

微政云基础引擎概要设计

1.引言

1.1 编写目的

在对微政云所有产品需求了解的前提下，已经将系统的职责、微政云以及系统的扩展性要求做了详细的分析。本文档旨在将基础引擎系统的需求详尽阐述及说明。

1.2 项目背景

在当前推进“互联网+政务服务”的指导下，互联网越来越多的被人们接受，提高了办事的效率。传统的方式正在逐渐过渡到“互联网”的方式，所以本项目是为了提高政府办事效率，推进政务信息共享，加快新型智慧城市建设而设。

1.3 定义

- MYSQL: 系统服务器使用的数据库。
- SQL: 一种用于访问查询数据库的语言。
- RPC: 远程过程调用协议，它是一种通过网络从远程计算机程序上请求服务，而不需要了解底层网络技术的协议。RPC协议假定某些传输协议的存在，如TCP或UDP，为通信程序之间携带信息数据。在OSI网络通信模型中，RPC跨越了传输层和应用层。RPC使得开发包括网络分布式多程序在内的应用程序更加容易。
- TOMCAT: JAVA服务器运行时容器。
- JDK: Java 语言的软件开发工具包,本项目使用版本为 1.8.0.101
- THRIFT: thrift是一个软件框架，用来进行可扩展且跨语言的服务的开发。它结合了功能强大的软件堆栈和代码生成引擎，以构建在 C++, Java, Go,Python, PHP, Ruby, Erlang, Perl, Haskell, C#, Cocoa, JavaScript, Node.js, Smalltalk, and OCaml 这些编程语言间无缝结合的、高效的服务。
- Nginx: 是一个高性能的HTTP和反向代理服务器，也是一个IMAP/POP3/SMTP服务器。Nginx是由Igor Sysoev为俄罗斯访问量第二的Rambler.ru站点开发的，第一个公开版本0.1.0发布于2004年10月4日。其将源代码以类BSD许可证的形式发布，因它的稳定性、丰富的功能集、示例配置文件和低系统资源的消耗而闻名

2. 任务概述

2.1 运行环境

系统由多个语言混合构成整体异构解决方案，目前使用到的语言：php，java。php作为客户端调用java提供的基础服务，来实现所有的业务功能。

系统最终运行在Linux主机上，部署环境要求：

- 软件：
 - jdk1.8.0.101
 - tomcat8
 - wzt-engine.war (程序部署包)
- 硬件：
 - 4核
 - 8G内存
 - 带宽下行1M以上
 - 硬盘70G以上

2.2 需求概述

为了支持日益增加的业务量，系统的架构必须有横向扩展的能力。在不影响整体前端业务的情况下，做到动态对服务器扩容，以便支撑更大的流量。并且，本系统提供了所有前端业务需要的数据库操作，屏蔽了前端的直接数据操作，可以做到监控，以及未来可能的限流降级功能。目前主要大的业务模块分别是：

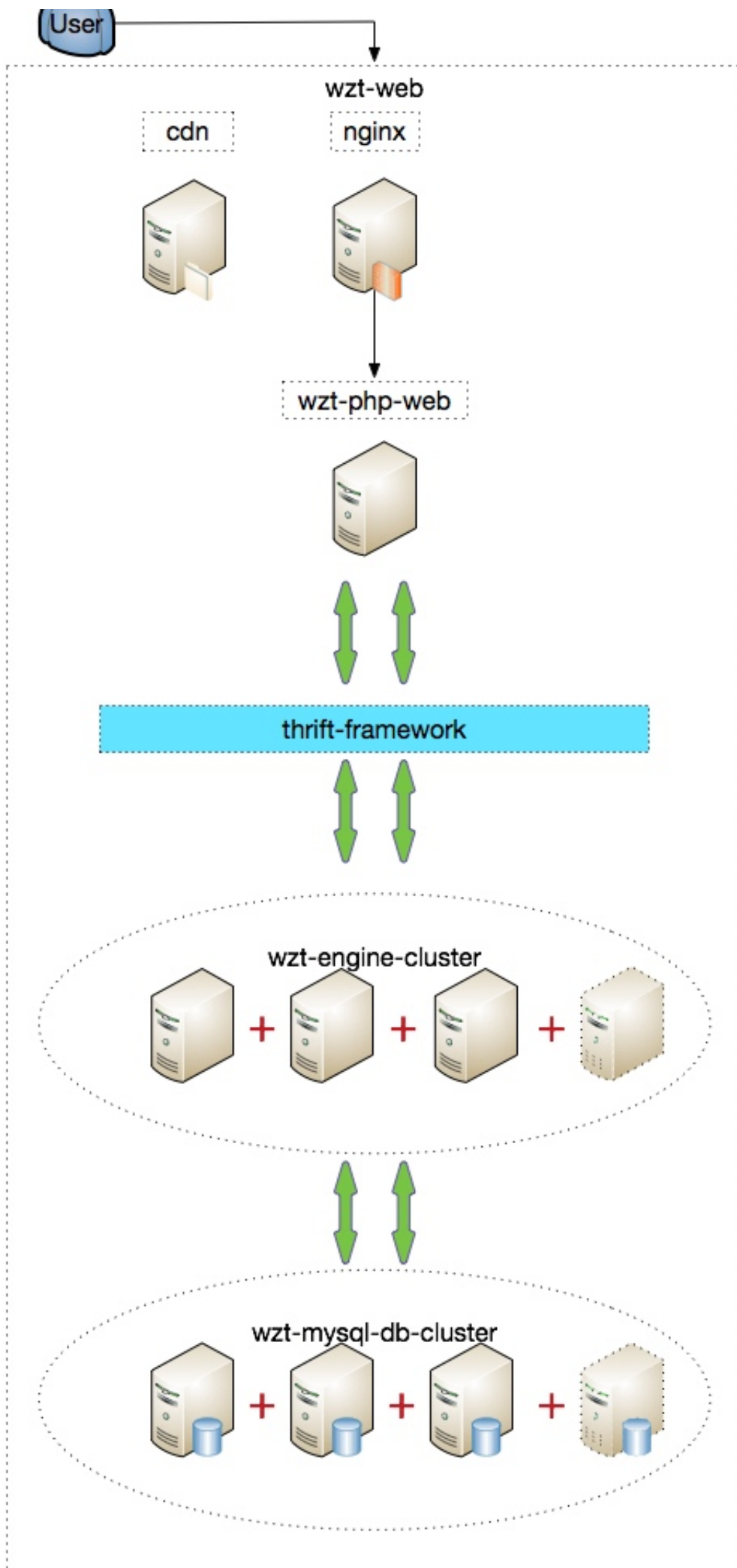
- 系统通用账户及权限管理
- 系统账户分组树及所属关系管理
- 系统资源统一管理
- 系统角色统一管理
- 微信公众号的绑定以及数据定向获取
- 账户增值服务管理

