# 数据权限

## 背景:

一般做一些项目中,每个客户会提一些权限的要求,比如某个人只能看到本部门内数据、或者控制某些人看到哪些数据（条件自定）。出现这种需求，一般做项目都会直接在项目代码中直接编程的方式修改，那么这就会造成一旦需求变更。代码就得改。需求继续变更，代码越来越难看。最终无法正常维护。也可能会影响到业务数据。所以我们需要做一个通用的权限控制模型来达到这个效果。

## 设计思路：

设计一个表里面保存条件子句，当用户声明过滤权限操作的时候，如果查询，则附加该条件。如果是更新或者添加、删除的。则利用该条件过滤判断的。

# 表结构

|  |  |  |
| --- | --- | --- |
| 字段 | 类型 | 描述 |
| Id | 主键 |  |
| name | 名称 |  |
| companyId | 单位id |  |
| modelClassName | Jpa实体类全名 | 业务表的MODEL模型CLass。包含  package的ClassName |
| relationId | 关系id | 关联用户ID、部门ID、群组ID、岗位  ID、角色ID |
| relationType | 关系类型 | 对应关系的类型包含：  user、group、department、post、  role |
| operationType | 过滤类型 | 过滤的类型：  READ 查询附加条件  CREATE 创建，无权限会抛异常  PRINGT 打印 查询附加条件  DELETE 删除 无权限件会抛异常  UPDATE 更新 无权限会抛异常 |
| conditionJpql | 自定义JPQL条件 | QL条件会附加到查询的where子句中 |

## 过滤逻辑

1. 查询
   1. 增加过滤权限声明testService.dataFilter(DataFilterType.READ)
   2. 后台执行查询的时候会根据 用户单位id、对应的jpa实体类、read类型 获取到当前用户有效的jpql 条件子句
   3. 然后类似下面的条件进行拼装：（子句1 or 子句2 or 子句3）并且将该条件 与 传入的条件 and 结合，形成类似如下的条件子句： （子句1 or 子句2 or 子句3）and id = 2
2. 增加
   1. 增加过滤权限声明 stService.dataFilter(DataFilterType.CREATE)
   2. 执行数据增加
   3. 在当前事务范围内，过滤数据权限，如果有权限则继续，否则回滚事务。
3. 删除
   1. 增加过滤权限声明 testService.dataFilter(DataFilterType.DELETE)
   2. 判断是否有删除当前数据的权限
   3. 如果有则继续执行删除，否则跑出没有权限的异常
4. 修改
   1. 增加过滤声明 testService.dataFilter(DataFilterType.UPDATE)
   2. 执行数据的更新
   3. 在当前事务范围内，判断是否有修改当前数据的权限，如果有则继续，否则跑出异常，回滚事务
5. 打印
   1. 增加过滤权限声明 testService.dataFilter(DataFilterType.PRINGT)
   2. 针对要过滤的数据范围附加打印类型的规则条件子句
   3. 返回的数据list 即为过滤过打印权限后的数据列表，该列表的长度肯定只会比未过滤之前的长度要小。

## 规则：

当用户service调用的时候，默认是没有过滤声明的。只有当用户显示的调用了dataFilter(DataFilterType) 的时候，才会附加权限过滤规则。

注意： 权限过滤只支持baseDao中未添加@Deprecated 注解的方法。并且当用户调用过权限过滤后，权限过滤的声明会自动移除。如果用户需要在同一个方法两次调用权限过滤，类似如下：

testService.dataFilter(DataFilterType.READ).findOne(2);

testService.dataFilter(DataFilterType.PRINGT).findAll();

需要在要过滤的方法前，显示调用下dataFilter方法。

## 变量

用户可以在jpql条件子句中可以直接使用的用户变量有:

|  |  |
| --- | --- |
| 登录用户id | userId |
| 用户的单位ID | userOrgId |
| 当前单位和上级单位的id集合 | userParentDeptIds |
| 用户所属部门的id集合 | userDeptIds |
| 用户所属群组id集合 | userGroupIds |
| 用户默认部门id | userDefaultDeptId |
| 用户所属部门和子部门的id集合 | userChildDeptIds |
| 用户所属单位和子单位的id集合 | userChildOrgIds |
| 当前时间 | dateNow |
| 当前年 | dateYear |

用法: conditionJpql 可以为如下:

1. 控制用户只能查看本部门及子部门数据 operationType = READ
   1. deptId = : userChildDeptIds
2. 控制用户只能填加本年度的数据 operationType = CREATE
   1. Nd = : dateYear
3. 控制用户只能查看个人数据 operationType = READ
   1. createUser = :userId
4. 控制用户只能查看本单位数据 operationType = READ
   1. companyId = :userOrgId