老四哥项目平台框架设计

|  |  |  |
| --- | --- | --- |
| 作者 | 版本 | 时间 |
| 王剑锋 | 0.1 | 2017/12/28 |
|  |  |  |

老四哥项目包括很多业务功能模块，也包括很多流程，以及一个比较灵活的组织架构，综合看，老四哥项目相当复杂，需要一个比较灵活的平台设计才能很好的兼顾众多的功能模块的开发，流程的交互以及灵活的组织架构的设计。

基于以上认识以及比邻平台的经验，我对老四哥项目的实现基础平台进行了设计，大致包括以下几个组成部分。

# 组织架构的实现模型

组织架构的实现是项目首先的基础，老四哥项目的组织架构要求比较复杂，包括总部，分部，总部和分部内部还包括各个组织部门，如采购部，销售部等，同时总部和分部之间还有相互交叉的部分，总部可以对分部能够进行控制，分部也能够独立的实现一些自己的功能，为了满足组织架构的实现，设计了以下的基础模型。

在平台中，所有的组织单元都是一个org，总部是一个org, 分部也是一个org， 总部和分部的内部部门，也是各自的org。

每个org都是一个相对独立的单元，包括了独自的一套 权限(roles)，独自的一套功能（oapps，operations），也包括一套独自的业务流(workflows, works)

每个org除了拥有自己的功能，角色，业务流等，每个org还和其他org建立各种关系，比如总部org和部门org之间是机构与部门的关系（main-department）,总部和分部之间是总部与分部关系（head-branch）

机构之间的关系主要通过机构之间的关注实现的，比如总部的供销部门和总部建立关系，那么供销部门的 org需要关注总部main关系并把总部设置为main，同时总部需要反过来关注部门的department关系并把org设置为department（关系最终需要批准才能建立，比如供销部门把总部加为main，需要得到总部的批准）

机构之间的关注实质是关注者将被关注者的角色进入被关注者的角色系统中，比如供销部门关注总部，就把一些总部的角色，比如总部员工角色，引入部门中，然后部门的人员可以加入这些角色，从而执行一些总部赋予的功能（比如享受总部员工的待遇等）

反过来，机构总部以总部的身份关注供销部门，也就是将某些监管供销部门的角色引入总部，然后总部的人员可以通过这些角色的功能实现对供销部门的控制和监管。

采用同样的方法，可以建立起总部和分部之间的关系。

机构拓扑图

总部和部门

总部和部门

总部与分部

# 业务模块的实现模型

业务模块，也叫功能模块。老四哥项目的每个功能模块对应一个oapp(org app)，每个oapp包含独立定义的角色(roles)，操作(operations)和业务流(workflows, works)

oapp中其实定义的是模版，而不是真正的角色，操作以及业务流，当oapp加入到org时候，才会根据具体的org的环境，添加真正的角色，操作和业务流。

其中roles, operations, workflows(work)之间的相互关系是：

* 用户可以加入机构中的角色
* 角色会被赋予访问operation的权限，用户通过加入角色最终实现访问功能（operation）
* 每个operation可以是一个页面完成的操作，也可以是多个页面完成的操作，但一个operation必须是由一个角色或者说一个人独立完成的操作。
* 操作可以加入业务流(workflows)的一个工作单元中(work)，从而成为业务流中的一部分，实际上，业务流程本质上就是一系列操作完成的一项任务。

Org app 模版

# 业务流程的实现模型

workflows

operation

….

operation

关于业务流模型，有几个要点：

1. 每个业务流都有发起点和结束点，通常来说每个业务流都需要一个管理人(master)，一般来说，管理人需要对业务流的过程进行管理和控制
2. 业务流由业务中的work组成，每个work对应一个operation，work中会包含对operation进行监测的功能，对operation随时进行监听，一旦operation执行完成，work会通知执行下一个work，work同时也会负责通知提醒operation的操作人执行下一步操作。
3. 每个机构都可能有很多业务流程，这些业务流程主要由本机构的操作组成，跨机构的业务流程可以通过引入别的机构的角色和operation的方式实现。