前言

　　sanhu项目创建于2015年5月8日，正在慢慢成长中，目的不仅仅是一个开发架构，而是努力打造一套从 前端模板 - 基础框架 - 分布式架构 - 开源项目 - 持续集成 - 自动化部署 - 系统监测 - 无缝升级 的全方位J2EE企业级开发解决方案。

## 项目介绍

　　基于Spring+SpringMVC+Mybatis分布式敏捷开发系统架构，提供整套公共微服务服务模块：内容管理、支付中心、用户管理（包括第三方）、微信平台、存储系统、配置中心、日志分析、任务和通知等，支持服务治理、监控和追踪，努力为中小型企业打造全方位J2EE企业级开发解决方案。

### 组织结构

sanhu

├── sanhu-common -- SSM框架公共模块

├── sanhu-admin -- 后台管理模板

├── sanhu-ui -- 前台thymeleaf模板

├── sanhu-config -- 配置中心

├── sanhu-power -- 用户权限管理系统

| ├── sanhu-power-common -- power系统公共模块

| ├── sanhu-power-dao -- 代码生成模块，无需开发

| ├── sanhu-power-client -- 集成power依赖包，提供单点认证、授权、统一会话管理

| ├── sanhu-power-rpc-api -- rpc接口包

| ├── sanhu-power-rpc-service -- rpc服务提供者

| └── sanhu-power-server -- 用户权限系统及SSO服务端

├── sanhu-pay -- 支付系统

| ├── sanhu-pay-common -- pay系统公共模块

| ├── sanhu-pay-dao -- 代码生成模块，无需开发

| ├── sanhu-pay-rpc-api -- rpc接口包

| ├── sanhu-pay-rpc-service -- rpc服务提供者

| ├── sanhu-pay-sdk -- 开发工具包

| ├── sanhu-pay-admin -- 后台管理

| └── sanhu-pay-web -- 演示示例

├── sanhu-ucenter -- 用户系统(包括第三方登录)

| ├── sanhu-ucenter-common -- ucenter系统公共模块

| ├── sanhu-ucenter-dao -- 代码生成模块，无需开发

| ├── sanhu-ucenter-rpc-api -- rpc接口包

| ├── sanhu-ucenter-rpc-service -- rpc服务提供者

| └── sanhu-ucenter-web -- 网站前台

├── sanhu-wechat -- 微信系统

| ├── sanhu-wechat-mp -- 微信公众号管理系统

| | ├── sanhu-wechat-mp-dao -- 代码生成模块，无需开发

| | ├── sanhu-wechat-mp-service -- 业务逻辑

| | └── sanhu-wechat-mp-admin -- 后台管理

| └── sanhu-ucenter-app -- 微信小程序后台

├── sanhu-api -- API接口总线系统

| ├── sanhu-api-common -- api系统公共模块

| ├── sanhu-api-rpc-api -- rpc接口包

| ├── sanhu-api-rpc-service -- rpc服务提供者

| └── sanhu-api-server -- api系统服务端

├── sanhu-oss -- 对象存储系统

| ├── sanhu-oss-sdk -- 开发工具包

| ├── sanhu-oss-web -- 前台接口

| └── sanhu-oss-admin -- 后台管理

├── sanhu-shop -- 电子商务系统

└── sanhu-demo -- 示例模块(包含一些示例代码等)

├── sanhu-demo-rpc-api -- rpc接口包

├── sanhu-demo-rpc-service -- rpc服务提供者

└── sanhu-demo-web -- 演示示例

### 技术选型

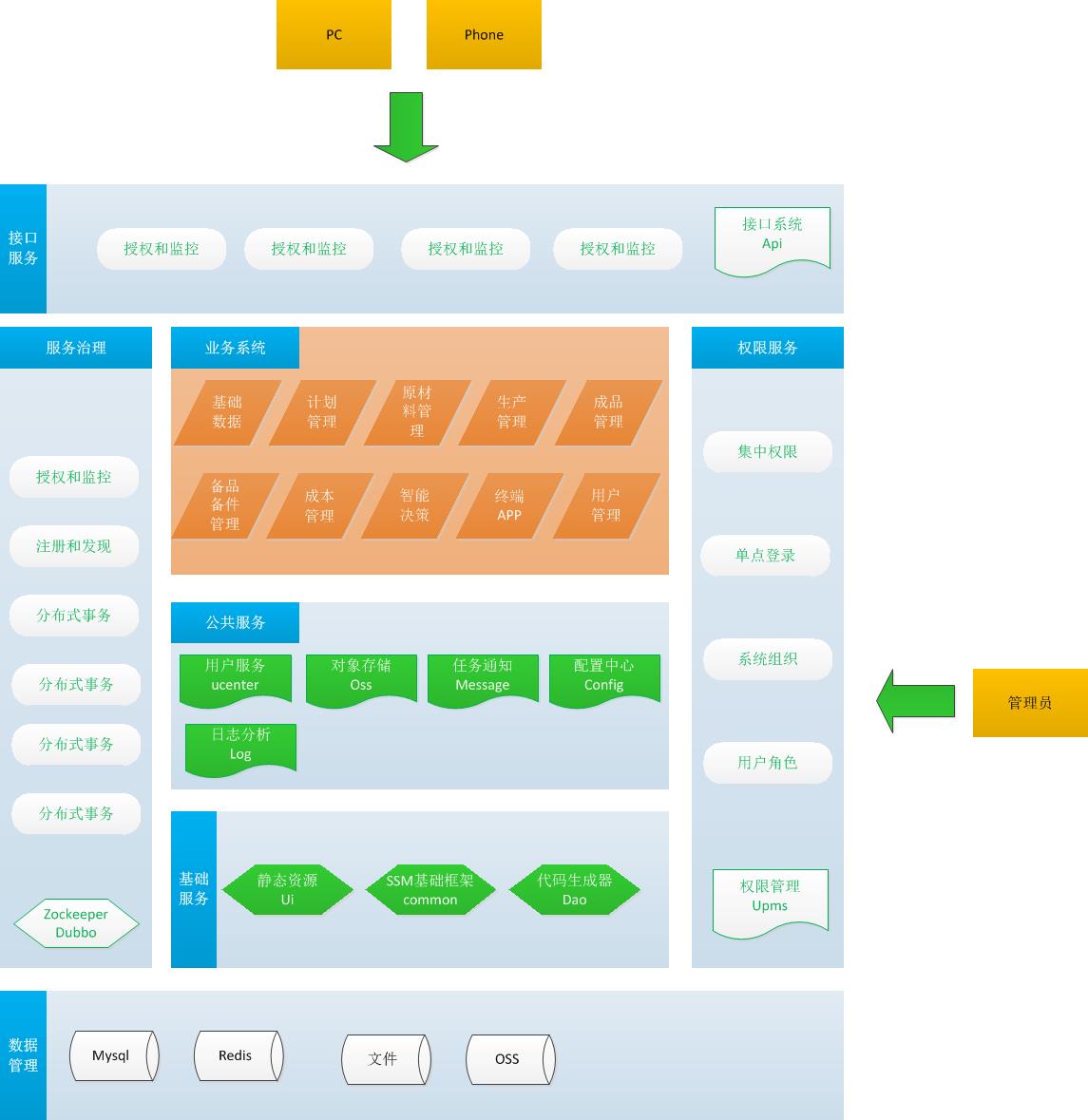
#### 后端技术:

