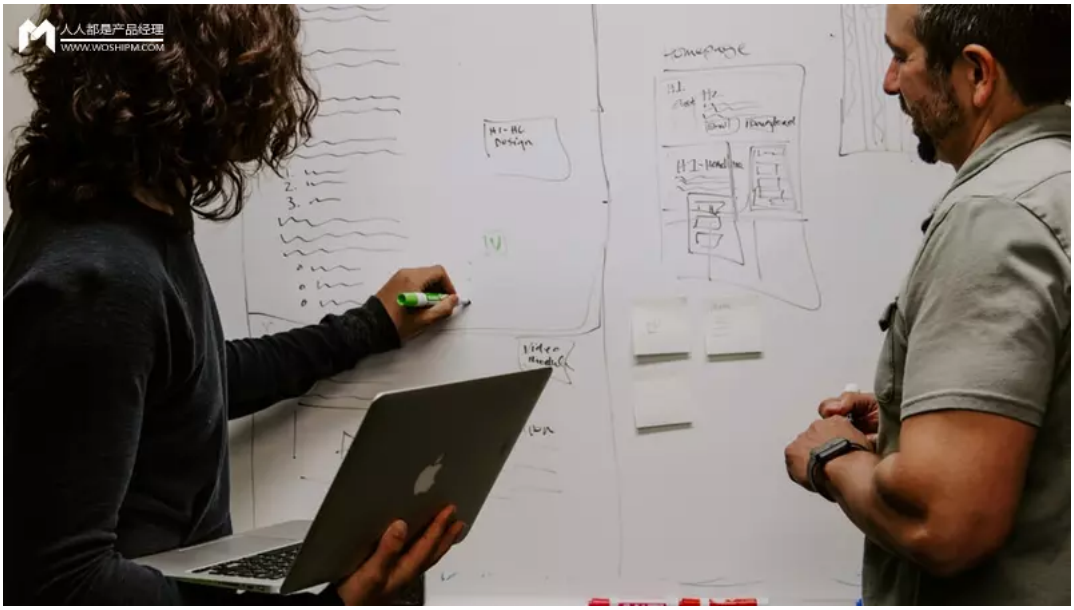
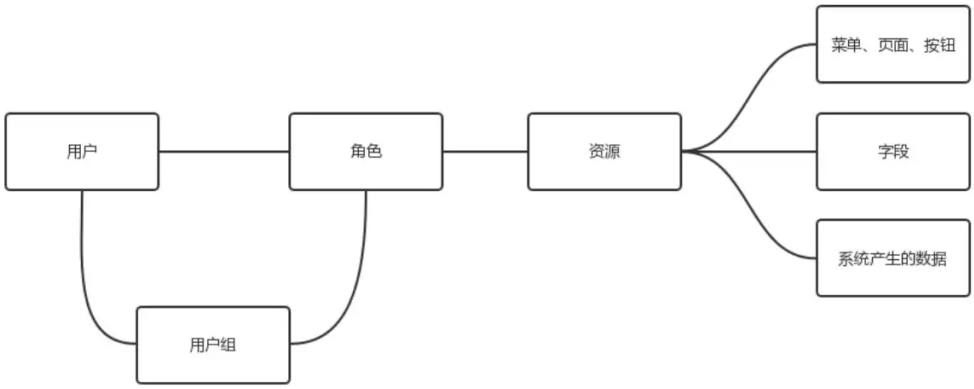


一、后台系统中权限管理设计的一般方法

在设计后台系统（如：CRM、EPR、EHR、电商管理后台等）时，权限管理是必不可少的功能，绝大部分的后台系统都是处理企业业务流程的，会涉及到多个部门的协同合作，必然需要对每个能够使用系统的用户进行权限管理。

在一般的单体应用的后台中权限管理的大体模式如下：



整体的业务逻辑如下：

1. 系统中的菜单、页面、按钮、字段以及运行时产生的数据都需要注册成为系统资源；
2. 系统资源打包组合成为角色；
3. 角色可以关联用户，也就完成了资源授权给用户的处理
4. 角色可以关联用户组，而用户组是多个用户组合而成的一个集合，用户能够继承用户组关联的角色

而在系统运行时，任何一个用户在使用系统资源时，都需要进行授权校验，也就是看这个用户关联的所有的角色囊括的资源是否包涵当前要访问的资源，如此就完成了用户权限管理的控制。