在后续文档中，会一一阐述这些需求，以及架构设计方式。

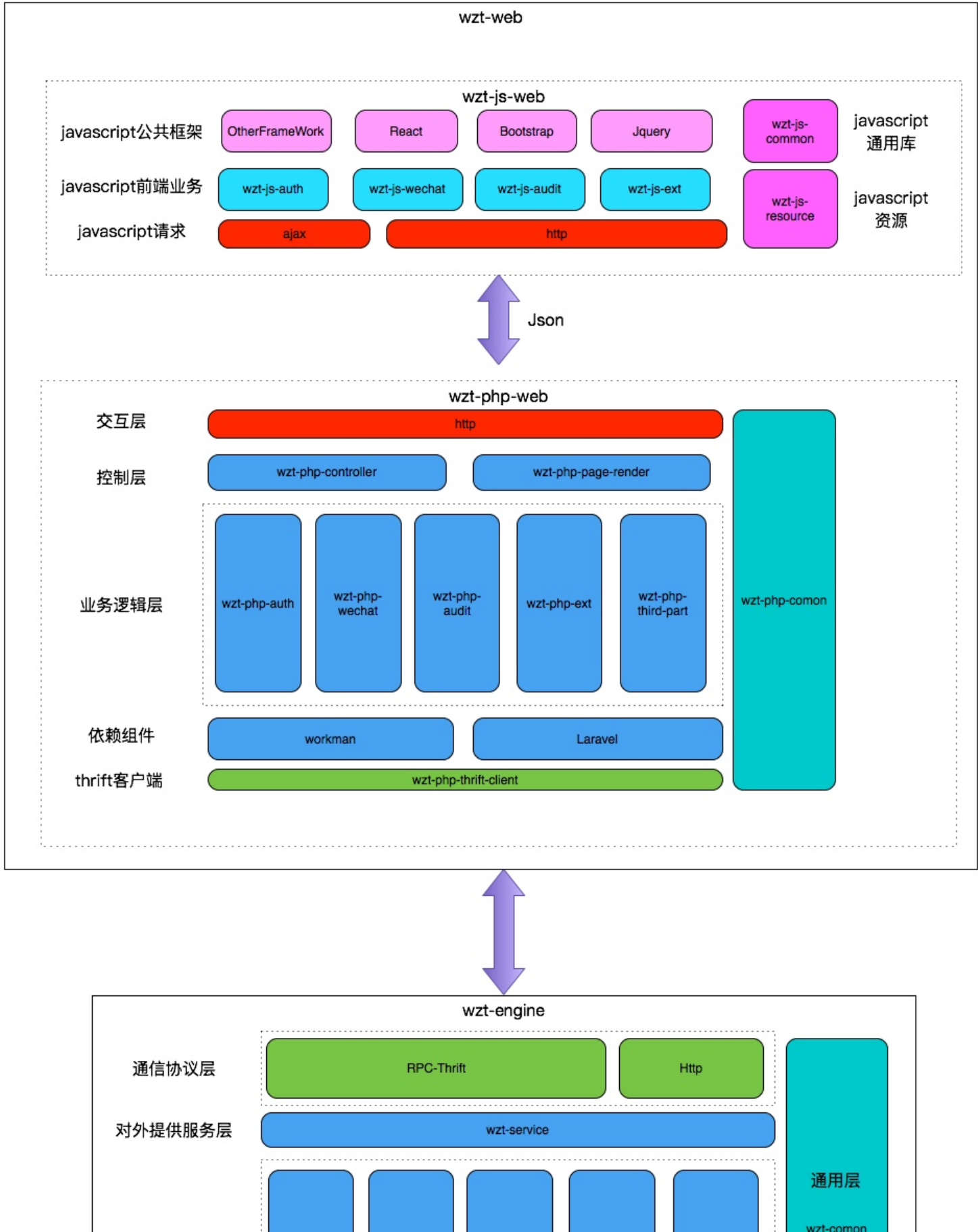
3 总体架构设计

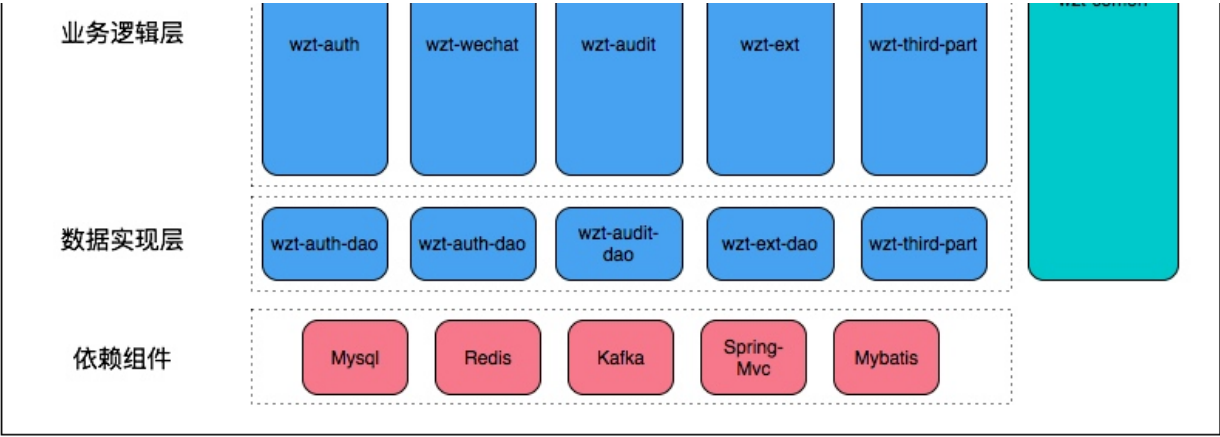
3.1 部署图





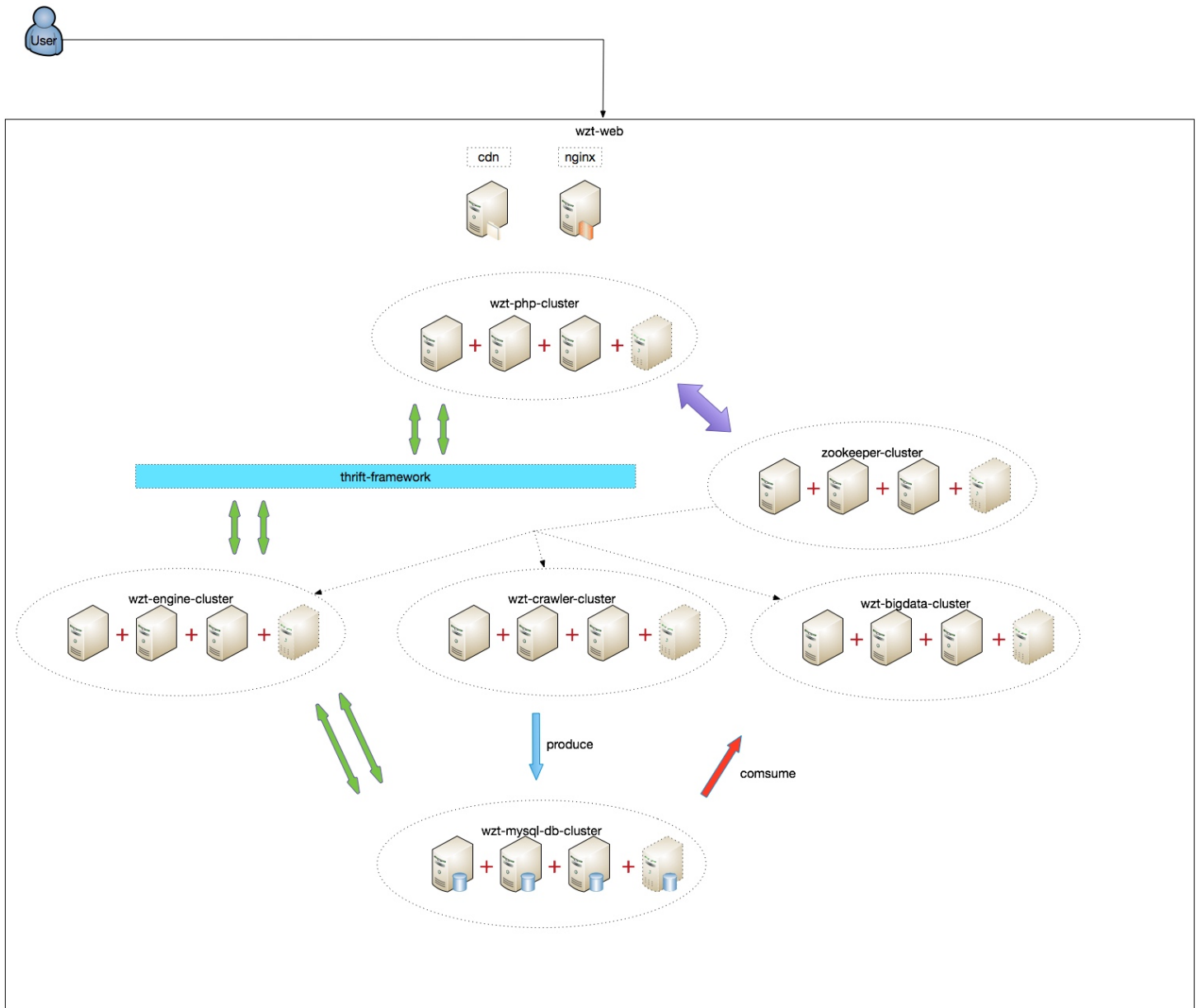
3.2 架构图





3.3 系统交互方式说明

nginx作为整个前端业务的流量入口，在配置了反向代理的情况下，可以动态将请求均摊到wzt-php-web(php前端业务入口),然后wzt-php-web通过thrift协议调用响应的基础服务(本系统框架wzt-engine)，从而得到业务的处理。考虑到对后期系统的扩展性，系统在随着业务量增长之后，每个节点都必须有动态扩展的能力，下一个版本的预计部署结构如下：



届时系统节点以及组织方式会根据具体的业务情况进行动态的调整。

3.4 模块设计

- wzt-auth

包含所有账户授权基础服务

- i. 资源相关操作

- 增加资源