|  |  |  |
| --- | --- | --- |
| **技术** | **名称** | **官网** |
| Spring Framework | 容器 | <http://projects.spring.io/spring-framework/> |
| SpringMVC | MVC框架 | <http://docs.spring.io/spring/docs/current/spring-framework-reference/htmlsingle/#mvc> |
| Apache Shiro | 安全框架 | <http://shiro.apache.org/> |
| Spring session | 分布式Session管理 | <http://projects.spring.io/spring-session/> |
| MyBatis | ORM框架 | <http://www.mybatis.org/mybatis-3/zh/index.html> |
| MyBatis Generator | 代码生成 | <http://www.mybatis.org/generator/index.html> |
| PageHelper | MyBatis物理分页插件 | <http://git.oschina.net/free/Mybatis_PageHelper> |
| Druid | 数据库连接池 | <https://github.com/alibaba/druid> |
| FluentValidator | 校验框架 | <https://github.com/neoremind/fluent-validator> |
| Thymeleaf | 模板引擎 | <http://www.thymeleaf.org/> |
| Velocity | 模板引擎 | <http://velocity.apache.org/> |
| ZooKeeper | 分布式协调服务 | <http://zookeeper.apache.org/> |
| Dubbo | 分布式服务框架 | <http://dubbo.io/> |
| TBSchedule & elastic-job | 分布式调度框架 | <https://github.com/dangdangdotcom/elastic-job> |
| Redis | 分布式缓存数据库 | <https://redis.io/> |
| Solr & Elasticsearch | 分布式全文搜索引擎 | <http://lucene.apache.org/solr/> <https://www.elastic.co/> |
| Quartz | 作业调度框架 | <http://www.quartz-scheduler.org/> |
| Ehcache | 进程内缓存框架 | <http://www.ehcache.org/> |
| ActiveMQ | 消息队列 | <http://activemq.apache.org/> |
| JStorm | 实时流式计算框架 | <http://jstorm.io/> |
| FastDFS | 分布式文件系统 | <https://github.com/happyfish100/fastdfs> |
| Log4J | 日志组件 | <http://logging.apache.org/log4j/1.2/> |
| Swagger2 | 接口测试框架 | <http://swagger.io/> |
| sequence | 分布式高效ID生产 | <http://git.oschina.net/yu120/sequence> |
| AliOSS & Qiniu & QcloudCOS | 云存储 | <https://www.aliyun.com/product/oss/><http://www.qiniu.com/><https://www.qcloud.com/product/cos> |
| Protobuf & json | 数据序列化 | <https://github.com/google/protobuf> |
| Jenkins | 持续集成工具 | <https://jenkins.io/index.html> |
| Maven | 项目构建管理 | <http://maven.apache.org/> |

#### 前端技术:

|  |  |  |
| --- | --- | --- |
| **技术** | **名称** | **官网** |
| jQuery | 函式库 | <http://jquery.com/> |
| Bootstrap | 前端框架 | <http://getbootstrap.com/> |
| Bootstrap-table | Bootstrap数据表格 | <http://bootstrap-table.wenzhixin.net.cn/> |
| Font-awesome | 字体图标 | <http://fontawesome.io/> |
| material-design-iconic-font | 字体图标 | <https://github.com/zavoloklom/material-design-iconic-font> |
| Waves | 点击效果插件 | <https://github.com/fians/Waves> |
| zTree | 树插件 | <http://www.treejs.cn/v3/> |
| Select2 | 选择框插件 | <https://github.com/select2/select2> |
| jquery-confirm | 弹出窗口插件 | <https://github.com/craftpip/jquery-confirm> |
| jQuery EasyUI | 基于jQuery的UI插件集合体 | [http://www.jeasyui.com](http://www.jeasyui.com/) |
| React | 界面构建框架 | <https://github.com/facebook/react> |
| Editor.md | Markdown编辑器 | <https://github.com/pandao/editor.md> |
| sanhuAdmin | 后台管理系统模板 | [https://github.com/sanhu/sanhuAdmin](https://github.com/shuzheng/zhengAdmin) |
| autoMail | 邮箱地址自动补全插件 | <https://github.com/shuzheng/autoMail> |
| zheng.jprogress.js | 加载进度条插件 | <https://github.com/shuzheng/zheng.jprogress.js> |
| zheng.jtotop.js | 返回顶部插件 | <https://github.com/shuzheng/zheng.jtotop.js> |

#### 架构图



#### 模块介绍

sanhu-common

Spring+SpringMVC+Mybatis框架集成公共模块，包括公共配置、MybatisGenerator扩展插件、通用BaseService、工具类等。

sanhu-admin

基于bootstrap实现的响应式Material Design风格的通用后台管理系统，zheng项目所有后台系统都是使用该模块界面作为前端展示。

sanhu-ui

各个子系统前台thymeleaf模板，前端资源模块，使用nginx代理，实现动静分离。

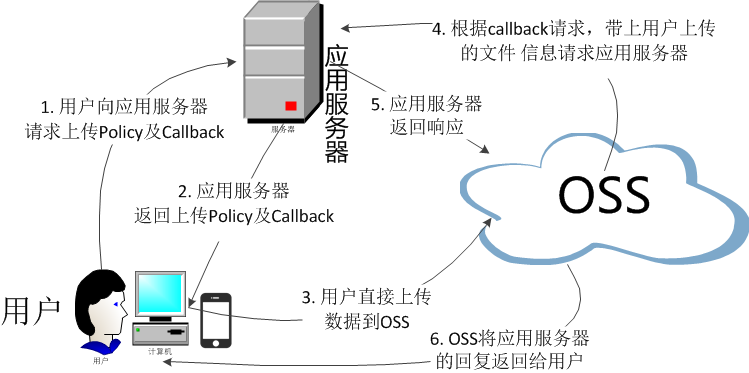
sanhu-upms

本系统是基于RBAC授权和基于用户授权的细粒度权限控制通用平台，并提供单点登录、会话管理和日志管理。接入的系统可自由定义组织、角色、权限、资源等。用户权限=所拥有角色权限合集+用户加权限-用户减权限，优先级：用户减权限>用户加权限>角色权限

