**类目中心设计**

# 定位

类目中心负责管理电商平台里的行业、类目、门户导航、商品类目对应属性的管理等，是电商平台最为核心的模块之一。

# 名称解释

* 类目：即商品的所属分类，如手机、家电等
* 叶子节点：指类目树下最底层的类目节点，下面没有子类目了，所有商品必须挂靠在叶子节点上。
* 前台类目：电商门户展示的导航菜单
* 后台类目：卖家维护商品时标准化挂靠的类目，与前台类目有映射关系。

# 详细设计



类目关系图

参照淘宝、京东等大型网站等类目设计，Jbone电商平台等类目体系采用前后端类目分离，中间采用映射关系进行关联。

商品属性采用属性-属性值的设计方式。

## 前后端类目分离设计

### 前端类目

 用户购买时看到的类目是前台类目。

 前台类目通过跟后台映射间接和商品关联。

  **前台类目由后台叶子类目+属性（前台类目只和叶子类目）组成，一个前台类目可以包含多个后台类目，一个后台类目也可以映射给多个前台类目。**

 前台类目不直接挂载属性模版，前台类目对应的属性来自后台叶子类目的公共属性。

 前台类目很灵活，可重叠，可删除。

### 后端类目

后台类目面向商家，主要用于商品的标准化分类和属性管理。商家上传商品时见到的就是后台类目。

* **后台类目树中最重要的是叶子类目，也就是类目树上不能再往下分的类目,任何商品都必须挂载到后台叶子类目上。**
* 叶子类目挂载属性模版，商家发布商品时选择好类目之后会根据属性模版，补充必填的商品属性信息，方可成功上传商品。
* 后台类目相对稳定，不能随便删除，叶子类目不能重复。

### 为什么进行前后端类目分离

这里贴出个网上的淘宝设计的例子

淘宝商品日益增多，类目层级也越来越深，买家越来越难找到想要的商品。淘宝的小二们就开始不断调整类目属性，把类目树变浅，让商品更容易被买家找到。只有一套类目体系的情况下，小二每次调整，卖家也必须跟着调整。如果只是改一次，卖家也能勉强接受，但这是周期性的。比如，在夏天服装这个类目下可能是连衣裙、衬衫等夏装，到了冬天就会变成打底裤、羽绒服等冬装了，而且卖家也都知道这些调整时合理的，但是一年到头这么调整，意味着他一年到头都在编辑类目，可能还得专门养个把人在那里，一天到晚改类目，无形中卖家的运营成本也就提高了。

08年的时候，有个淘宝的产品经理从线下零售巨头沃尔玛得到启发，他发现沃尔玛仓库里面的类目分区和货架区的类目分区是分离的，仓库里面的类目分区比较稳定，很少变化，而货架区的类目分区会根据活动和季节经常调整。类似的，淘宝也可以把原来的一套类目体系分为后台分类和前台分类，后台分类面向商家，用来挂载商品和属性模板，比较稳定很少变化；前台分类面向用户，主要方便用户查找商品，很灵活，可以经常调整。后台类目和前台类目之间通过映射联系起来，一个后台类目可以映射到多个前台类目，一个前台类目也可以包含多个后台类目。

从技术的角度来看的话，前台类目就是在后台类目的基础上建立了一个虚拟类目。

# 数据库设计

## 行业表

行业实际上是一级类目的集合，比方说健身行业，后面可能会有健身器材、运动服等一级类目。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **行业表（industry）** | | | | |
| **字段** | **类型** | **注释** | **非空** | **索引** |
| Id | int | 自增主键，行业ID | 是 | 主键索引 |
| name | varchar(200) | 行业名字 | 是 |  |
| business\_type | int | 业务类型；  1：普通 | 是，默认1 |  |
| description | varchar(500) | 行业描述 | 否 |  |
| sort\_num | int | 排序号 | 是，默认0 |  |
| status | int | 状态，1:可用，0: 不可用 | 是，默认1 |  |
| features | varchar(4096) | 预留行业扩展字段 |  |  |
| operator | varchar(100) | 最后更新者，当前登录名 | 是 |  |
| add\_time | timestamp | 添加时间 | 是 |  |
| update\_time | timestamp | 更新时间 | 是 |  |
| version | int | 版本号 | 是 |  |

