与原则
- B2B锣鼓网平台架构分析
- B2C演出票务平台架构分析
- 商业广告系统架构分析

提示：三个案例是结合实际项目经历按照时间先后展开讨论的，同时符合系统规模从小到大的发展过程

技术架构设计是什么？

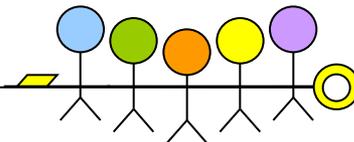


■ 硬件架构

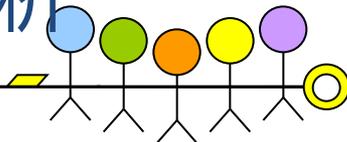
- 网络拓扑结构设计
- 系统服务器部署方案

■ 软件架构

- 系统分层设计
- 集群与负载均衡技术
- 开发框架技术运用
- 缓存系统架构
- 存储体系架构
- 。 。 。



- 优先满足业务需求
- 系统可扩展、易维护
- 不同规模的平台有不同的目标
- 技术和业务共同驱动架构完善

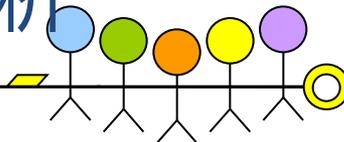


■ 主要功能

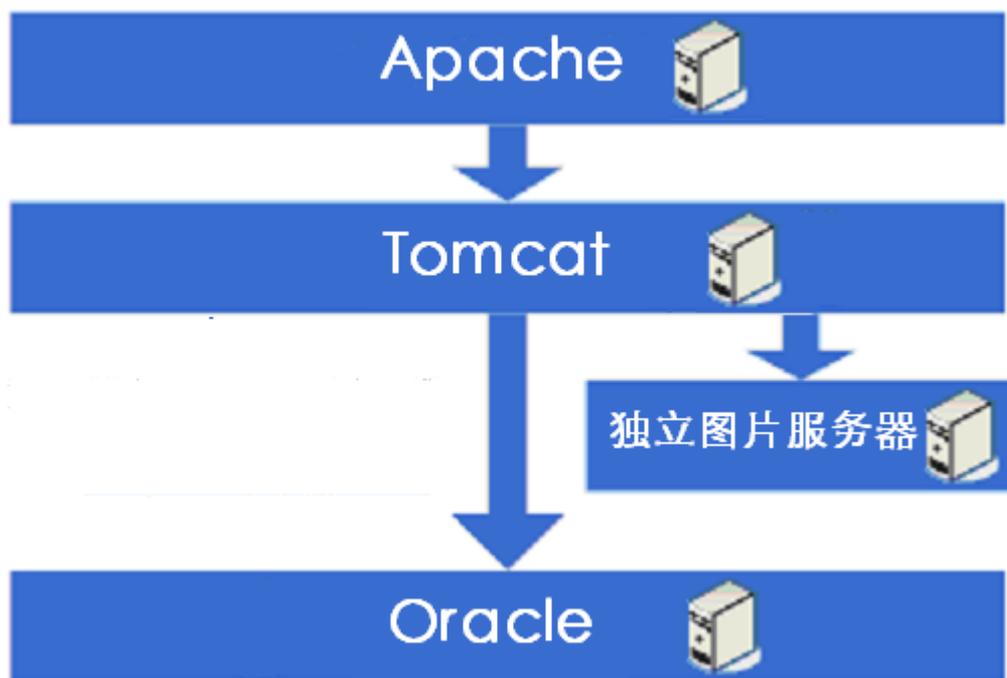
- 时间：2008年初
- 实现为垂直行业的中小企业向国内外推广商品的需求
- 平台建设期正值公司初创，所以主要目的为了满足基本业务功能，以便进一步协助业务推广