sanhu-oss

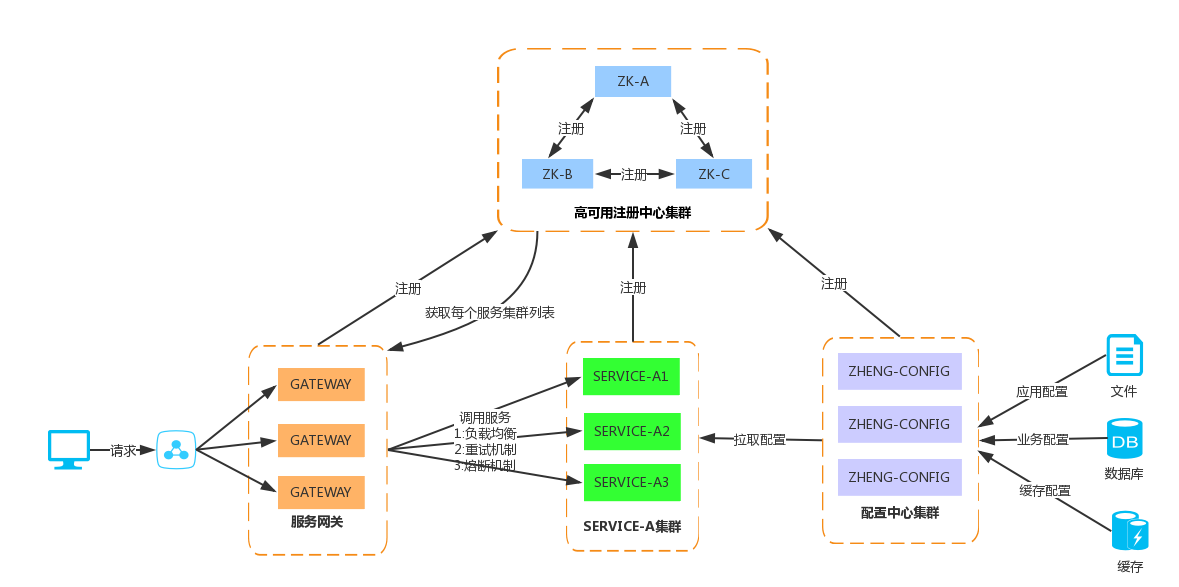
文件存储系统，提供四种方案：

* **阿里云** OSS
* **腾讯云** COS
* **七牛云**
* 本地分布式存储

[](https://github.com/shuzheng/zheng/blob/master/project-bootstrap/aliyun-oss-post-callback.png)

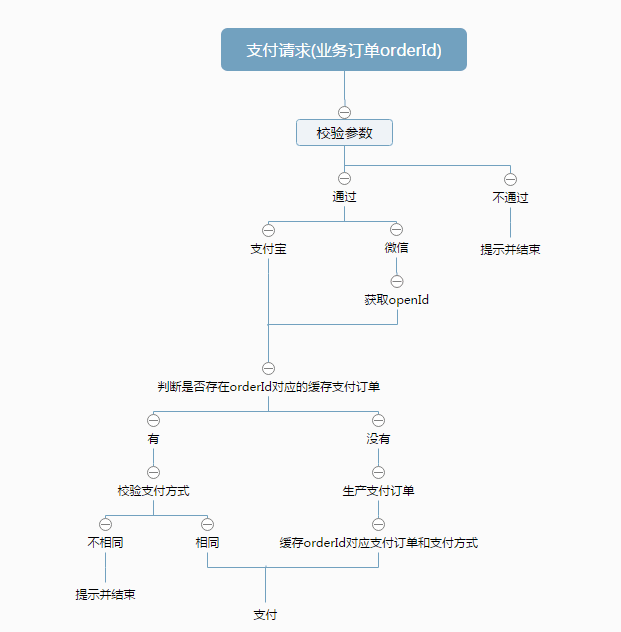
sanhu-api

服务网关，对外暴露统一规范的接口和包装响应结果，包括各个子系统的交互接口、对外开放接口、开发加密接口、接口文档等服务，可在该模块支持验签、鉴权、路由、限流、监控、容错、日志等功能。示例图：

[](https://github.com/shuzheng/zheng/blob/master/project-bootstrap/gateway_config.png)

sanhu-pay

* 一站式支付解决方案，统一下单接口，支持支付宝、微信、网银等多种支付方式。不涉及业务的纯粹的支付平台。
* 统一下单（统一下单接口、统一扫码）、订单管理、数据分析、财务报表、商户管理、渠道管理、对账系统、系统监控。

[](https://github.com/shuzheng/zheng/blob/master/project-bootstrap/zheng-pay.png)

sanhu-ucenter

通用用户管理系统， 实现最常用的用户注册、登录、资料管理、个人中心、第三方登录等基本需求，支持扩展二次开发。

## 环境搭建

#### 开发工具:

* MySql: 数据库
* jetty: 开发服务器
* Tomcat: 应用服务器
* SVN|Git: 版本管理
* Nginx: 反向代理服务器
* Varnish: HTTP加速器
* IntelliJ IDEA: 开发IDE
* PowerDesigner: 建模工具
* Navicat for MySQL: 数据库客户端

