**用户管理需求文档**

**2014-03-17 俞张平**

# 背景

目前部门使用orgv2系统来维护部门、用户信息的管理，除了管理功能以外系统还提供了Web service调用接口满足其它系统的方法。经过了这么多年的使用应该说orgv2还是比较稳定可靠的而且几乎可以满足所有关于用户管理的需求。不过作为系统的开发者之一也从中发现了一些问题。主要问题我列举以下几点:

1. 功能设计没有站在用户的角度出发

很多功能业务操作不知道该点如何操作，找不到操作入口，譬如换用户换部门，是到用户管理还是到部门管理操作，这个也可能是系统用户的定位问题，可能设计时就定位开发进行维护的关系。

1. 用户、部门的实体关系有缺陷

用户、部门以多对一的关系建立，其实这个关系在大部分系统中都是成立的，不过直到2012年部分系统要求需要有一个用户同时存在于多各部门的情况，只能对系统进行修改。其实类似的情况在以前的工作中也发深谷

1. 用户部门版本功能华而不实

原先设计用户部门版本的目的是为了记录用户业务操作时的所属那个部门。这样就可以不用在各个业务系统中保留用户历史信息减少冗余。不过实际情况是记录历史信息的要求比较少。而且缺点也比较多1加大程序的复杂度。2.从org中获取历史数据的效率会比较低。3.用户属性、部门属性都是经常需要做筛选条件的，所以保存用户、部门版本记录不便于筛选。所以这个功能目前还是比较鸡肋的。

1. 客户端对象不够简单

获取每个对象的属性的时候还需调用getValue()，太2了、初始化webserviceClient代码也过于复杂。

1. 使用Xfire作为Web service组件

首先这个不是个问题，Xfire是个很优秀的组件，功能强大、性能稳定而且便于开发。所以基本上已经成为java Webservice解决方案的首选。不过运行Xfire需要同时依赖几十个jar包，我个人非常不喜欢（纯属个人恩怨）。

# 目标：

重新开发一个org系统，界面功能借鉴目前的org，然后根据现在存在的问题进行改进，具体改进目标如下：

1. **优化数据结构**

1去掉用户部门版本概念简化数据结构。如果需要保留用户当前的部门信息、请各个业务系统独立保存。

2. 目前OA中遇到一个用户属于多个处室的情况是使用使用一些变通的办法如为一个用户设置多个账号等方法来实现的，这样操作肯定不是最好的方法。所以希望调整用户、部门的实体关系为多对多，从底层结构上支持用户多处室，这样才能彻底解决这个问题。

1. **以业务操作为导向优化界面设计**

1把用户作为系统的使用者，优化界面设计，提高系统可用性。就算最终用户还是把维护工作交给我们，以业务为导向设计的系统也方便做移交给维护部门。

1. **提高接口可用性**
2. 替换xfire，改用轻量级的实现譬如java内置的JAX-WS(JWS)
3. 对客户端接口进行再封装，避免业务代码中嵌入Webservice技术实现代码，方便用户调用、也为方便日后更换其它技术实现。

# 用例简述：

## 用户界面





## 接口相关