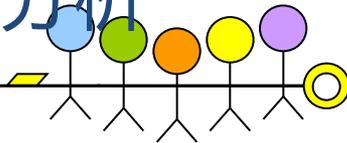
■ 数据规模

- 日独立IP5000以下，单表数据规模在百万以下，典型的小网站



■ 系统架构图



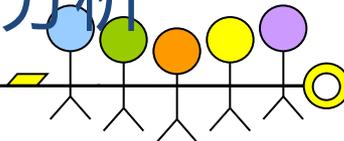


■ 主要功能

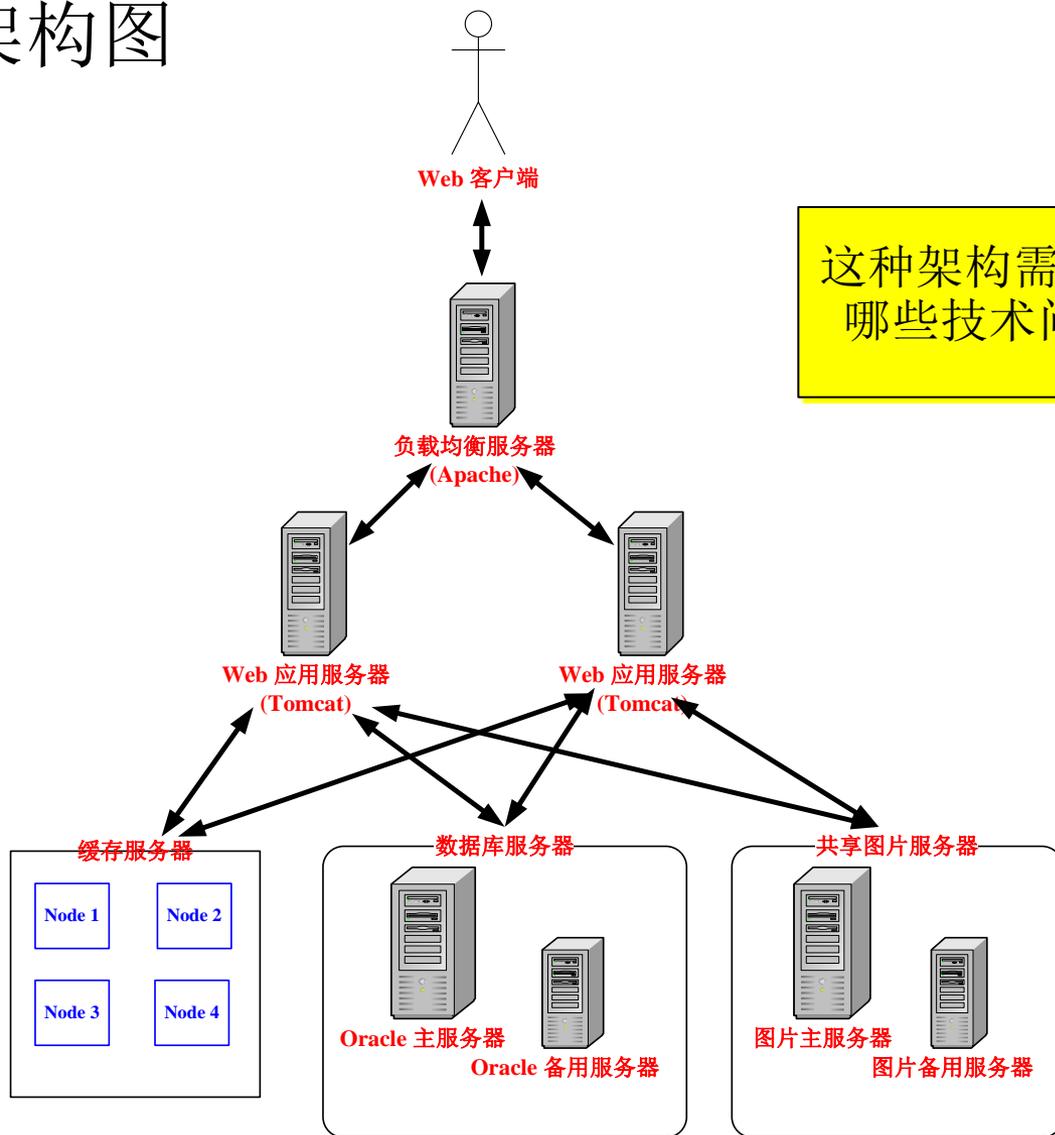
- 时间：2009年6月
- 通过网站向普通观众销售各类演出门票

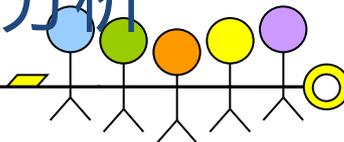
■ 业务规模

- 日独立IP 2w以下，单表数据量百万级，属于中小型系统平台



■ 系统架构图



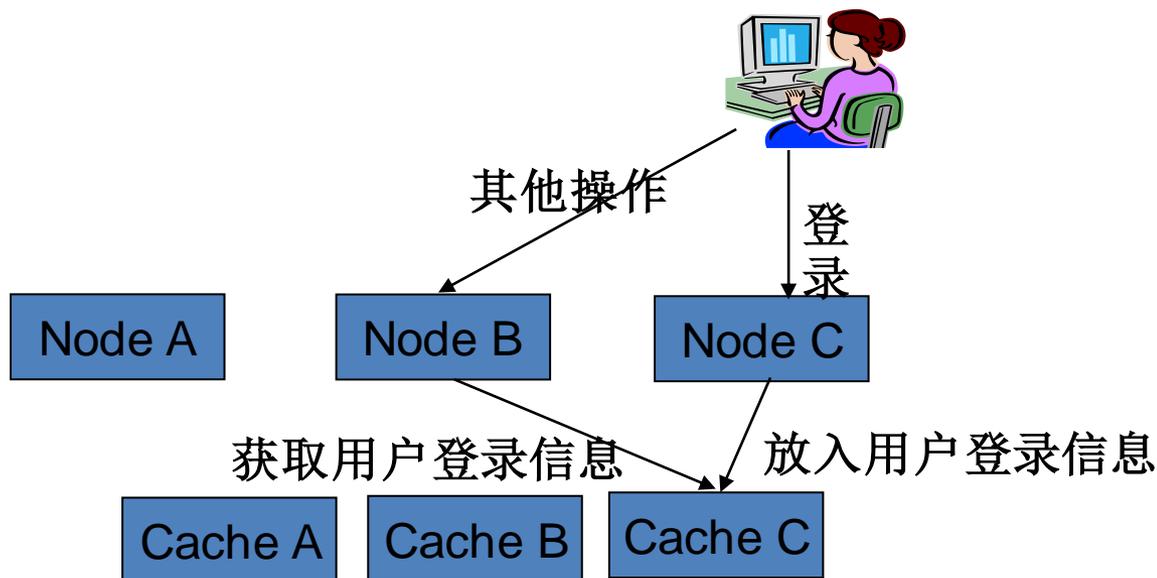


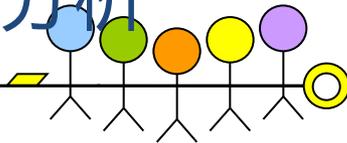
■ 系统架构设计说明

– 系统采用反向代理负载均衡

– Apache+tomcat集群和负载均衡

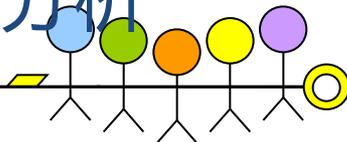
– 利用memcached的一致性hash和cookie技术实现基于分布式缓存的session一致





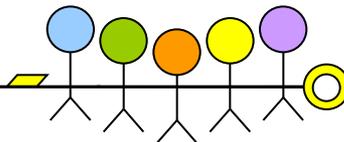
■ 系统架构设计说明

- 由memcached节点组成缓存服务器
 - 实现常用数据的缓存化，减少数据库压力
 - 将memcached作为hibernate二级缓存
 - 方便水平扩展，内存大小不受限
- 图片服务器（FTP+http）
 - 通过ftp机制上传到图片服务器
 - 独立lighttpd服务器提供图片访问
 - 解决了图片服务稳定、不影响web服务器性能、适用集群部署web服务器等问题
- 使用FreeMarker实现web页面静态化
- 使用Oracle RAC集群技术管理DB