#### 开发环境：

* Jdk7+
* Mysql5.5+
* Redis
* Zookeeper
* ActiveMQ
* Dubbo-admin
* Dubbo-monitor

### 工具安装

查看网站环境搭建目录

### 资源下载

* JDK7 <http://www.oracle.com/technetwork/java/javase/downloads/java-archive-downloads-javase7-521261.html>
* Maven <http://maven.apache.org/download.cgi>
* Redis <https://redis.io/download>
* ActiveMQ <http://activemq.apache.org/download-archives.html>
* ZooKeeper <http://www.apache.org/dyn/closer.cgi/zookeeper/>
* Dubbo <http://dubbo.io/Download-zh.htm>
* Elastic Stack <https://www.elastic.co/downloads>
* Nginx <http://nginx.org/en/download.html>
* Jenkins <http://updates.jenkins-ci.org/download/war/>
* dubbo-admin-2.5.3 <http://download.csdn.net/detail/shuzheng5201314/9733652>
* dubbo-admin-2.5.4-SNAPSHOT-jdk8 <http://download.csdn.net/detail/shuzheng5201314/9733657>
* 更多资源请加QQ群

## 开发指南:

* 1、本机安装Jdk7、Mysql、Redis、Zookeeper、ActiveMQ并**启动相关服务**，使用默认配置默认端口即可
* 2、克隆源代码到本地并打开，**推荐使用IntelliJ IDEA**，本地编译并安装到本地maven仓库

### 修改本地Host

* 127.0.0.1 ui.sanhu.cn
* 127.0.0.1 power.sanhu.cn
* 127.0.0.1 cms.sanhu.cn
* 127.0.0.1 pay.sanhu.cn
* 127.0.0.1 ucenter.sanhu.cn
* 127.0.0.1 api.sanhu.cn
* 127.0.0.1 oss.sanhu.cn
* 127.0.0.1 config.sanhu.cn
* 127.0.0.1 zkserver
* 127.0.0.1 rdserver
* 127.0.0.1 dbserver
* 127.0.0.1 mqserver

### 编译流程

maven编译安装sanhu/pom.xml文件即可

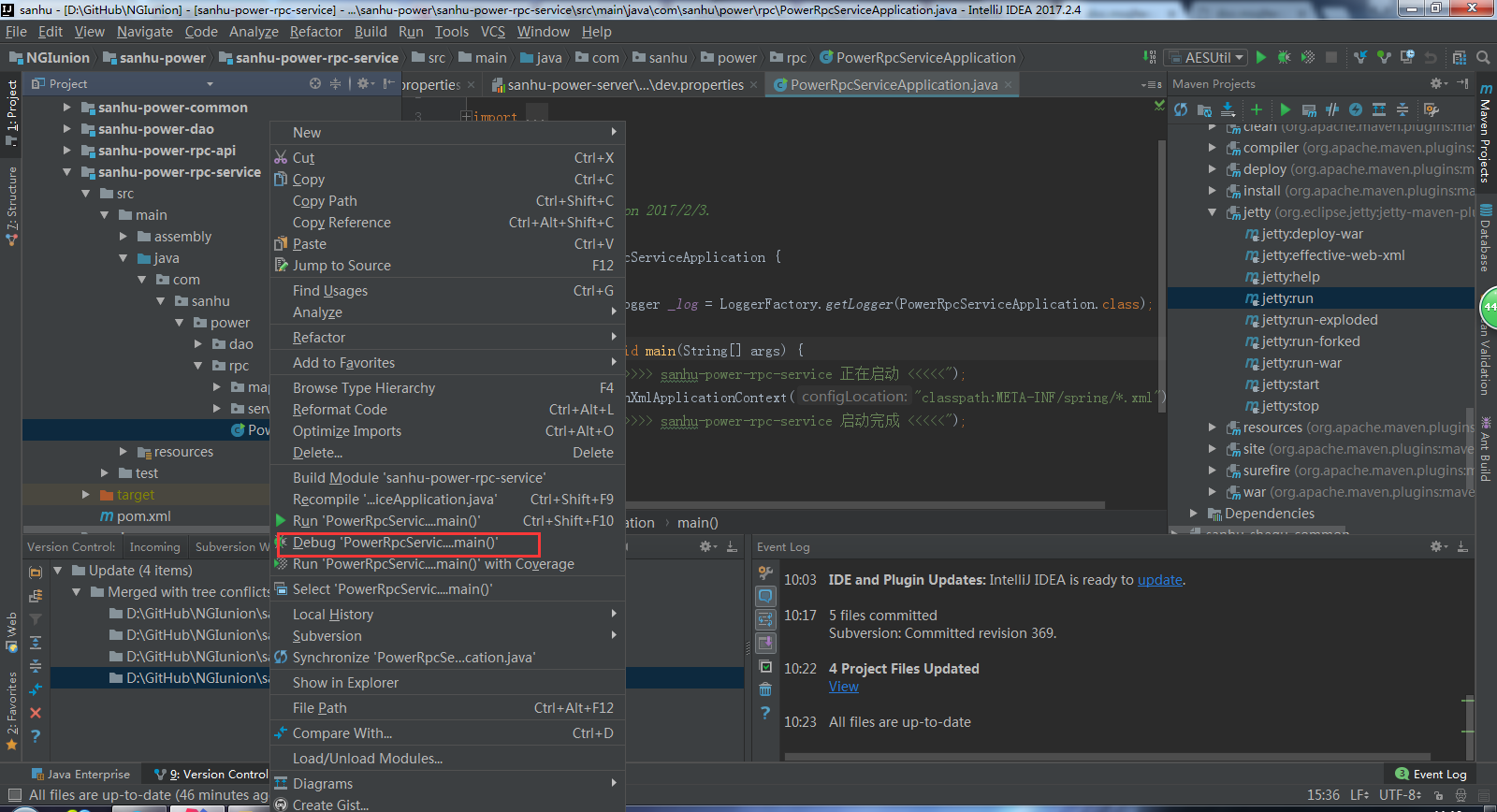
### 启动顺序（后台）

准备工作

* 新建sanhu数据库，导入project-datamodel文件夹下的sanhu.sql
* 修改各dao模块和rpc-service模块的redis.properties、jdbc.properties、generator.properties数据库连接等配置信息，其中master.redis.password、master.jdbc.password、slave.jdbc.password、generator.jdbc.password密码值使用了AES加密，请使用com.zheng.common.util.AESUtil工具类修改这些值
* 启动Zookeeper、Redis、ActiveMQ、Nginx（配置文件参考project-tools/nginx下的\*.conf文件）

**sanhu-power**

* 首先启动 sanhu-power-rpc-service(直接运行src目录下的PowerRpcServiceApplication#main方法启动) => sanhu-power-server(jetty)，然后按需启动对应子系统xxx的sanhu-xxx-rpc-service(main方法) => sanhu-xxx-webapp(jetty)



