

芋道源码 —— 知识星球

我是一段不羁的公告!

记得给艿艿这 3 个项目加油,添加一个 STAR 噢。

https://github.com/YunaiV/SpringBoot-Labs

https://github.com/YunaiV/onemall

https://github.com/YunaiV/ruoyi-vue-pro

<u>2018-01-18</u> 数据库实体设计

数据库实体设计 —— 商品(1.1)之商品信息

艿艿目前正在做一个开源的电商项目,胖友可以 star 下。https://gitee.com/zhijiantianya/onemall

1. 概述

本文主要分享商品模块的商品信息的数据库实体设计。

基于如下信息,逆向猜测数据库实体:

有赞云提供的商品API 有赞微商城的商品管理 有赞云开发指南的商品场景

【护脸旁白】

笔者非电商行业出身 && 非有赞工程师,所以有错误或不合理的地方,烦请斧正和探讨。 有赞是个各方面都很 NICE 的公司,推荐。

2. 背景了解

在看具体的数据库实体设计之前,我们先一起了解下电商的名词定义和有赞微商城界面。

2.1 名词定义

参考 《产品 SKU 是什么意思?与之相关的还有哪些?》 整理。

SKU: Stock Keeping Unit

中文翻译为库存单位。SKU 从库存视角,以库存进出为单位,可以是件、瓶、箱等等。

例如, iPhone 手机, 按照规格(颜色 + 内存)可以组合出如下多个 SKU:

SKU 颜色 内存

A 白色 16G

SKU 颜色 内存

- B 白色 64G
- C 黑色 16G
- D 黑色 64G

可以看出,颜色(白色、黑色)与内存(16G、64G)组合排列出四种 iPhone SKU。

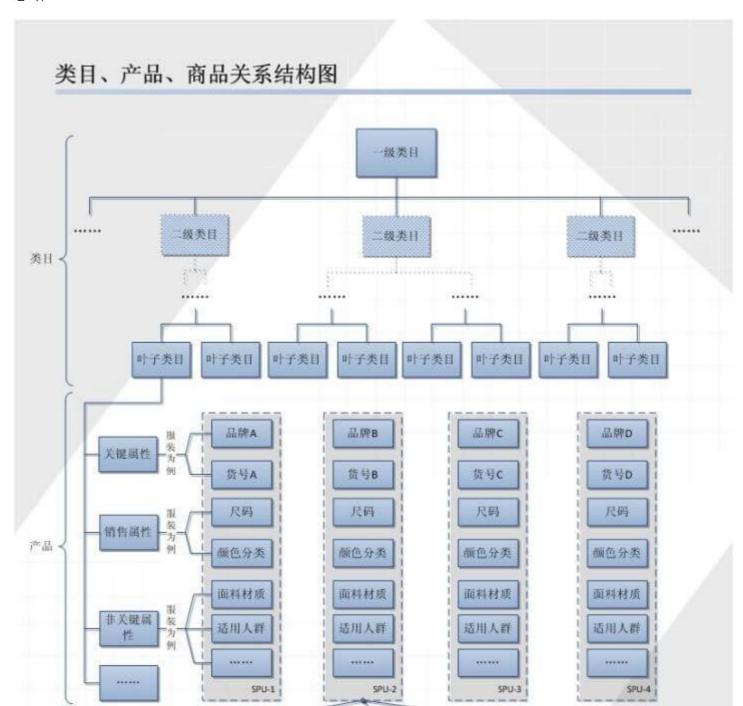
SPU: Standard Product Unit

中文翻译为标准产品单位。SPU 从产品视角,是产品信息聚合的最小单位,是一组可复用、易检索的标准化信息的集合,该集合描述了一个产品的特性。通俗点讲,属性值、特性相同的商品就可以被称为一个 SPU 。例如 iPhone 8 就是一个 SPU ,iPhone 8 Plus 也是一个 SPU ,这个与商家无关,与颜色、款式、套餐等规格无关。

商品:

商家出售某个 SPU ,那么这就是一个商品。商品在 SPU 之上,增加了销售价格、促销活动、运费等等信息。另外,一个商品可以包含多个 SKU。

总结



现实的场景往往比定义复杂的多,在本文中,SKU 代表销售的单元。主要考虑如下两方面:

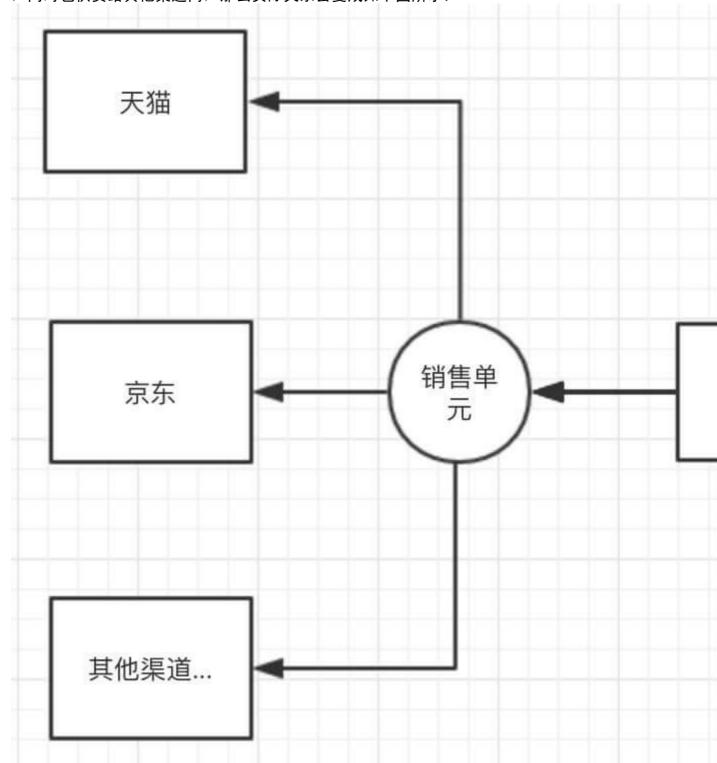
实际我们看到的商品详情页,购买的是一个销售组合单元。例如,很多商家会打包 【iPhone X:银色-64G-套餐三】,其中套餐三为赠送贴膜 + 保护壳等等,当然价格上会更贵。这明显就违背了我们上述提到 SKU 库存的概念,已经变成了多个 SKU 的销售组合单元。

Apple/苹果 iPhone X 苹果X 5.8英寸手机iphon 日版

日版是无锁支持3网4G; 【官方标配 = 全新激活 单机无配件, 套套配件】【原装充电头+数据线=80元 全套原装充电头+数据线+耳价】【支持分期购 0首付 顺丰包邮】

¥6680.00-8760.00 价格 配送 广东深圳 至 北京海淀区 ▼ 快递 免运费 ▼ 24小 机身颜色 银色 灰色 套餐类型 官方标配 套餐 套餐= 存储容量 256GB 64GB 版本类型 港澳台 韩国 美国 日本 中国大陆 售后服务 手机全国联保/延保1年 ¥187.04 (可选) 手机意外保1年 ¥382.09

一个商家会在不同平台销售商品,例如三只松鼠,其在天猫、京东等等平台都有官方旗舰店 ,同时也供货给其他渠道商,那么实际关系会变成如下图所示:



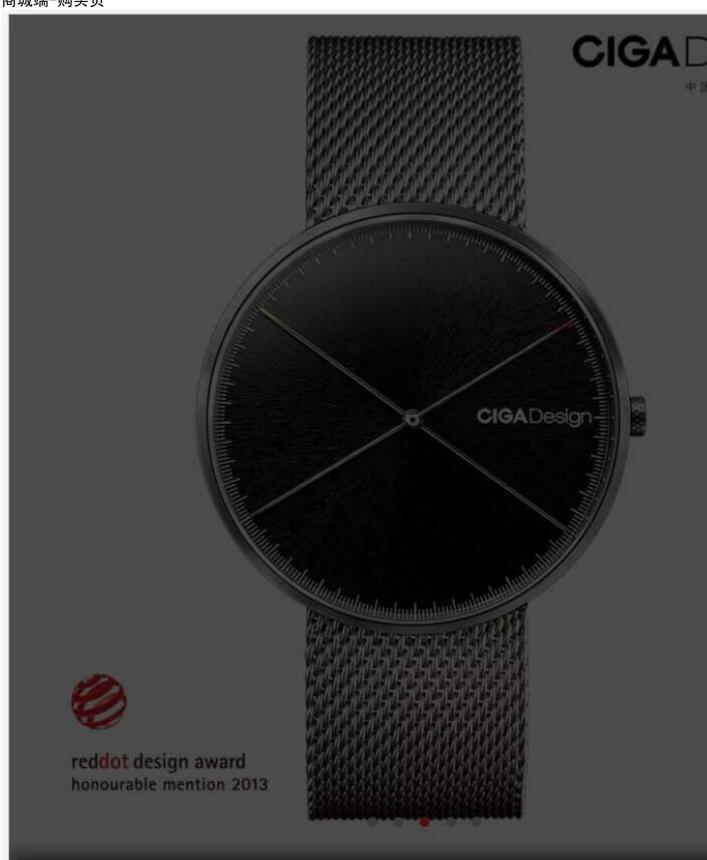
通过这样的方式,三只松鼠在不同的平台,定义不同的价格,设置不同的促销信息等等个性化的运营。

那么注意了!!!

