

# 一、简述

## 1.1 技术栈

- 这套技术体系的技术栈？

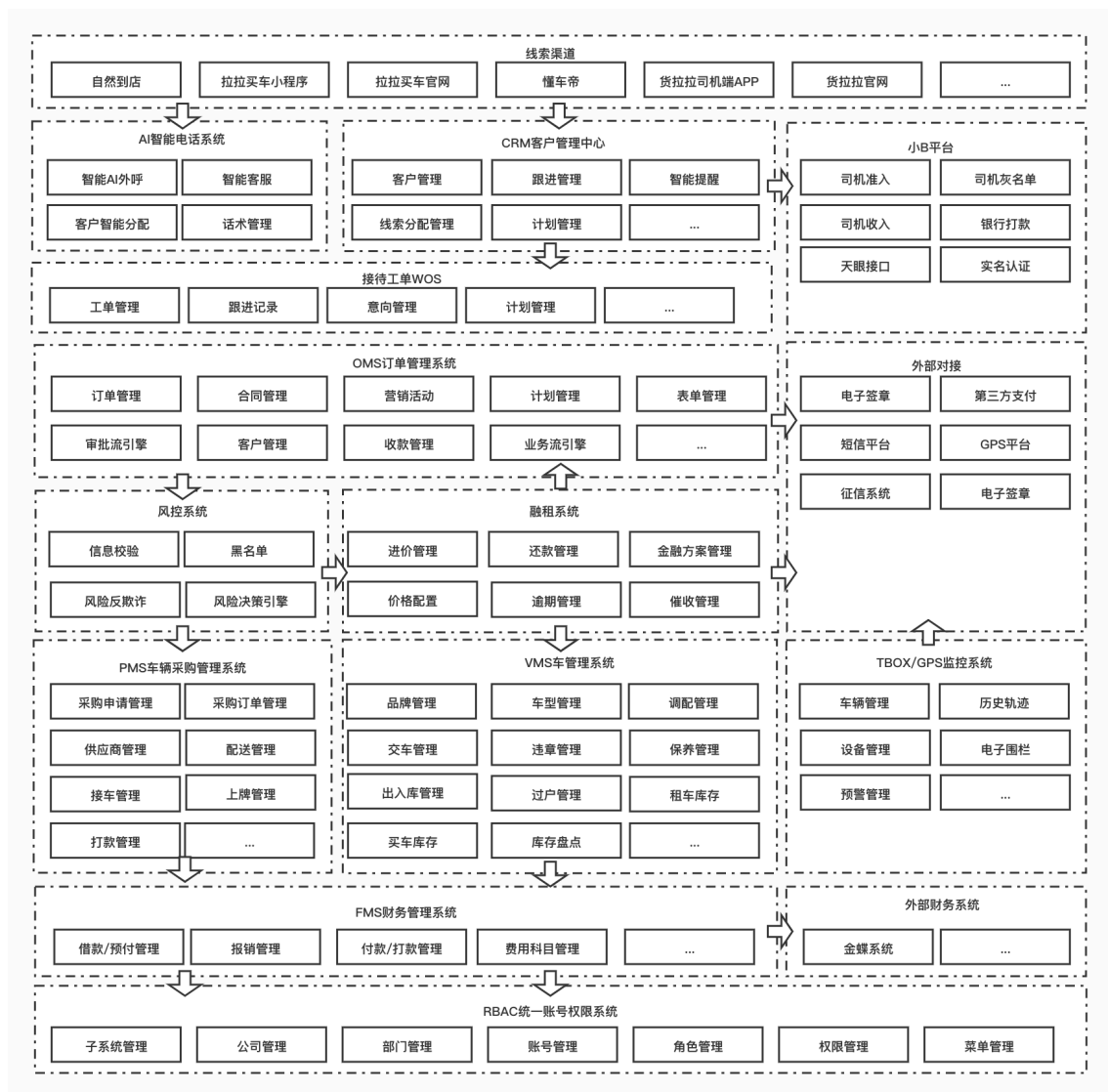
1. Nginx
2. Spring Boot
3. Spring Cloud
4. Eurka
5. Feign
6. Xxl-job
7. Kafka
8. Redis/Mysql/MongoDB/ES