* 访问 [http://power.sanhu.cn:1111/](http://upms.zhangshuzheng.cn:1111/)，子系统菜单已经配置到sanhu-power权限中，不用直接访问子系统，默认帐号密码：admin/123456
* 登录成功后，可在右上角切换已注册系统访问

**sanhu-oss**

* 首先启动sanhu-oss-web服务
* 开发阶段，如果sanhu-oss-web没有公网域名，推荐使用ngrok内网穿透工具，为开发环境提供公网域名，实现上传回调
* 启动nginx代理sanhu-ui静态资源

### 开发演示

* 创建数据表（建议使用PowerDesigner）
* 直接运行对应项目dao模块中的generator.main()，可自动生成单表的CRUD功能和对应的model、example、mapper、service代码

生成的model和example均已实现Serializable接口，支持分布式

已包含抽象类BaseServiceImpl，只需要继承抽象类并传入泛型参数，即可默认实现mapper接口所有方法，特殊需求直接扩展即可

BaseServiceImpl默认已实现四种根据条件分页接口

selectByExampleWithBLOBsForStartPage()

selectByExampleForStartPage()

selectByExampleWithBLOBsForOffsetPage()

selectByExampleForOffsetPage()

BaseServiceImpl方法根据读写操作自动切换主从数据源，继承的扩展接口，可手动通过DynamicDataSource.setDataSource(DataSourceEnum.XXX.getName())指定数据源

* 启动流程：优先rcp-service服务提供者，再启动其他webapp
* 扩展流程：可扩展和拆分rpc-api和rpc-service模块，可按微服务拆分或场景拆分

### 部署方式

* war包项目：使用tomcat等web容器启动
* rpc-service服务提供者jar包：将打包后的sanhu-xxx-rpc-service-assembly.tar.gz文件解压，使用bin目录的管理脚本运行即可，支持优雅停机。

### 框架规范约定

约定优于配置(convention over configuration)，此框架约定了很多编程规范，下面一一列举：

- service类，需要在叫名`service`的包下，并以`Service`结尾，如`DemoServiceImpl`

- controller类，需要在以`controller`结尾的包下，类名以Controller结尾，如`DemoController.java`，并继承`BaseController`

- spring task类，需要在叫名`task`的包下，并以`Task`结尾，如`TestTask.java`

- mapper.xml，需要在名叫`mapper`的包下，并以`Mapper.xml`结尾，如`DemoMapper.xml`

- mapper接口，需要在名叫`mapper`的包下，并以`Mapper`结尾，如`DemoMapper.java`

- model实体类，需要在名叫`model`的包下，命名规则为数据表转驼峰规则，如`Demo.java`

- spring配置文件，命名规则为`applicationContext-\*.xml`

- 类名：首字母大写驼峰规则；方法名：首字母小写驼峰规则；常量：全大写；变量：首字母小写驼峰规则，尽量非缩写

- springmvc配置加到对应模块的`springMVC-servlet.xml`文件里

- 配置文件放到`src/main/resources`目录下

- 静态资源文件放到`src/main/webapp/resources`目录下

- jsp文件，需要在`/WEB-INF/jsp`目录下

- `RequestMapping`和返回物理试图路径的url尽量写全路径，如：`@RequestMapping("/manage")`、`return "/manage/index"`