## 后台类目表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **后台类目表（back\_category）** | | | | |
| **字段** | **类型** | **注释** | **非空** | **索引** |
| id | int | 自增主键，行业ID | 是 | 主键索引 |
| name | varchar(200) | 类目名字 | 是 |  |
| pid | int | 父级类目 | 是，默认0为一级类目 |  |
| cids | varchar(4096) | 子类目ID，逗号隔开，方便查询该类型下所有子类目包含的商品 |  |  |
| description | varchar(500) | 类目描述 | 否 |  |
| sort\_num | int | 排序号 | 是，默认0 |  |
| status | int | 状态，1:可用，0: 不可用 | 是，默认1 |  |
| features | varchar(4096) | 预留类目扩展字段（JSON格式） |  |  |
| creator | varchar(100) | 创建者，当前登录名 | 是 |  |
| add\_time | timestamp | 添加时间 | 是 |  |
| update\_time | timestamp | 更新时间 | 是 |  |
| version | int | 版本号 | 是 |  |

## 行业-类目关系表

行业只和一级类目关联

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **行业类目关系表（industry\_back\_category）** | | | | |
| **字段** | **类型** | **注释** | **非空** | **索引** |
| id | int | 自增主键，行业ID | 是 | 主键索引 |
| industry\_id | int | 行业ID | 是 |  |
| back\_category\_id | int | 一级类目ID，行业只跟一级类目产生关联，子类目同属于本行业 | 是 |  |
| add\_time | timestamp | 添加时间 | 是 |  |
| update\_time | timestamp | 更新时间 | 是 |  |
| version | int | 版本号 | 是 |  |

## 前台类目表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **前台类目表（front\_category）** | | | | |
| **字段** | **类型** | **注释** | **非空** | **索引** |
| id | int | 自增主键，行业ID | 是 | 主键索引 |
| name | varchar(200) | 类目名字 | 是 |  |
| pid | int | 父级类目 | 是，默认0为一级类目 |  |
| cids | varchar(4096) | 子类目ID，逗号隔开，方便查询该类型下所有子类目包含的商品 |  |  |
| description | varchar(500) | 类目描述 | 否 |  |
| sort\_num | int | 排序号 | 是，默认0 |  |
| status | int | 状态，1:可用，0: 不可用 | 是，默认1 |  |
| img\_url | varchar(200) | 类目导航图 |  |  |
| min\_price | Int | 价格区间最小值 | 默认-1 |  |
| max\_price | int | 价格区间最大值 | 默认-1 |  |
| operator | varchar(100) | 最后更新者，当前登录名 | 是 |  |
| add\_time | timestamp | 添加时间 | 是 |  |
| update\_time | timestamp | 更新时间 | 是 |  |
| version | int | 版本号 | 是 |  |

## 前后台类目关系表

用于在前台导航映射到后台类目

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **类目表（front\_back\_category\_relation）** | | | | |
| **字段** | **类型** | **注释** | **非空** | **索引** |
| id | int | 自增主键，行业ID | 是 | 主键索引 |
| front\_category\_id | int | 前台类目ID | 是 |  |
| back\_category\_id | int | 后台类目ID | 是 |  |
| operator | varchar(100) | 最后更新者，当前登录名 | 是 |  |
| add\_time | timestamp | 添加时间 | 是 |  |
| update\_time | timestamp | 更新时间 | 是 |  |
| version | int | 版本号 | 是 |  |