下文开始,SKU 代表销售的单元 下文开始,SKU 代表销售的单元 下文开始,SKU 代表销售的单元

2.2 界面

1. 商城端-购买页





【德国红点设计,只属于你的时间】CIGA D.

¥650.00 - 1299.00

2. 运营后台-商品发布页

商品管理

商品管理

商品分组

商品页模版

商品导入

我要分销

商品名:

iPhone 8

分享描述:

微信分享给好友时会显

商品图:

+添加图片

建议尺寸: 800*800像

▶ 更多设置

价格库存

商品规格:

规格名:

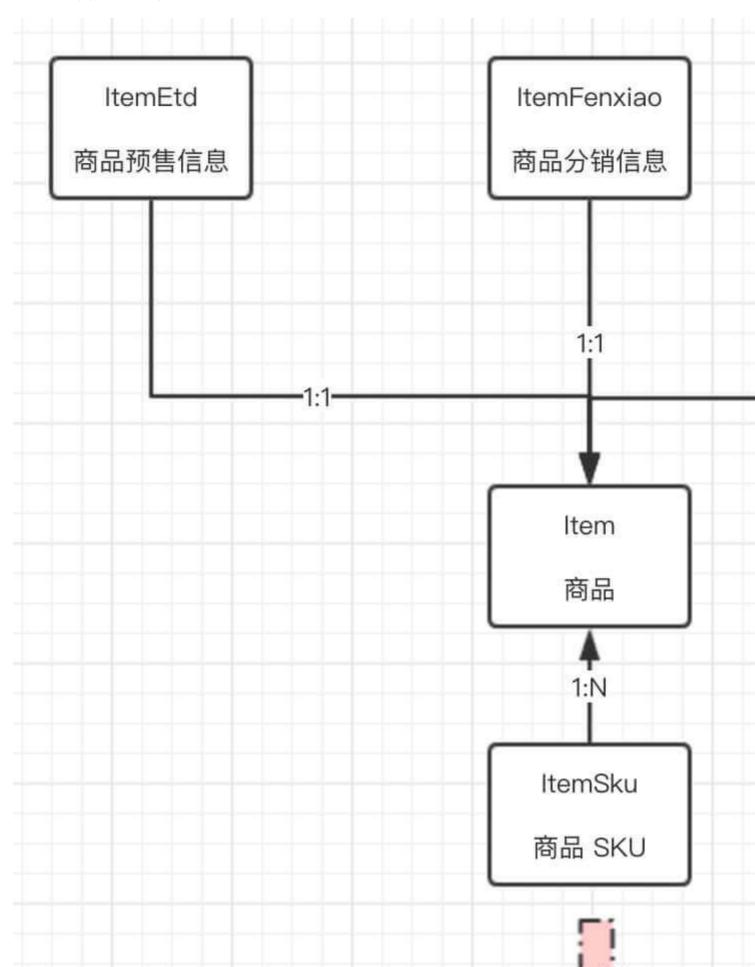
颜色

规格值:

红色

3. 数据库实体

整体实体类关系如下图:



全部实体在 Github 商品实体目录 下可见。

3.1 Item

Item 字段较多,我们进行简单的切块。

3.1.1 基础字段

```
/**
* 编号
private Integer id;
* 别名
* 系统生成,作为唯一标识。例如,2fpa62tbms19h
private String alias;
/**
* 店铺编号
private Integer shopld;
/**
* 创建时间
private Date createTime;
/**
* 更新时间
private Date updateTime;
/**
* 状态
* 1-正常
* 2-删除
*/
private Integer status;
```