■ 系统架构的问题

- 业务子系统规划与分离
- 缓存服务器的容错处理
- 负载均衡服务器的单点故障问题
-

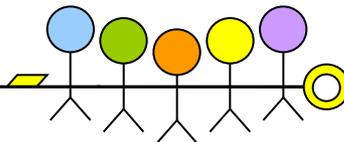


■ 主要功能

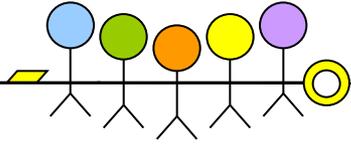
- 为广告商在不同产品、不同网站资源中投放广告，达到用户产品推广效果
- 由于广告系统本身很复杂、且系统组成很多，此处只介绍提供给广告商、客服、管理方的后端相关业务系统设计

■ 业务规模

- 核心广告业务系统日独立IP在10w以上，关键数据表数据规模达到千万~亿级别



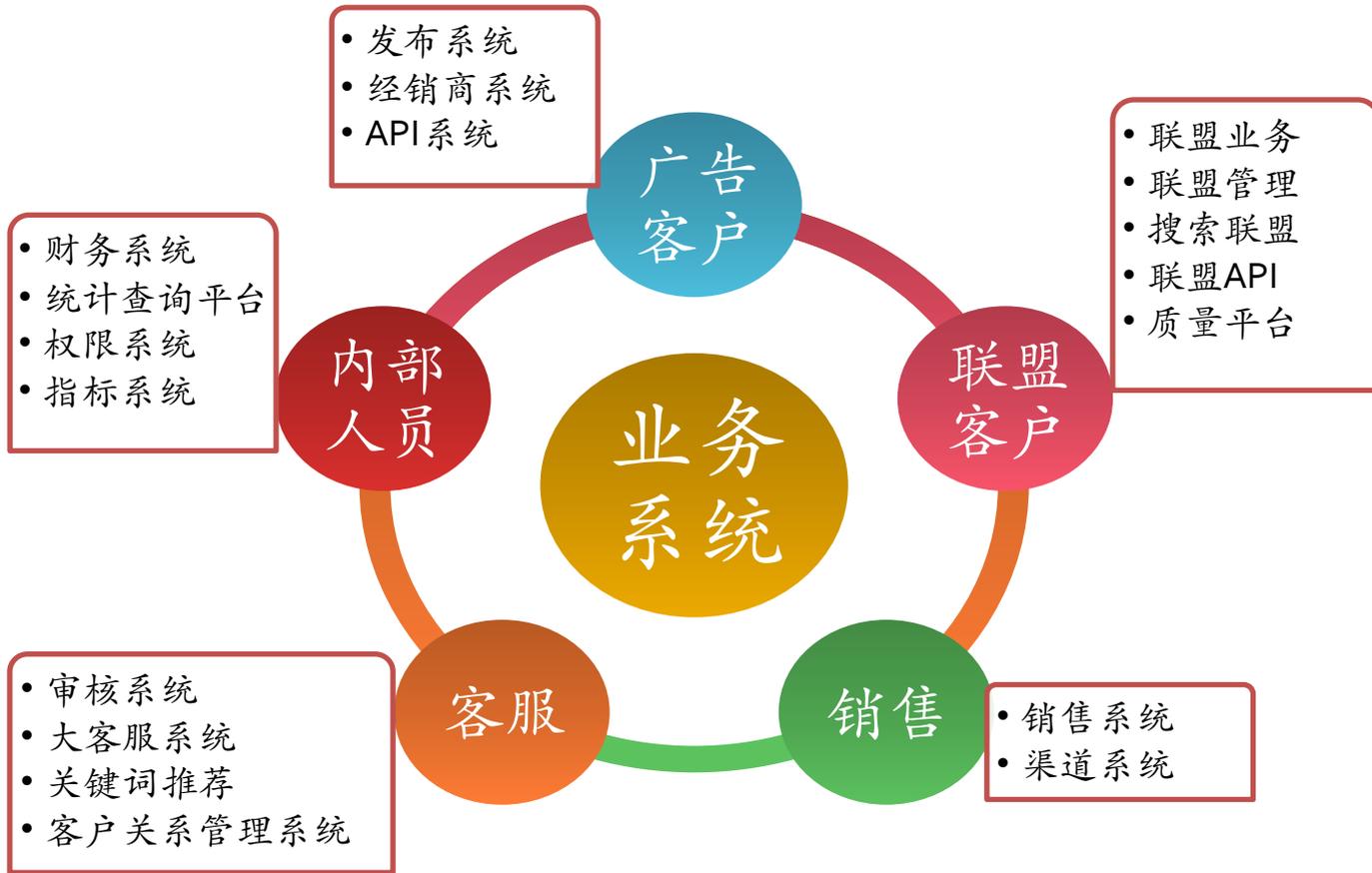
- 设计目标：支持更多流量和客户
 - 高并发性
 - 高实时性
 - 高可靠性
 - 高扩展性
 - 大数据量