- 该技术栈的由来？

该技术栈是由传统项目升级到微服务架构演变而来, 适用于汽销部门的业务场景特点 —— 并发、性能相对要求不高。

## 1.2 业务块

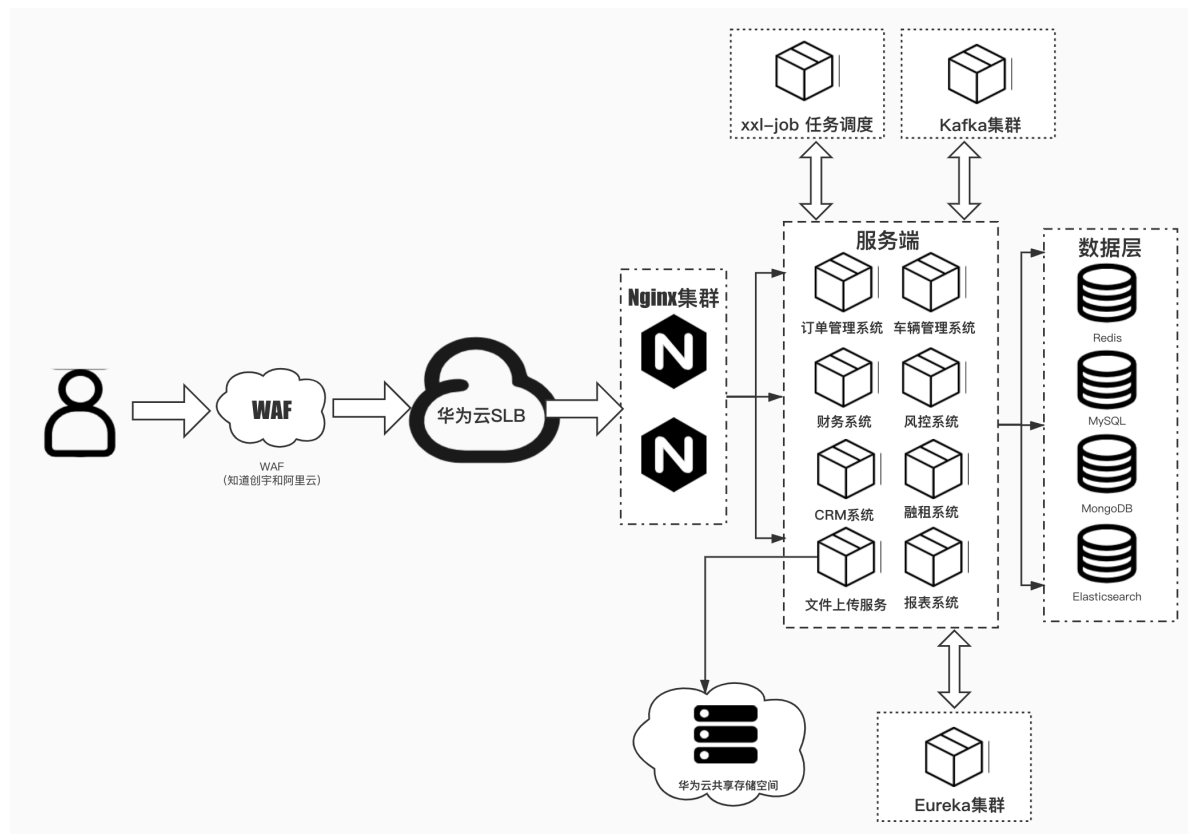
- 累积的业务模块？



## 二、架构

### 2.1 设计图

采用该架构的系统模块: OMS/VMS/PMS/FMS/DMS/CEM



### 2.2 设计概要

- 架构设计的由来

鉴于业务性质，系统在 '三高' 方面仅对可用性有一定的要求所以

- 采用了Nginx作为应用层的网关

它有轻量级且高性能、支持插槽式扩展网关功能等特点, 方便日后的扩展

- 引入了SpringCloud作为分布式系统的基础设计
- 采用的是当时最出色的Eureka作为高可用注册中心
- 使用Feign/Ribbon声明式服务调用组件简化开发
- 文件系统采用NFS挂载让开发无感知读写文件
- 异步业务的处理引入了Kafka作为通用的消息中间件
- 定时任务的处理采用的是xxl-job作为统一调度中心