id , Item 编号,自增。数据类型是使用 Integer 还是 Long , 胖友可以根据自己的系统需要做调整。

alias ,别名,系统自动生成,作为唯一标识。例如,"2fpa62tbms19h" 。目前有赞商城商品详情地址使用别名而不使用编号,防止无脑抓取。例如

, https://h5.youzan.com/v2/goods/3f1o7jxm0gshh 。

shopld ,店铺编号。有赞是基于 Sass 模式,支持多商户(店铺)。

3.1.2 基本信息

```
/**
 * 商品标题
 *
 * 不能超过100字,受违禁词控制
 */
private String title;
/**
```

```
* 副标题,分享链接时显示
private String summary;
/**
* 商品描述。
* 字数要大于5个字符, 小于25000个字符, 受违禁词控制
*/
@Deprecated
private String desc;
/**
* 商品分类的叶子类目编号
* 有赞——店铺主营类目和商品类目对应表: https://bbs.youzan.com/forum.php?mod=viewthread&tid=25252
private Integer cid;
/**
* 商品主图地址
* 数组, 以逗号分隔
* 建议尺寸: 800*800像素, 你可以拖拽图片调整顺序, 最多上传15张
private String picURLs;
/**
* 商品类型
* 0: 普通商品(物流发货)
* 3: UMP降价拍
* 5: 外卖商品
* 10: 分销商品
* 20: 会员卡商品
* 21: 礼品卡商品
* 22: 团购券
* 25: 批发商品
* 30: 收银台商品
* 31: 知识付费商品
* 35: 酒店商品 (无需物流)
* 40: 美业商品
* 60: 虚拟商品(无需物流)
* 61: 电子卡券 (无需物流)
*/
private Integer itemType;
/**
* 商品类型
* 0: 自营商品
* 10: 分销商品
private Integer goodsType;
```

desc ,商品描述,HTML 富文本。考虑到数据库性能,单独成 ItemContent 表,在 <u>《数据库</u> <u>实体设计 —— 商品(1.2)之商品详情》</u> 有详细分享。

cid ,商品分类的叶子类目编号。有赞——店铺主营类目和商品类目对应表

: https://bbs.youzan.com/forum.php?mod=viewthread&tid=25252 。

itemType ,商品类型。不同的商品类型,会有相关的自定义逻辑。

goodsType ,商品类型(已经不知道咋翻译好,和 itemType 无关系)。店铺可以从<u>分销市场</u>引入 其他店铺的商品进行分销。TODO 6002: 商品分销文章解析。

3.1.3 价格库存

```
/**
* 价格,单位分
private Integer price;
* 商品重量,没有SKU时用
private Double itemWeight;
* 商品货号(商家为商品设置的外部编号)
private String itemNo;
/**
* 总库存
* 基于 sku 的库存数量累加
private Integer quantity;
/**
* 总销量
*/
private Integer soldNum;
* 是否隐藏商品库存。在商品展示时不显示商品的库存。
* 0 - 显示库存(默认)
* 1 - 不显示库存
*/
private Integer hideStock;
/**
* 商品划线价格,可以自定义。例如 促销价: 888
* 商品没有优惠的情况下,划线价在商品详情会以划线形式显示。
private Double originPrice;
/**
* 是否参加会员折扣。
*1-参加会员折扣(默认)
* 0 - 不参加会员折扣
private Integer joinLevelDiscount;
```

Item SKU 相关,有赞分成两种情况:

- 第一种,无 SKU 的情况,price、itemWeight、quantity、soldNum、itemNo 对应这个商品的价格 库存等信息。
- 第二种,有 SKU 的情况,price、itemWeight、quantity、soldNum、itemNo 对应这个商品 SKU 的整体情况。其中,price 对应 SKU 最低价格。
- 。 ps: 关于第一种的情况,也可以通过虚拟没有规格的 SKU,这样和第二种的情况就可以 等价了。

price ,价格,单位为分,避免 Double 在根据营销优惠信息计算价格时,精度丢失。注意 ,如果胖友一定要使用单位为元,在 Java 里请使用 BigDecimal 。

joinLevelDiscount ,是否参加会员折扣。有赞支持会员卡功能,可以对商品进行优惠打折。

itemNo ,商品货号。商家为商品设置的外部编号,例如,ERP 系统。通过该字段,打通不同系统的信息。

3.1.4 运费信息

```
/**

* 运费类型

*

* 1-统一运费

* 2-运费模板

*/
private Integer postType;
/**

* 运费,单位分

*/
private Integer postFee;
/**

* 运费模版id

*/
private Integer deliveryTemplateId;
```