## 属性表

比如颜色、国家、语言、品牌等

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **类目表（public\_property）** | | | | |
| **字段** | **类型** | **注释** | **非空** | **索引** |
| id | int | 自增主键，行业ID | 是 | 主键索引 |
| name | Varchar(100) | 属性名字 | 是 |  |
| status | int | 状态：  1：可用  0:不可用 | 是 |  |
| alias\_name | varchar(100) | 属性别名 |  |  |
| description | varchar(500) | 属性描述/备注 |  |  |
| operator | varchar(100) | 最后更新者，当前登录名 | 是 |  |
| add\_time | timestamp | 添加时间 | 是 |  |
| update\_time | timestamp | 更新时间 | 是 |  |
| version | int | 版本号 | 是 |  |

## 属性值表

确定的值，如白色、中国、苹果、等等

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **属性值表（public\_value）** | | | | |
| **字段** | **类型** | **注释** | **非空** | **索引** |
| id | int | 自增主键，行业ID | 是 | 主键索引 |
| content | varchar(1024) | 属性值 | 是 |  |
| status | int | 状态：  1：可用  0:不可用 | 是 |  |
| description | varchar(500) | 描述/备注 |  |  |
| code\_upper | varchar(500) | 属性值大写编码，用于去重等 |  |  |
| operator | varchar(100) | 最后更新者，当前登录名 | 是 |  |
| add\_time | timestamp | 添加时间 | 是 |  |
| update\_time | timestamp | 更新时间 | 是 |  |
| version | int | 版本号 | 是 |  |

## 属性-值关系表

维护属性有哪些值，比如颜色有白色、黑色、等等

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **属性-值关系表（public\_property\_value）** | | | | |
| **字段** | **类型** | **注释** | **非空** | **索引** |
| id | int | 自增主键，行业ID | 是 | 主键索引 |
| property\_id | int | 属性ID | 是 |  |
| value\_id | int | 属性值ID | 是 |  |
| description | varchar(500) | 描述/备注 |  |  |
| status | int | 状态：  1：可用  0:不可用 | 是 |  |
| operator | varchar(100) | 最后更新者，当前登录名 | 是 |  |
| add\_time | timestamp | 添加时间 | 是 |  |
| update\_time | timestamp | 更新时间 | 是 |  |
| version | int | 版本号 | 是 |  |

## 后端类目-属性关系表

此表用来存储类目下有哪些属性，比如颜色、型号等；注意只有叶子节点才拥有属性，和属性进行关联

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **后端类目-属性关系表（public\_property\_value）** | | | | |
| **字段** | **类型** | **注释** | **非空** | **索引** |
| id | int | 自增主键，行业ID | 是 | 主键索引 |
| back\_category\_id | int | 后端类目ID（叶子节点） | 是 |  |
| property\_id | int | 属性ID | 是 |  |
| Is\_must | int | 是否必填  1:必填  0:非必填 |  |  |
| Is\_multi | int | 是否多值  1:多值  0:非多值 |  |  |
| Is\_input\_prop | Int | 是否可输入  0:不可（默认）  1:可以 |  |  |
| description | varchar(500) | 描述/备注 |  |  |
| status | int | 状态：  1：可用  0:不可用 | 是 |  |
| operator | varchar(100) | 最后更新者，当前登录名 | 是 |  |
| add\_time | timestamp | 添加时间 | 是 |  |
| update\_time | timestamp | 更新时间 | 是 |  |
| version | int | 版本号 | 是 |  |