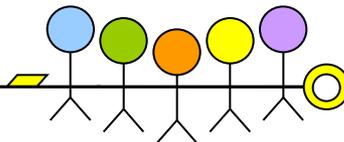


■根据业务做切分

-设计多种独立业务系统

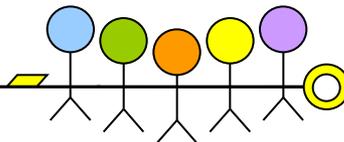
-根据业务实现数据分库存储





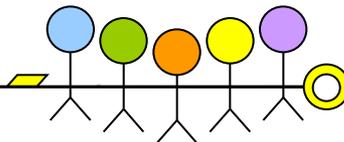
■ 系统架构图





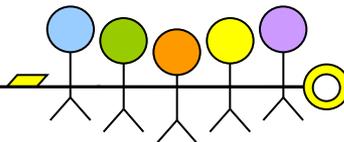
■ 系统架构技术说明

- 前端优化、CDN
- Web服务器集群部署、负载均衡
- memcached、redis等缓存技术
- ODFS分布式文件系统
- MySQL双主多从 (mmm)
- DDB分布式数据库系统
- Cassandra、MongoDB等NoSQL数据库技术
- Oimage独立图片代理/存储服务
- 系统监控技术