- `RequestMapping`指定method

- 模块命名为`项目`-`子项目`-`业务`，如`sanhu-shequ-admin`

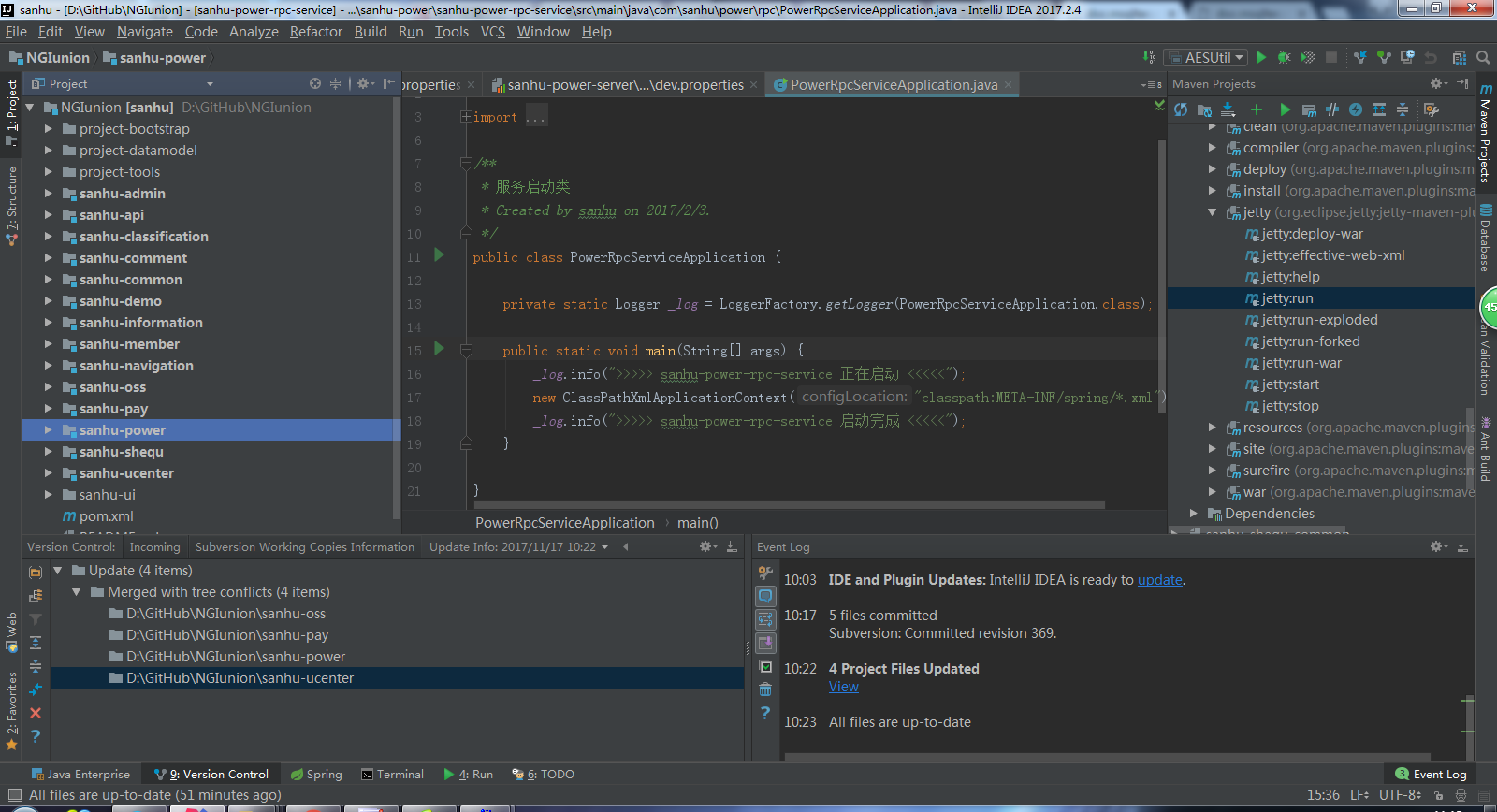
- 数据表命名为：`子系统`\_`表`，如`shequ\_article`

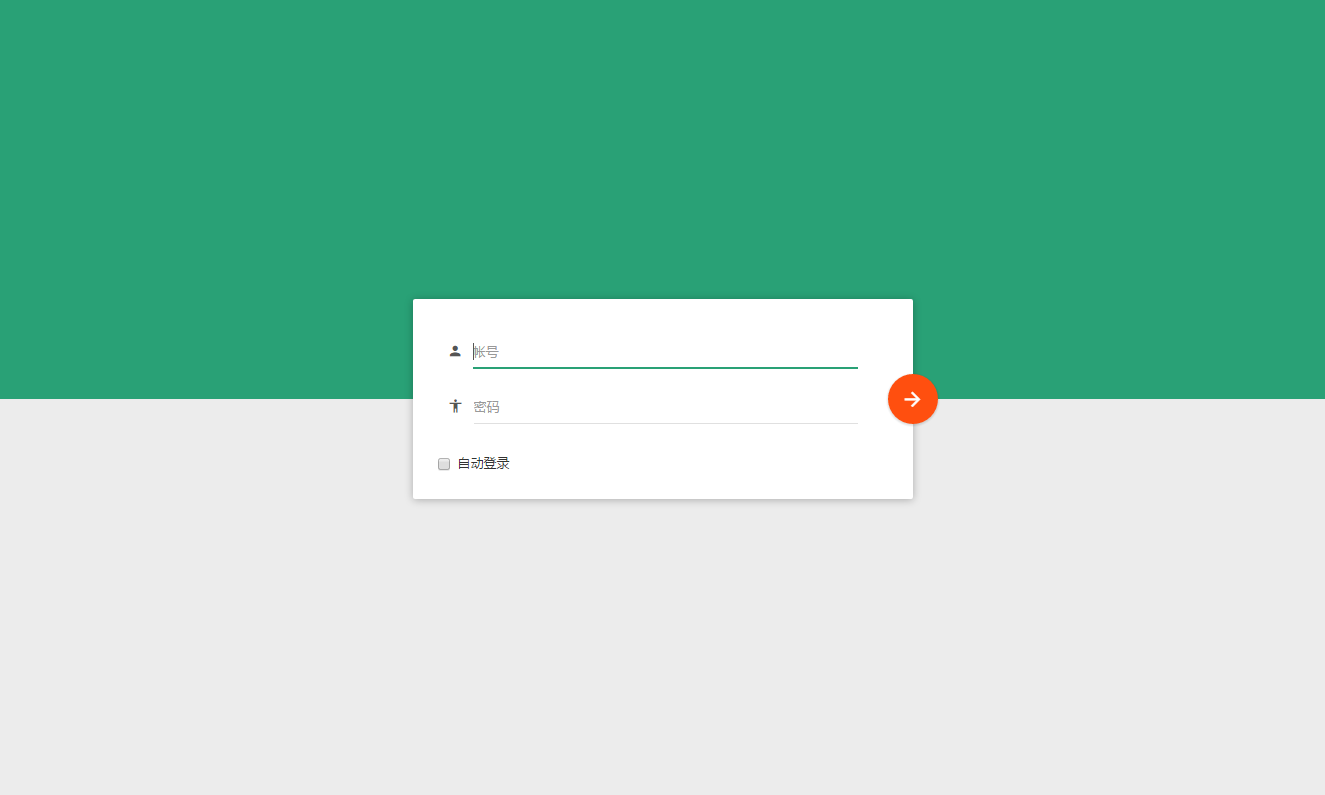