## 后端类目-属性-值关系表

此表用来存储类目下属性有哪些值，比如iphone有颜色为白色的。也就是有白色的Iphone手机

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **后端类目-属性-值关系表（back\_category\_property\_value）** | | | | |
| **字段** | **类型** | **注释** | **非空** | **索引** |
| id | int | 自增主键，行业ID | 是 | 主键索引 |
| back\_category\_id | int | 后端类目ID（叶子节点） | 是 |  |
| property\_id | int | 属性ID | 是 |  |
| Is\_must | int | 是否必填  1:必填  0:非必填 |  |  |
| Is\_multi | int | 是否多值  1:多值  0:非多值 |  |  |
| Is\_input\_prop | Int | 是否可输入  0:不可（默认）  1:可以 |  |  |
| description | varchar(500) | 描述/备注 |  |  |
| status | int | 状态：  1：可用  0:不可用 | 是 |  |
| operator | varchar(100) | 最后更新者，当前登录名 | 是 |  |
| add\_time | timestamp | 添加时间 | 是 |  |
| update\_time | timestamp | 更新时间 | 是 |  |
| version | int | 版本号 | 是 |  |

商品基本信息（不在类目中心中，这里是为了梳理思路）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **商品基本信息表（item\_info）** | | | | |
| **字段** | **类型** | **注释** | **非空** | **索引** |
| id | bigint | 自增主键，行业ID | 是 | 主键索引 |
| back\_category\_id | int | 后端类目ID（叶子节点） | 是 |  |
| item\_id | bigint | 商品ID，使用生成器生成 | 是 |  |
| shop\_id | bigint | 店铺ID | 是 |  |
| name | varchar(200) | 商品名字 | 是 |  |
| price | bigint | 商品价格,单位为分 | 是 |  |
| stock | int | 库存量 |  |  |
| sold | int | 销量 |  |  |
| img\_head | varchar(200) | 商品主图 | 是 |  |
| description | varchar(500) | 描述/备注 |  |  |
| status | int | 状态：  1：可用  0:不可用 | 是 |  |
| operator | varchar(100) | 最后更新者，当前登录名 | 是 |  |
| add\_time | timestamp | 添加时间 | 是 |  |
| update\_time | timestamp | 更新时间 | 是 |  |
| version | int | 版本号 | 是 |  |

商品SKU

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **商品SKU信息表（item\_sku）** | | | | |
| **字段** | **类型** | **注释** | **非空** | **索引** |
| Id | bigint | 自增主键，行业ID | 是 | 主键索引 |
| shop\_id | bigint | 后端类目ID（叶子节点） | 是 |  |
| item\_id | bigint | 商品ID，使用生成器生成 | 是 |  |
| Item\_sku\_id | bigint | 商品SKU ID，生成器生成 | 是 |  |
| title | varchar(200) | 型号标题 | 是 |  |
| img | varchar(200) | SKU图片 | 是 |  |
| status | int | 状态：  1：可用  0:不可用 | 是 |  |
| operator | varchar(100) | 最后更新者，当前登录名 | 是 |  |
| add\_time | timestamp | 添加时间 | 是 |  |
| update\_time | timestamp | 更新时间 | 是 |  |
| version | int | 版本号 | 是 |  |

商品SKU属性表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **商品SKU属性表（item\_sku\_attr）** | | | | |
| **字段** | **类型** | **注释** | **非空** | **索引** |
| Id | bigint | 自增主键，行业ID | 是 | 主键索引 |
| Item\_sku\_id | bigint | 商品SKU ID | 是 |  |
| property\_id | int | 属性ID | 是 |  |
| value\_id | bigint | 属性值ID | 是 |  |
| value\_type | int | 属性值类型  1: 系统提供  2: 卖家自定义值 | 是 |  |
| operator | varchar(100) | 最后更新者，当前登录名 | 是 |  |
| add\_time | timestamp | 添加时间 | 是 |  |
| update\_time | timestamp | 更新时间 | 是 |  |
| version | int | 版本号 | 是 |  |