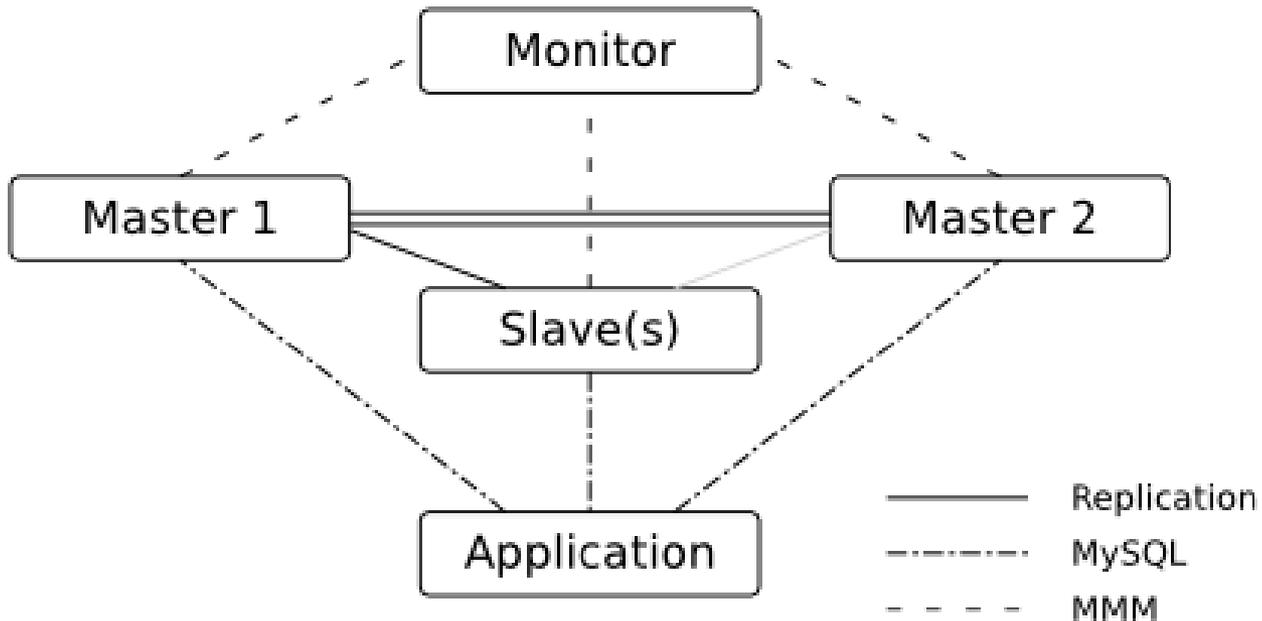


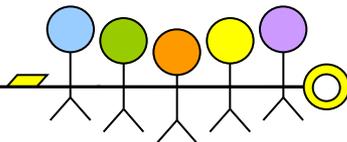
■ ODFS 分布式文件系统

- 分布式文件存储
- 针对大文件优化
- 文件只读，只能被顺序写入。文件被创建之后，就不能被修改和追加数据
- 针对顺序读操作优化，适应 CoWork 的需求
- 多副本。默认情况下，一个数据块有 3 个副本，可保证磁盘损坏时不丢失数据
- 类似 GFS、HDFS、NDFS 等



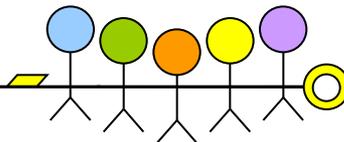
■MySQL双主多从（mmm）





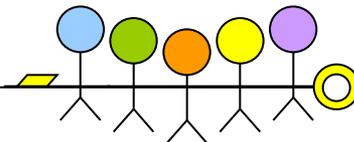
■DDB分布式数据库系统

- 网易公司基于海量数据的各种前台应用提供统一的数据库管理平台
- 该平台通过中间件的形式为前台提供一台虚拟的数据库服务器（**Master**部分）
- 中间件在后台实际管理着多个数据库节点（**DBN**部分）
- DBN**说明：为了解决单张表的数据和访问量过多造成的系统性能瓶颈，该系统允许将单张表分布到不同的数据库节点上进行性能均衡。同时允许后台数据库节点的动态加入，支持在线的可扩展，而不会影响系统的正常服务



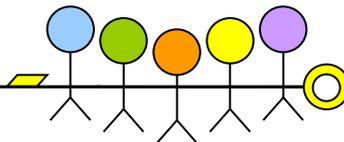
■ Oimage 图片服务器

- Oimage 是一个图片代理/存储服务，它是基于 cache+voldemort+OMap 存储的，目前开放的服务主要有获取网络图片、上传图片
- voldemort 是一个开源的分布式的 key-value 存储系统，实现了 Dynamo 的部分功能
- OMap (Outfox Map) 是一个分布式大规模数据表存储系统。类似的系统有 Google BigTable。OMap 可以看作一个大表。它支持基本的 put(), get() 和 cursor() 操作，另外还具备查询等高级特性。OMap 运行在 ODFS 之上，ODFS 负责下层数据的存储和备份。



交流时间





讲座

5月25日 软件工程集成研发管理平台

6月15日 价值驱动的多版本产品需求

QQ群

157086258 (架构设计)

更多...

uml.org.cn 文章, 文档, 课程