- 删除资源
- 修改资源
- 查询资源

- ii. 角色相关操作

- 增加角色
- 删除角色
- 修改角色
- 查询角色信息

iii. 分组相关操作

- 增加分组
- 删除分组
- 修改分组
- 查询分组信息

iv. 账户相关操作

- 增加账户
- 删除账户
- 修改账户
- 查询账户信息

v. 角色-资源关联操作

- 增加角色可访问资源
- 修改角色可访问资源
- 删除角色可访问资源
- 查询角色可访问资源

vi. 账户-角色关联操作

- 绑定账户角色
- 修改账户角色
- 删除账户角色
- 查询账户角色

vii. 账户-分组关联操作

- 指定账户分组
- 修改账户分组
- 删除账户分组
- 查询账户分组

- wzt-common

通用层，包括一些工具类，常用的注解类

- wzt-dao

数据库访问层，所有数据库操作，按模块分包，为所属模块提供数据库操作

- wzt-rpc

远程调用层，提供远程调用服务，以及调用远程服务的客户端，和PHP进行交互

- wzt-service

服务包装层，对各个基础服务模块的组装，完成业务

- i. 所有用户

- 用户登录，获取可以访问的资源列表
 - 用户调用资源，查看是否有对应资源权限

- ii. 公众号管理员服务

- 创建账户，可以绑定用户为：一审人员，二审人员，三审人员，或者自定义的其他角色服务
 - 接受区域管理员的邀请绑定，成为该区域管理员的下级，可以收到来自该区域管理员的广播消息
 - 主动解除绑定，并通知区域管理员(不再接受该区域管理员的广播消息)

- iii. 区域管理员服务（分级管理员）

- 创建账户，可以绑定该用户为：下一级区域管理员，公众号管理员，有来自超级管理员的创建数量限制，可以分配配额给下一级区域管理员。
 - 广播功能(素材,消息等)
 - 邀请绑定指定公众号管理员
 - 主动解除指定自己管辖分组中的公众号管理员，区域管理员

- iv. 超级管理员服务

- 创建角色，并制定角色的权限范围(资源列表)
 - 创建账户，可以绑定该用户为：区域管理员，公众号管理员
 - 分配指定账户可以创建账户的数量
 - 修改指定账户的角色
 - 强制增加/删除/修改 某账户的分组