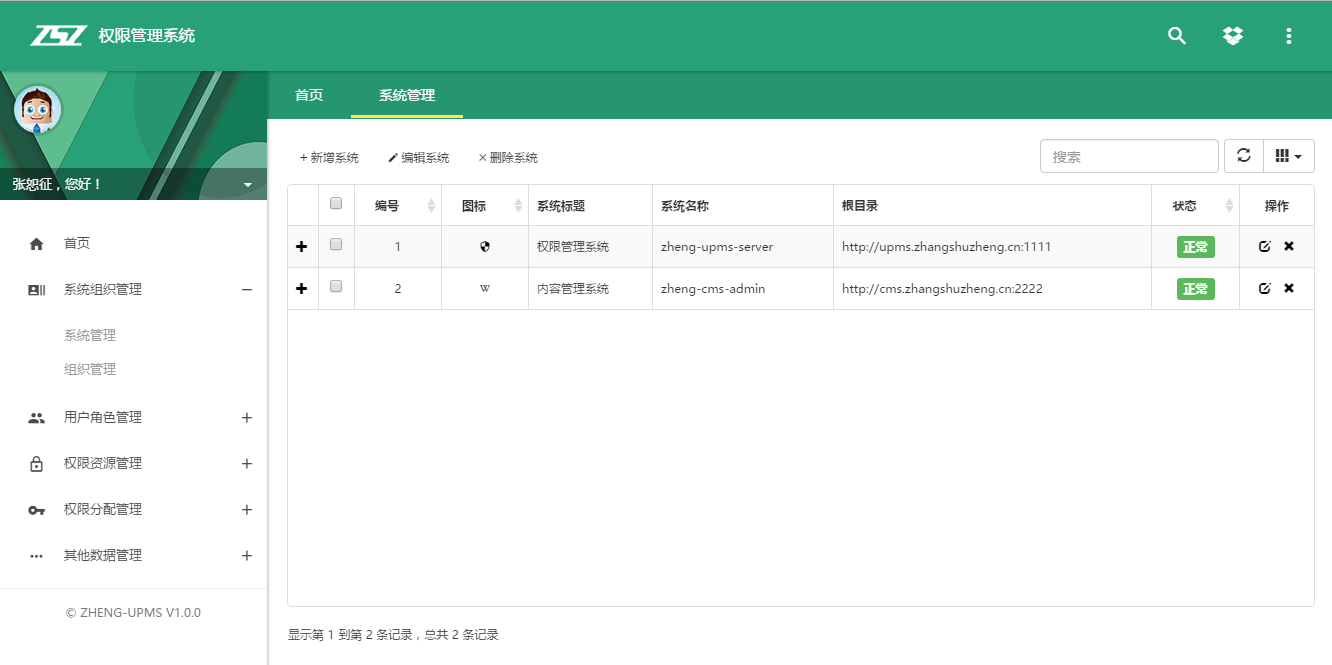
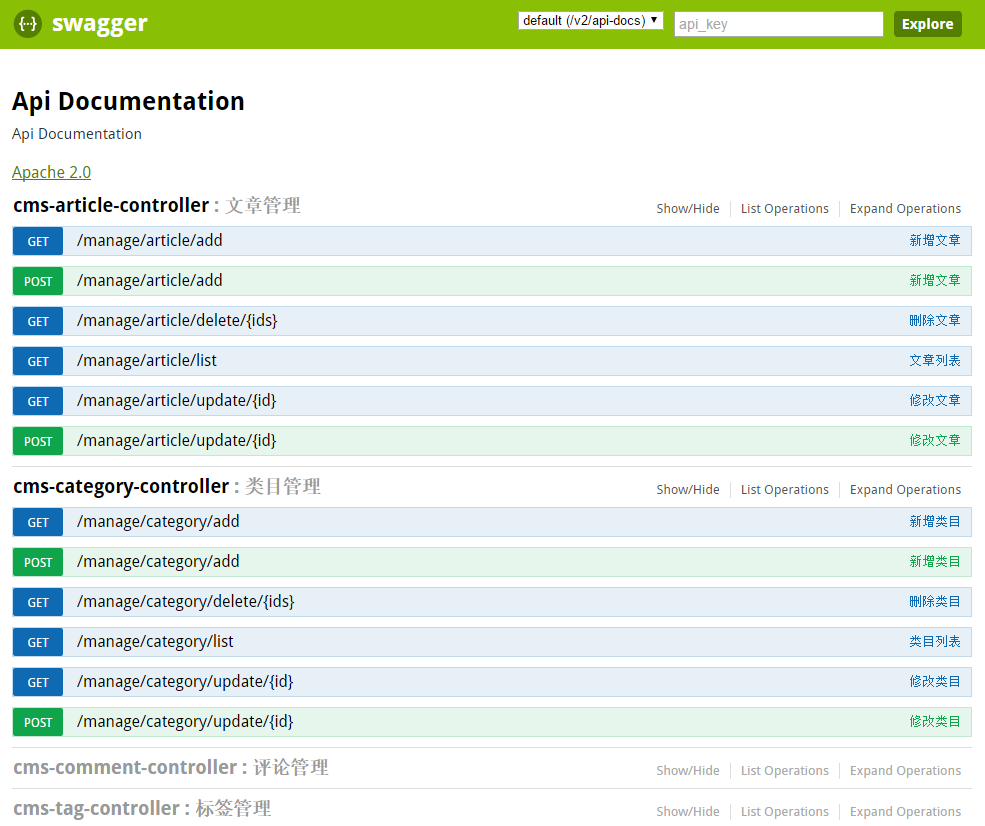
- 更多规范，参考[[阿里巴巴Java开发手册] http://git.oschina.net/shuzheng/zheng/attach\_files