你没有看错，所有的单体应用的权限管理的实现逻辑都是如此。

但在基于业务中台的基础之上去做权限管理的设计我们需要额外引入更多的概念（租户、应用实例等）以完成业务逻辑。

二、基于业务中台的多租户权限设计需要解决的问题

所有中台建设的目的都是为了业务快速且低成本创新，绝大部分的企业基于中台都会开发大量的业务应用，一般基于业务中台的架构如下图：

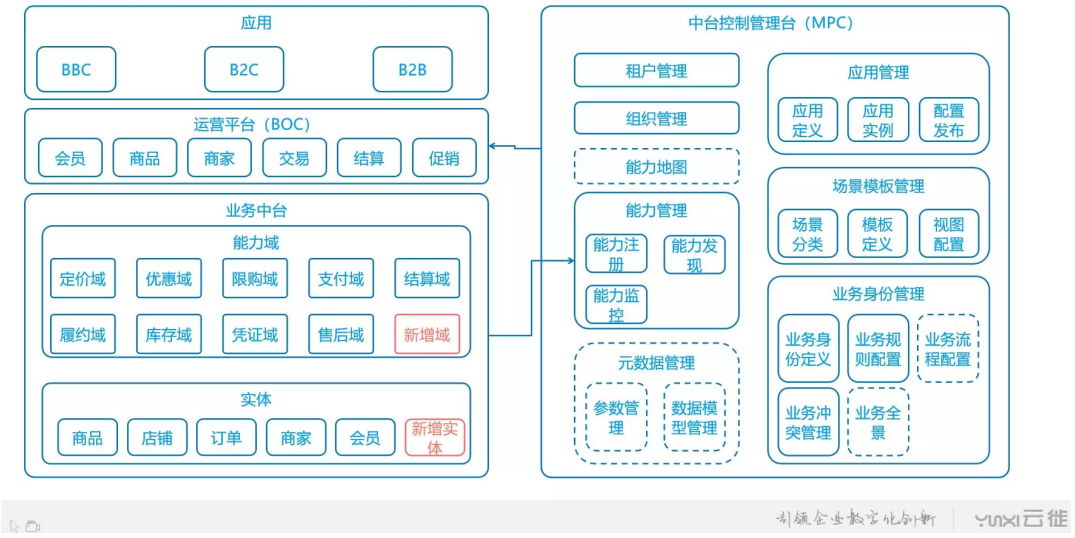
从图中可以看到，在中台之上有针对各个业务开展的各种应用，而笔者所在的企业是一家中台标准产品的厂商（即把中台作为基础设施的SaaS厂商），更是加入了多租户的机制以满足不同客户对应个性化的需求。

在基于中台的多租户、多应用的场景下，我们做权限管理的设计面临如下主要问题：

1. 在出厂时需要提供特殊的初始化权限管理流程；
2. 对于购买SaaS产品的客户而言，权限需要集中进行管理，以减少运营人员的工作内容；
3. 对于不同的角色/场景有不同的权限管理的需求。

三、具体的设计方案阐述

在解决以上问题之前我首先介绍下我们公司的整体产品架构：



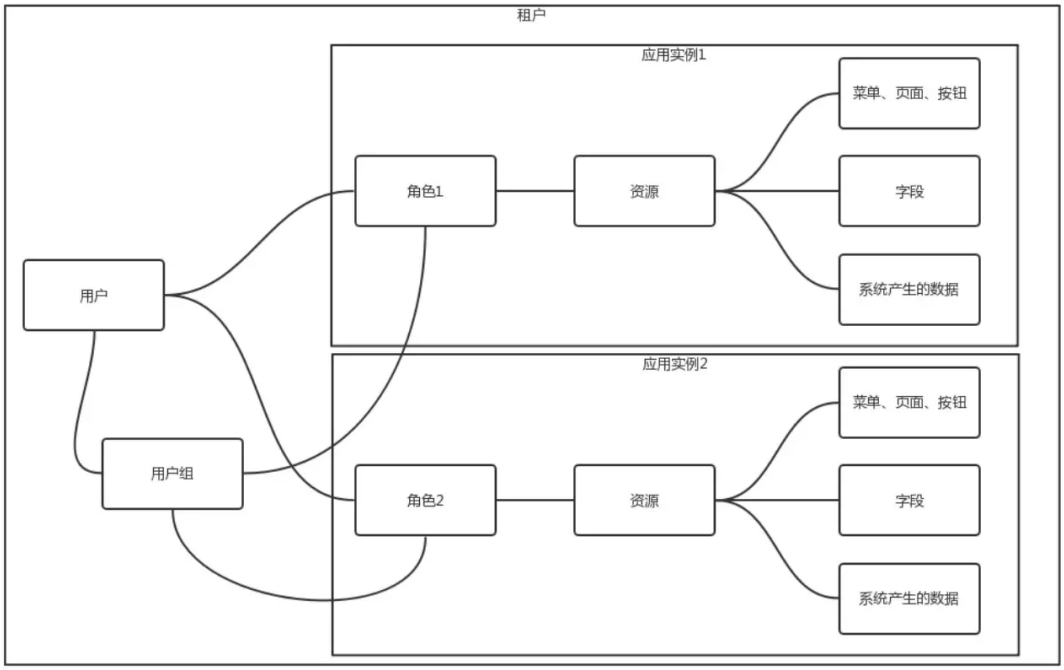
业务中台是我们所有应用的基础设施，我们能够通过MPC 配置各个应用所需要的业务能力，把业务能力组合起来就能形成一个应用，如此我们实现了业务中台的能力复用以及快速支撑业务创新。