- v. 系统内置服务

- 加载资源树
- 加载角色列表
- 加载分组树

- wzt-web

对外服务，主要暴露http服务， 包括日后的微信接入， 监控页面等,直接与外部交互

4. 数据结构设计

4.1 数据库结构设计

- 账户表 wzt_account

列名	字段类型	备注
id	BIGINT UNSIGNED	自增主键
nickname	VARCHAR	昵称
role_id	BIGINT UNSIGNED	角色代码
username	VARCHAR	用户名
passwd	VARCHAR	密码
salt	VARCHAR	
phone_number	VARCHAR	手机号
email	VARCHAR	电子邮箱
remark	VARCHAR	备注
joinip	VARCHAR	注册ip
lastip	VARCHAR	最后一次登录ip
last_visit_time	INT	最后一次访问时间
sex	TINYINT	性别， 0-未知， 1-男 , 2-女
status	TINYINT	账户状态， 0-正常， 1-冻结
create_time	INT	创建时间

update_time	INT	最后一次更新时间
is_deleted	TINYINT	是否被逻辑删除 0 -否 1-是

• 账户配置表 wzt_account_config

列名	字段类型	备注
id	BIGINT UNSIGNED	自增主键
create_time	INT	创建时间
update_time	INT	最后一次更新时间
account_id	BIGINT	账户id
is_deleted	TINYINT	是否被逻辑删除 0 -否 1-是
create_cnt_remain	INT UNSIGNED	剩余公众号账户创建数量
create_cnt_total	INT UNSIGNED	可以创建公众号账户创建总数量

• 账户公众号信息表 wzt_account_wechat

列名	字段类型	备注
id	BIGINT UNSIGNED	自增主键
create_time	INT	创建时间
update_time	INT	最后一次更新时间
is_deleted	TINYINT	是否被逻辑删除 0 -否 1-是
gzh_account_id	VARCHAR	微信公众号id
status	TINYINT	状态 0- 正常 1- 未绑定 2-已绑定 3-冻结
account_id	BIGINT UNSIGNED	微政通账号id
token	VARCHAR	公众号token
access_token	VARCHAR	
name	VARCHAR	
secret	VARCHAR	

auth_refresh_token	VARCHAR	
account_username	VARCHAR	微政通用户名，冗余字段,只为了方便查询
gzh_name	VARCHAR	公众号名字

- 账户分组信息表 wzt_group

列名	字段类型	备注
id	BIGINT	自增主键
create_time	INT	创建时间
update_time	INT	更新时间
is_deleted	TINYINT	是否被逻辑删除 0-否， 1-是
parent_id	BIGINT	父节点id
name	VARCHAR	节点名称
group_tree_path	VARCHAR	分组节点树形路径

- 分组公众号信息表 wzt_group_gzh

列名	字段类型	备注
id	BIGINT UNSIGNED	自增主键
create_time	INT	创建时间
update_time	INT	更新时间
is_deleted	TINYINT	是否被逻辑删除 0-否， 1-是
group_admin_name	VARCHAR	组管理员名字
group_admin_account_id	BIGINT UNSIGNED	组管理员id
group_id	BIGINT UNSIGNED	分组id
gzh_account_id	VARCHAR	公众号id
gzh_name	VARCHAR	公众号名字
gzh_admin_id	BIGINT	公众众号管理员id

gzh_admin_name	VARCHAR	公众号管理员名字
gs_nickname_id	BIGINT UNSIGNED	公众号在清博账号的id

- 分组邀请表 wzt_group_invitation

列名	字段类型	备注
id	BIGINT UNSIGNED	自增主键
create_time	INT	创建时间
update_time	INT	更新时间
is_deleted	TINYINT	是否被逻辑删除 0-否， 1-是
group_id	BIGINT UNSIGNED	分组id
group_name	VARCHAR	组名字
group_admin_account_id	BIGINT UNSIGNED	组管理员id
group_admin_name	VARCHAR	邀请人名字
invite_account_id	BIGINT	被邀请者id
invite_account_name	VARCHAR	被邀请者名字
invite_gzh_id	VARCHAR	邀请的公众号id
invite_gzh_name	VARCHAR	邀请的公众号名字
status	TINYINT UNSIGNED	状态 0-发出邀请 1- 已经接受 2-已经拒绝

- 清博账户信息表 wzt_gsdata_account

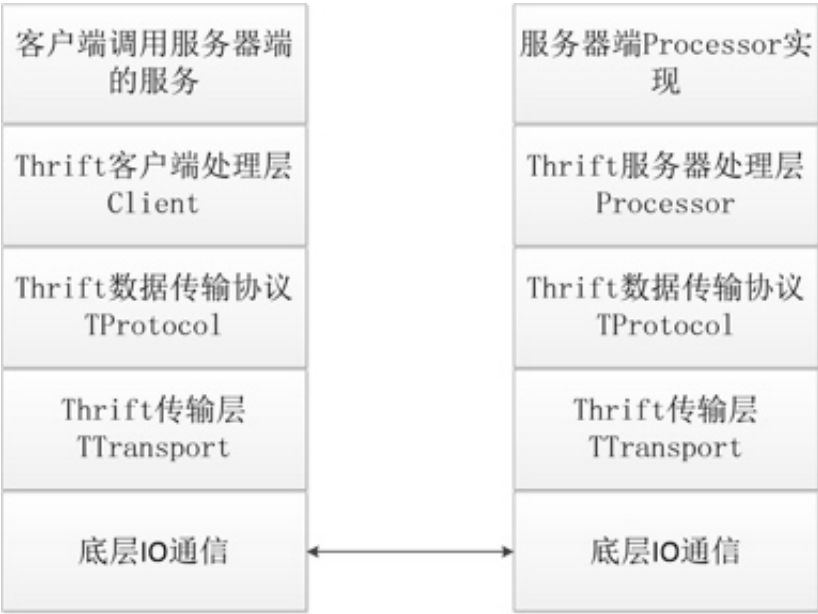
列名	字段类型	备注
id	BIGINT	自增主键
create_time	INT	创建时间
update_time	INT	更新时间
is_deleted	TINYINT	是否被逻辑删除 0-否， 1-是
account_id	BIGINT	微政通账号id