## 演示地址

演示地址： [http://power.sanhu.cn/](http://115.28.229.85:8014/)

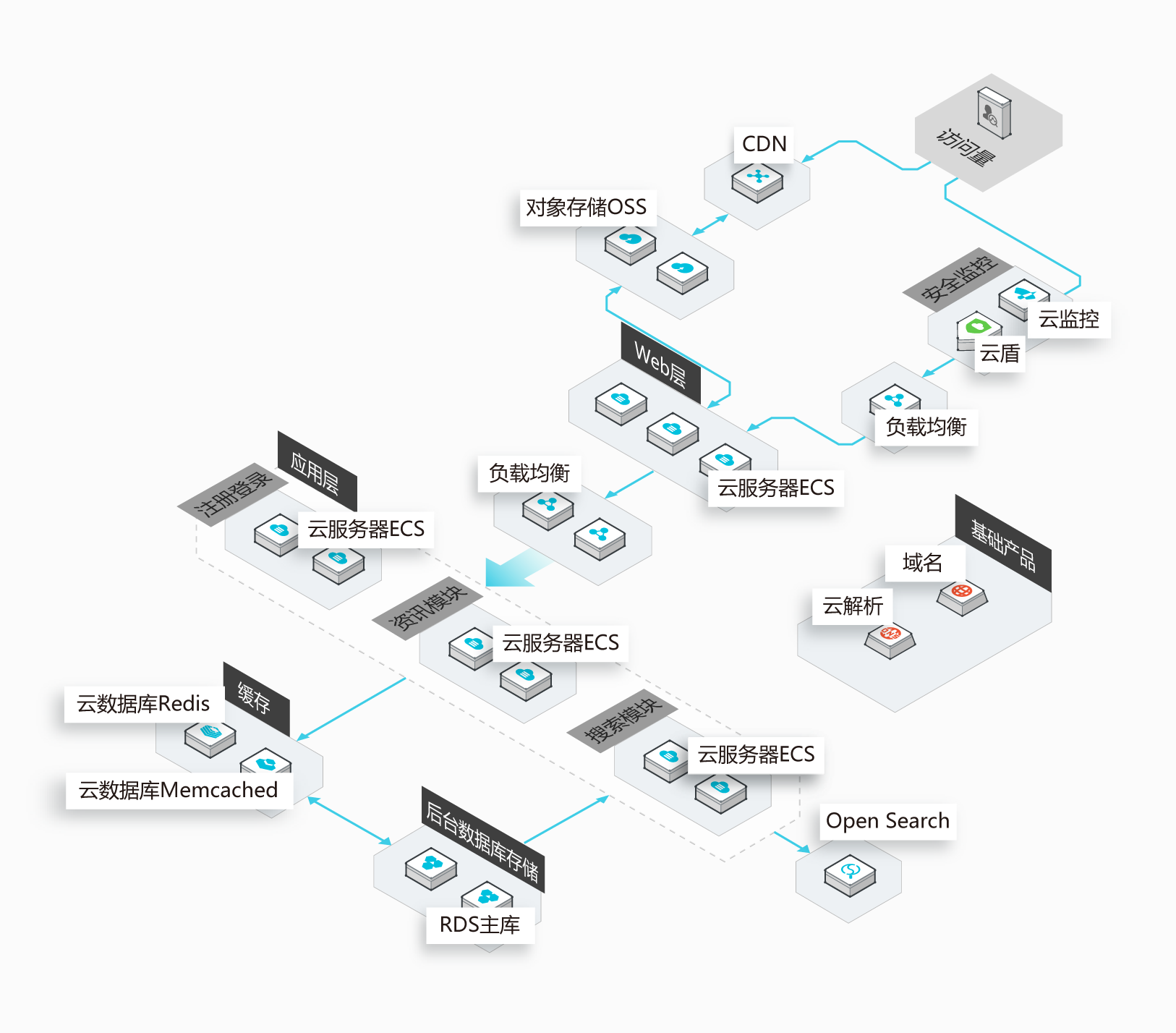
### 预览图



[](https://github.com/shuzheng/zheng/blob/master/project-bootstrap/zheng-login.png)[](https://github.com/shuzheng/zheng/blob/master/project-bootstrap/zheng-upms.png)[](https://github.com/shuzheng/zheng/blob/master/project-bootstrap/zheng-cms.png)[](https://github.com/shuzheng/zheng/blob/master/project-bootstrap/api.png)

### 数据模型

### 拓扑图

[](https://github.com/shuzheng/zheng/blob/master/project-bootstrap/distributedSystem.png)

### 常见问题

* Eclipse下，dubbo找不到dubbo.xsd报错，不影响使用，如果要解决，可参考 <http://blog.csdn.net/gjldwz/article/details/50555922>
* 报sanhu-xxx.jar包找不到,请按照文档编译顺序，将源代码编译并安装到本地maven仓库
* sanhu-cms-admin启动卡住：因为没有启动activemq
* sanhu-upms-server访问报session不存在：因为没有启动redis服务
* 界面没有样式：因为sanhu-admin没有编译安装到本地仓库
* linux下执行rpc-service脚本报“bin/bash^M 坏的解释器”，使用sed -i 's/\r$//' filename删除脚本中\r字符

### 优秀文章和博客

* [创业互联网公司如何搭建自己的技术框架](http://shuzheng5201314.iteye.com/blog/2330151)
* [微服务实战](https://segmentfault.com/a/1190000004634172)
* [单点登录原理与简单实现](http://shuzheng5201314.iteye.com/blog/2343910)
* [ITeye论坛关于权限控制的讨论](http://www.iteye.com/magazines/82)
* [RBAC新解：基于资源的权限管理(Resource-Based Access Control)](http://globeeip.iteye.com/blog/1236167)
* [网站架构经验随笔](http://jinnianshilongnian.iteye.com/blog/2289904)
* [支付系统架构](http://shuzheng5201314.iteye.com/blog/2355431)
* [Spring整合JMS](http://elim.iteye.com/blog/1893038)
* [跟我学Shiro目录贴](http://jinnianshilongnian.iteye.com/blog/2018398)
* [跟我学SpringMVC目录汇总贴](http://jinnianshilongnian.iteye.com/blog/1752171)
* [跟我学spring3 目录贴](http://jinnianshilongnian.iteye.com/blog/1482071)
* [跟我学OpenResty(Nginx+Lua)开发目录贴](http://jinnianshilongnian.iteye.com/blog/2190344)
* [Redis中文网](http://www.redis.net.cn/)
* [读懂Redis并配置主从集群及高可用部署](http://mp.weixin.qq.com/s?__biz=MzIxNTYzOTQ0Ng==&mid=2247483668&idx=1&sn=cd31574877d38cf7ff9c047b86c9bf23&chksm=979475eda0e3fcfb6b5006bcd19c5a838eca9e369252847dbdf97820bf418201dd75c1dadda3&mpshare=1&scene=23&srcid=0117KUiiITwi2ETRan16xRVg#rd)
* [Redis哨兵-实现Redis高可用](http://redis.majunwei.com/topics/sentinel.html)
* [ELK(ElasticSearch, Logstash, Kibana)搭建实时日志分析平台](http://www.open-open.com/lib/view/open1451801542042.html)
* [Nginx基本功能极速入门](http://xxgblog.com/2015/05/17/nginx-start/)
* [mybatis-genarator 自定义插件](https://my.oschina.net/alexgaoyh/blog/702791)
* [Elasticsearch权威指南（中文版）](https://es.xiaoleilu.com/510_Deployment/20_hardware.html)
* [springMVC对简单对象、Set、List、Map的数据绑定和常见问题.](http://blog.csdn.net/z_dendy/article/details/12648641)
* [如何细粒度地控制你的MyBatis二级缓存](http://blog.csdn.net/luanlouis/article/details/41800511)
* [做个男人，做个成熟的男人，做个有城府的男人](http://shuzheng5201314.iteye.com/blog/1387820)

### 在线小工具

* [在线Cron表达式生成器](http://cron.qqe2.com/)
* [在线工具 - 程序员的工具箱](http://tool.lu/)

### 在线文档

* [JDK7英文文档](http://tool.oschina.net/apidocs/apidoc?api=jdk_7u4)
* [Spring4.x文档](http://spring.oschina.mopaas.com/)
* [Mybatis3官网](http://www.mybatis.org/mybatis-3/zh/index.html)
* [Dubbo官网](http://dubbo.io/)
* [Nginx中文文档](http://tool.oschina.net/apidocs/apidoc?api=nginx-zh)
* [Freemarker在线手册](http://freemarker.foofun.cn/)
* [Velocity在线手册](http://velocity.apache.org/engine/devel/developer-guide.html)
* [Bootstrap在线手册](http://www.bootcss.com/)
* [Git官网中文文档](https://git-scm.com/book/zh/v2)
* [Thymeleaf](http://www.thymeleaf.org/doc/tutorials/3.0/thymeleafspring.html)