在这个业务模式中，应用均是通过配置在进行一定的前端页面开发形成，我们可以为每个租户生产其所需要的应用实例，租户下的数据是隔离的。

在客户购买我们整个标准产品后（包括业务中台、MPC、BOC以及预置应用），首先我们在MPC中预置了一个root账户，通过该账户我能够创建租户，并为租户实例化应用，在实例化应用的同时，为该租户生成在该应用实例下的租户管理员。

租户管理员能够进入BOC进行全局的权限管理，例如：他能在该租户下创建用户，并设置该用户能够登录的应用；他能为租户下的任一应用实例创建角色，并把该角色分配租户下的用户。

租户管理员管理权限的模式如下：



整体业务逻辑：

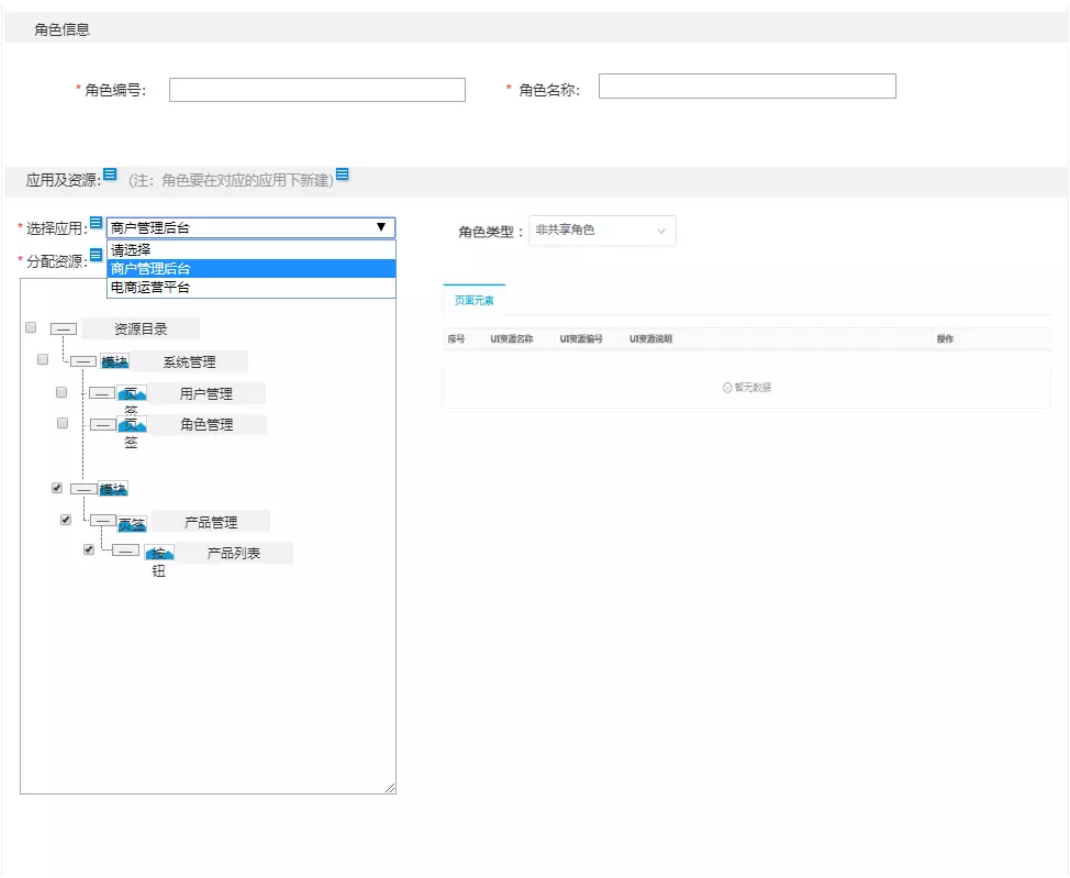
1. 系统初始化时，需要生成root账号，该账号由系统预置所有资源权限
2. root账号能够创建租户，并为租户实例化应用
3. 实例化应用的时候需要为租户生成租户管理员并赋予租户管理员该应用实例的管理员权限（管理员角色为应用预置）

以上是解决出厂初始化时的特殊的权限管理处理逻辑。

租户管理员能够在全局管理（BOC）中管理租户下的用户信息，并能够为用户关联应用及应用中的角色；

原型示意图

租户管理员能够在全局管理中管理每一个应用实例中角色；



具有应用实例权限的用户能够进入应用，并创建该应用实例下的用户、角色。

四、总结

以上就是我在基于业务中台多租户下权限管理设计的整体方案，租户是在SaaS模式下隔离数据使用，在数据层面有自己的独立空间；

应用实例指的是租户数据空间中运行的应用；用户是使用系统的直接对象，其能够使用资源是由其关联的角色决定；资源指的是系统中的菜单、页面、按钮、字段以及运行时产生的数据。

理清这些概念后即使是再复杂的系统我们进行权限管理设计也是不在话下。

对应上内容如有异议，欢迎大家随时与我探讨。

2年业务中台初级权限管理设计