gs_group_id	BIGINT	清博分组id
gs_nickname_id	BIGINT	清博账号id
gs_wx_nickname	VARCHAR	清博微信昵称
gs_wx_name	VARCHAR	清博微信公众号账号
gs_wx_title	VARCHAR	最新的一篇文章标题
gs_wx_url	VARCHAR	最新文章的连接
gs_url_posttime	DATETIME	最新的文章发布时间

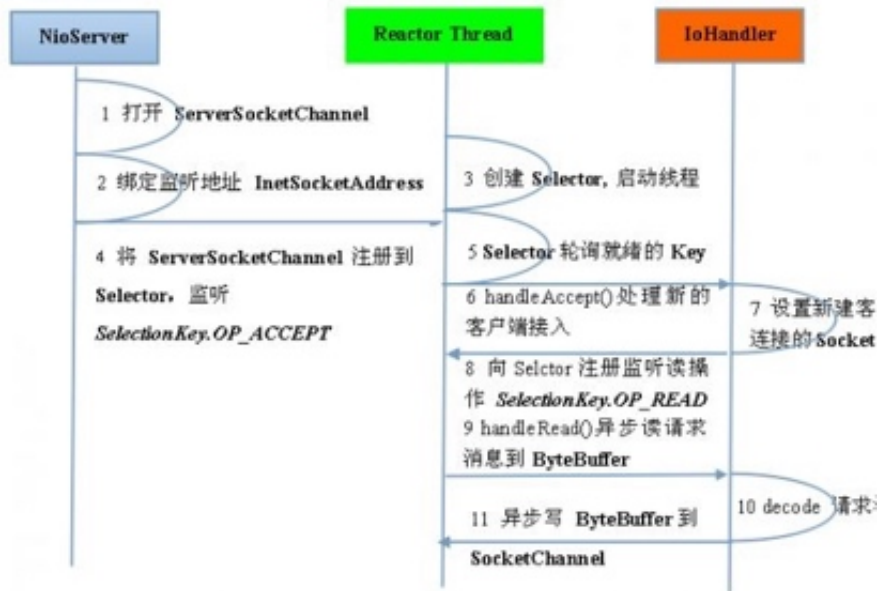
5. 接口设计

5.1 Rpc远程调用设计

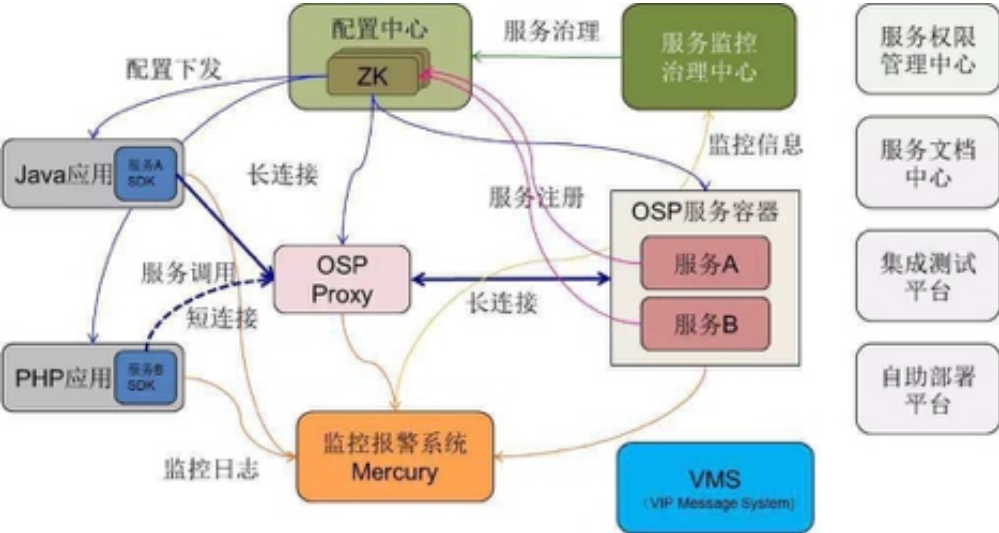
系统thrift通信结构图:



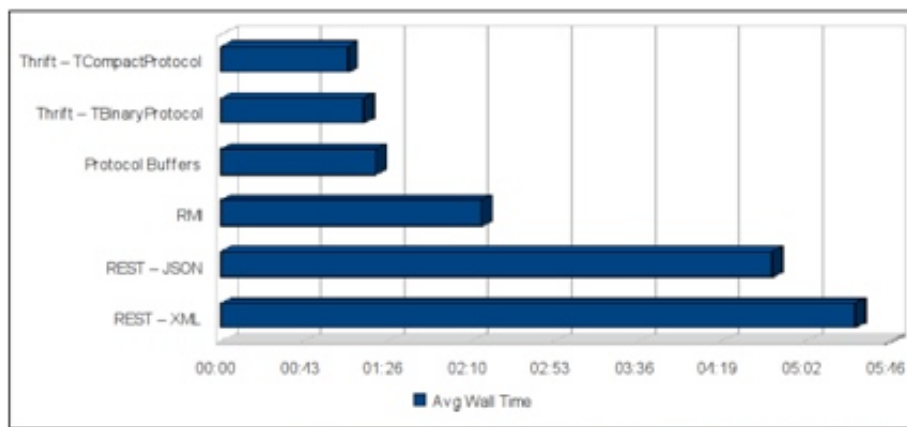
thrift内部通信原理图:



thrift的异构方案为未来的扩展提高了极大的可能性，以及健壮性,未来的整体系统架构会逐渐演化为下图：

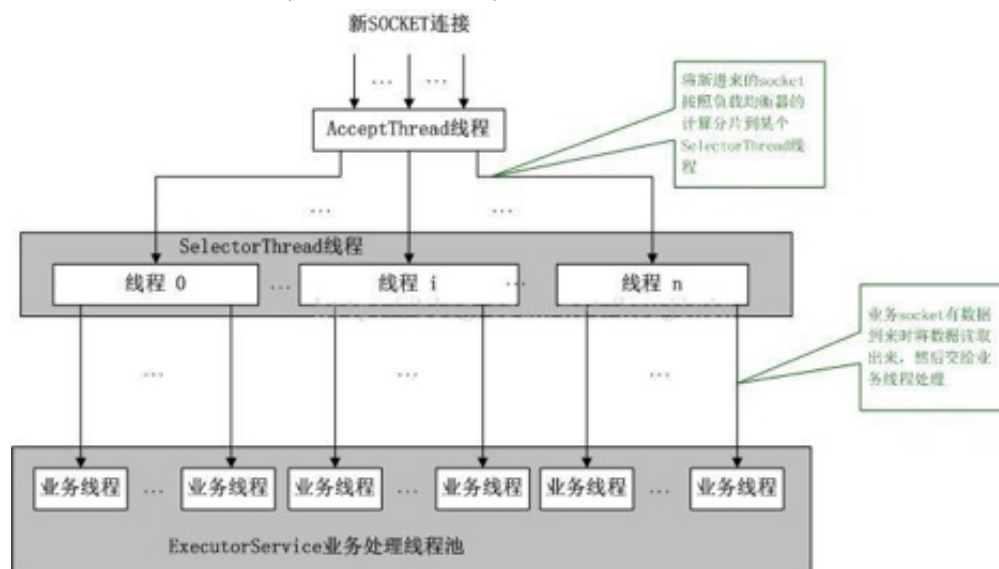


thrift的性能十分适合基础引擎的架构，与其他rpc协议的比较如下图：

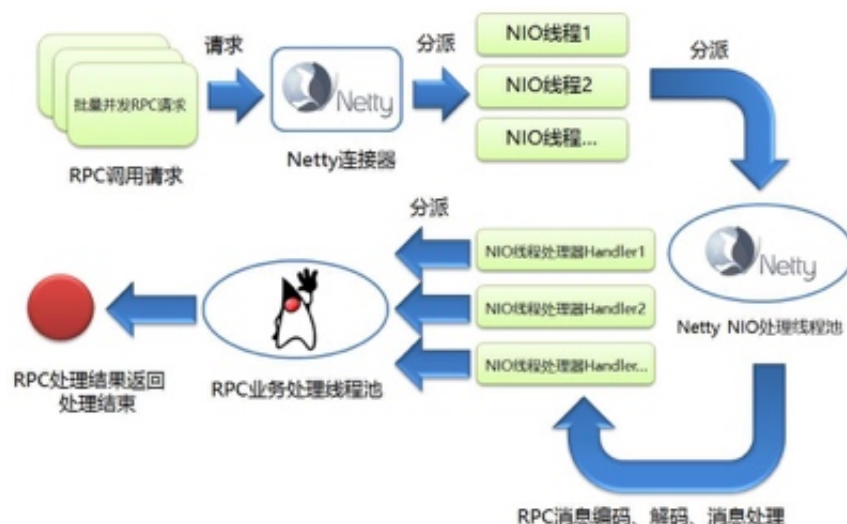


	Server CPU %	Avg Client CPU %	Avg Wall Time
REST — XML	12.00%	80.75%	05:27.45
REST — JSON	20.00%	75.00%	04:44.83
RMI	16.00%	46.50%	02:14.54
Protocol Buffers	30.00%	37.75%	01:19.48
Thrift — TBinaryProtocol	33.00%	21.00%	01:13.65
Thrift — TCompactProtocol	30.00%	22.50%	01:05.12