它解决了分布式场景下统一调度的问题

- 以MySQL做业务数据库
- 用Redis做业务热点数据的缓存
- MongoDB/Es是基础/历史数据存储的备选方案

## 三、部署

### 3.1 部署方案

#### 3.1.1 Jenkins自动化部署

- 结合Git将 *Jenkins* 打造成自动化部署/打包的工具, 使开发团队达到: 快上传, 可视化效果。

#### 3.1.2 Jar手动上传

- 没有代码合并、多分支覆盖, 有"本地既远程的效果"。

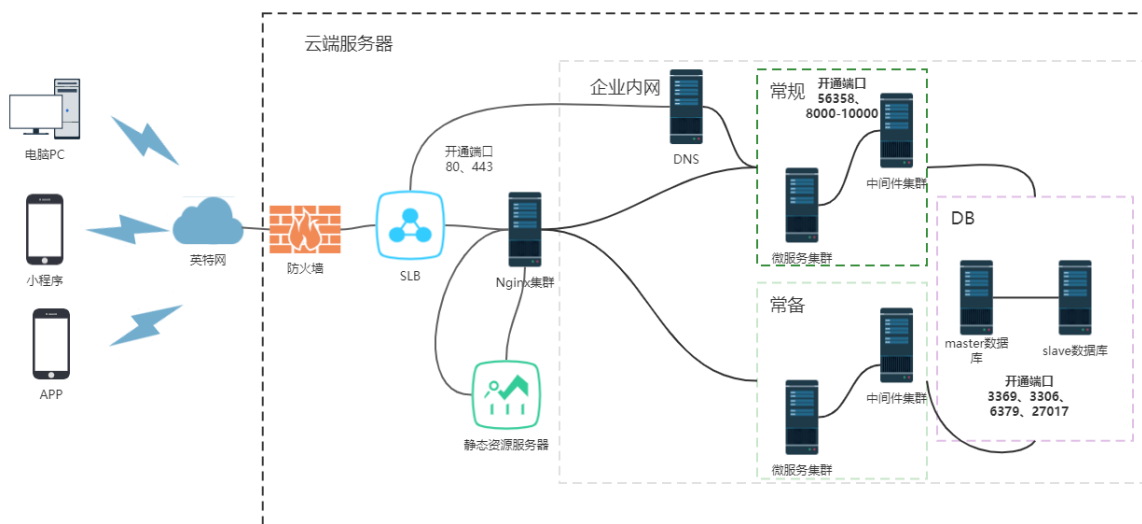
### 3.2 环境支持

- 支持哪些环境?
  - 本地环境: loc
  - 开发环境: dev
  - 测试环境: stg
  - 预发环境: pre
  - 生产环境: prd
- 多环境支持的意义?
  - 可以将不同环境的内容进行隔离
  - 可以方便不同环境的部署和切换
  - 可以适配不同的用户, 比如:

本地是开发者开测使用, 开发环境是开发者联调/对接使用, 测试环境是测试人员使用, 预发环境是验收专用, 生产是真实客户专用。

### 3.3 部署架构

#### 3.3.1 网络拓补图



### 3.3.2 部署概要

- 运维采用阿里云/华为云负载均衡解决方案
- 将前端静态资源采用Nginx作为服务器独立部署
- 内网启用公司内部DNS做内网域名解析
- 启用业务服务器常备方案部署

独立部署两套业务服务, 采用不同域名作为入口, 将代码隔离

- oms.huolala.cn: 常规业务服务域名
- oms2.huolala.cn: 常备业务服务域名: 只在常规服务崩溃时紧急使用
- Redis采用哨兵模式部署
- Mongodb/Es采用集群模式部署

## 四、结语

---

- 以上是OMS/DMS等系统采用的技术设计.