商品SKU库存表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **商品SKU属性表（item\_sku\_stock）** | | | | |
| **字段** | **类型** | **注释** | **非空** | **索引** |
| Id | bigint | 自增主键，行业ID | 是 | 主键索引 |
| Item\_sku\_id | bigint | 商品SKU ID，如果商品没有SKU，则为0，并将价格、库存、销量同步过来 | 是 |  |
| Item\_stock\_id | bigint | 商品库存ID | 是 |  |
| shop\_id | bigint | 店铺ID | 是 |  |
| price | int | 价格，单位为分 | 是 |  |
| stock | int | 库存量 | 是 |  |
| sold | int | 销量 |  |  |
| status | int | 状态：  1：可用  0:不可用 | 是 |  |
| add\_time | timestamp | 添加时间 | 是 |  |
| update\_time | timestamp | 更新时间 | 是 |  |
| version | int | 版本号 | 是 |  |

商品SKU库存记录表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **商品SKU属性表（item\_sku\_stock\_status）** | | | | |
| **字段** | **类型** | **注释** | **非空** | **索引** |
| Id | bigint | 自增主键，行业ID | 是 | 主键索引 |
| order\_id | bigint | 订单ID | 是 |  |
| item\_stock\_id | bigint | 商品库存ID | 是 |  |
| stock\_status | int | 库存状态  0:初始值  1:已预知  2:已扣减  3:已回滚 | 是 |  |
| sold\_status | int | 销量状态  0:初始值  1:已加量  2:已回退 | 是 |  |
| num | int | 增量库存数，用于预扣、扣减、回滚读取 | 是 |  |
| add\_time | timestamp | 添加时间 | 是 |  |
| update\_time | timestamp | 更新时间 | 是 |  |
| version | int | 版本号 | 是 |  |

# 接口设计

## 新增类目

### 入参

类目名称

父级类目ID

类目描述

类目状态

类目扩展属性（JSON）

创建者

### 出参

成功/失败状态（参考API中Result标准）

### 处理逻辑注意事项

注意创建子类目是，更新父级的cids，一级类目不需要更新

## 更新类目

### 入参

类目ID

类目名称

父级类目ID

类目描述

类目状态

类目扩展属性（JSON）

### 出参

成功/失败状态（参考API中Result标准）

### 处理逻辑注意事项

注意创建子类目是，更新父级的cids，一级类目不需要更新

## 修改类目状态

### 入参

类目ID

类目新状态

### 出参

成功/失败状态（参考API中Result标准）

### 处理逻辑注意事项

暂无

## 分配行业

### 入参

类目ID

行业ID集合

### 出参

成功/失败状态（参考API中Result标准）

### 处理逻辑注意事项

参考用户-权限管理处理方式，这里不需要写逻辑，利用JPA的多对多自动处理就好。

## 删除类目

### 入参

类目ID

### 出参

成功/失败状态（参考API中Result标准）

### 处理逻辑注意事项

包含子类目的提示失败，不能删除

## 批量获取类目详情

### 入参

类目ID集合

### 出参

成功/失败状态（参考API中Result标准）

类目详情集合（排序）

### 处理逻辑注意事项

根据sort\_num进行排序

只展示可用状态的类目

## 获取类目树

### 入参

父级类目ID（0为根类目，代表从根类目开始）

是否及联包含子类目（true：递归包含子类目，false：只查询当前PID的类目集合）

### 出参

成功/失败状态（参考API中Result标准）

类目树

### 处理逻辑注意事项

根据传入的父级类目ID，递归往下查询，形成类目树

父级类目传0，表示获取所有

只展示可用状态的类目

## 获取类目行业

### 入参

类目ID

### 出参

成功/失败状态（参考API中Result标准）

行业详情集合

### 处理逻辑注意事项

可利用JPA多对多关系，查询类目时就可以把行业详情带出来。不用写很复杂的逻辑。但要注意状态的过滤

## 行业接口

### 接口列表

1. 新增
2. 修改
3. 删除
4. 改变行业状态
5. 获取所有行业列表
6. 根据ID集合获取行业列表

### 注意事项

注意删除时及联删除行业-分类关系表，这里可利用JPA特性自动删除

注意获取行业时候的状态过滤