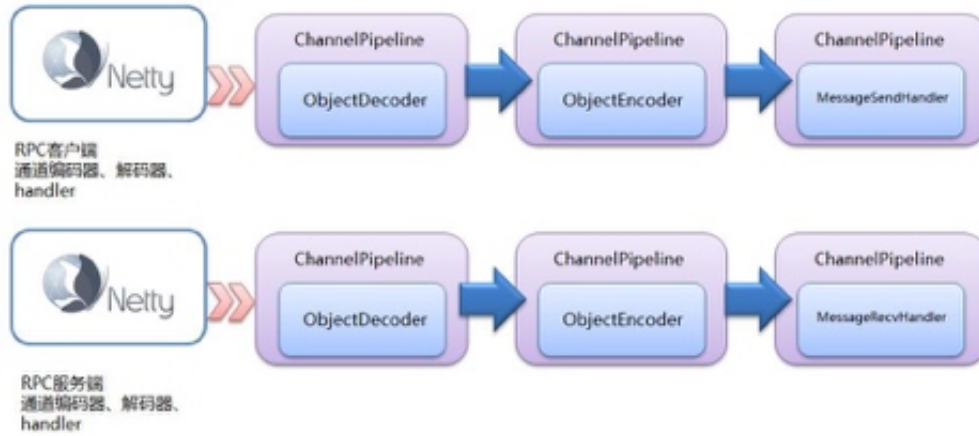
作为高可用的架构，在多核时代，多线程的thrift框架能把性能发挥得更好，工作池模式如下图：



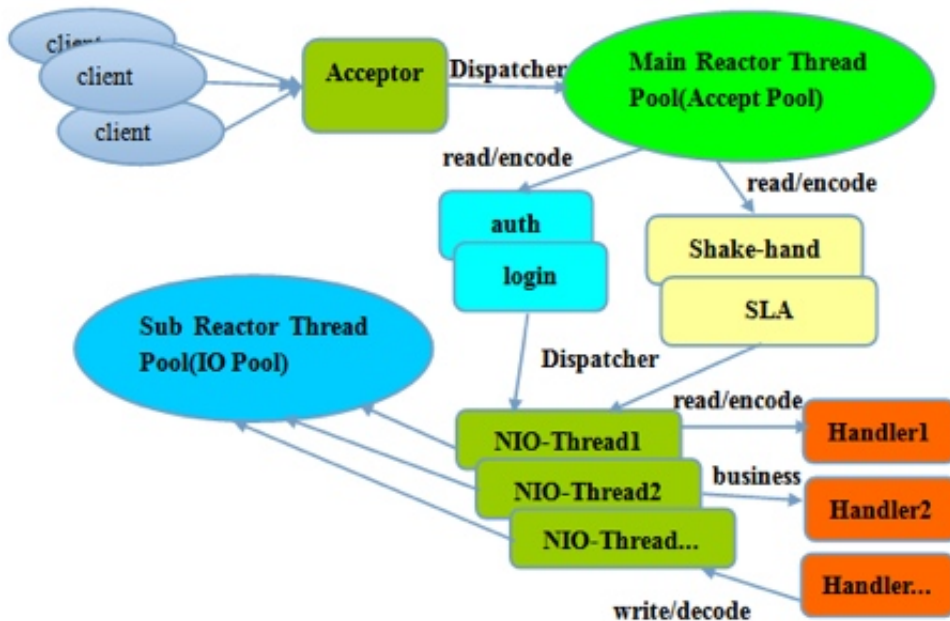
在诸多的开源类库中，netty的设计十分优良，也是本引擎选取的类库，netty的服务流程如下：



Netty对于RPC消息的解码、编码、处理对应的模块和流程，具体如下图所示：



Netty线程模型如下图所示：



5.2 Rpc远程调用接口设计

- UserThriftService
 - i. 用户登录接口
 - ii. 用户同意/拒绝 加入分组邀请
 - iii. 用户获取分组邀请列表
 - iv. 更新用户信息
- GroupThriftService
 - i. 获取用户分组接口
 - ii. 获取分组下账户
 - iii. 在指定分组下增加分组
 - iv. 删除指定分组

- v. 更新分组名
- vi. 指定分组下批量添加公众号
- vii. 给指定分组添加管理员
- viii. 获取指定分组下公众号
- ix. 获取区域管理员列表
- x. 获取用户树状分组

- StaticsThriftService

- i. 获取分组下的公众号排名（包括日榜，周榜，月榜）
 - ii. 获取分组下最新（最热）文章列表
 - iii. 获取区域排名
 - iv. 批量更新图片的url