根据不同的 postType ,使用 postFee 或 deliveryTemplateId 。



运费模版支持按地区设置运费, 按购买件数计算运

运费模板的操作见: 《如何设置运费模版(按件按重量)?》 文章。

3.1.5 其他信息

```
/**

* 是否上架商品。

*

* true 为已上架

* false 为已下架

*/
private Boolean isListing;
/**

* 排序字段

*/
private Integer order;
/**

* 开始出售时间。

*

* 没设置则为空

*/
```

```
private Date autoListingTime;
/**
* 商品是否锁定。
* true 为已锁定
* false 为未锁定
private Boolean isLock;
/**
* 留言表单数组配置
* JSON 字符串 [{
   name: // 表单名,String
    required: // 是否必填,Integer,1-必填; 0-选填
    type: // 表单类型, String, 枚举: 文本格式/数字格式/邮件/日期/时间/身份证号/图片
    multiple: // 是否多行, Integer, 1-多行, 0-单行
    datetime: // 是否包含日期,用于 `type=时间`
* }]
*/
private String messages;
isListing ,是否上架商品。
  ○ true ,已上架,用户可见,直到售罄( quantity = 0 )。售罄时,该状态不变,通过库存判
    断。
  ○ false ,已下架,在仓库,用户不可见。
        商品管理
                               出售中
        商品管理
                                 发布商品
order ,排序字段。手动填写数字设置,序号越大越靠前。
       商品 价格
                                            访问量
                                                              库存
                                       访客数:1
                       【金牌代理专享
                                                            336704
                                           浏览量:1
                              【一年VIP
                              ¥ 18.80
                25
messages ,留言表单数组配置。
```

留言: 図言1 文本格式 ▼ 3行

3.1.6 ItemEtd

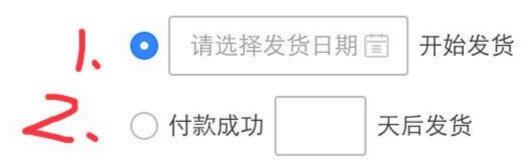
ItemEtd,预售扩展信息

```
/**
* Item 编号
* {@link Item#id}
*/
private Integer id;
/**
* 发货类型
* 0 - xxx 时间开始发货
* 1 - 付款 n 天后发货。
private Integer etdType;
/**
* 预计发货开始时间,字符串格式的时间,格式如:2018-01-01
private Date etdStartDate;
/**
* 付款成功 后发货天数,默认0
*/
private Integer etdDays;
```

《预售界面优化及新增N天后发货,适合更多预售场景》

通过 Item 的 <u>etdStatus</u> 字段,开启和关闭预售功能。 id , Item 编号,1: 1 指向对应的 Item 。 根据不同的 etdType ,使用 etdStartDate 或 etdDays 。

预售: 无现货,下单后需过段时间才能发货



只允许设置90天内的发货时间,请务必按照约定时

来自老徐的提示:

- 1. 字段以 Date 结尾,用于仅有日期格式的时间。
- 2. 字段以 Time 结尾,用于带有时间格式的时间。

3.1.7 ItemFenxiao

ItemFenxiao,分销扩展信息

```
/**

* Item 编号

*

* {@link Item#id}

*/

private Integer id;
/**

* 供货店铺Id

*/

private Integer supplierShopId;
/**

* 供货商品Id

*/

private Integer supplierItemId;

通过 Item 的 goodsType = 10 判断为分销商品。
id , Item 编号, 1: 1 指向对应的 Item 。
```

3.1.8 ItemPurchaseRight

<u>ItemPurchaseRight</u>,购买权限拓展信息

```
/**

* Item 编号

*

* {@link Item#id}

*/

private Integer id;
/**

* 允许购买的粉丝标签用,号分隔

*

* 数组,以逗号分隔

*/

private String umpTags;
/**

* 允许购买的粉丝等级,用逗号分隔

*/

private String umpLevels;
/**

* 每人限购多少件。0代表无限购,默认为0

*/

private Integer buyQuota;
```

通过 Item 的 <u>purchaseRightStatus</u> 字段,开启和关闭商品购买权限。id ,Item 编号,1: 1 指向对应的 Item 。

限购: 限制每人可购买数量

每个买家帐号累计限购



件

✓ 只允许特定用户购买

请选择拥有哪些会员卡的用户可以购买该商品

3.2 ItemSku

<u>ItemSku</u>,商品 SKU 。

规格明细:

颜色	内存		
红	sku	16G	
	sku	64G	
蓝	sku	16G	
	sku	64G	

```
/**

* 唯一编码,店铺Id 和 商品skuld 组合

*

* 分销场景下,skuld 多个店铺相同,uniqueCode 不同

*/
private String uniqueCode;
/**

* sku 编号

*

* 非唯一

*/
private Integer skuld;
/**

* 商品编号

*/
```

```
private Integer itemId;
/**
* 店铺编号
private Integer shopld;
/**
* 状态
* 1-正常
* 2-删除
*/
private Integer status;
/**
* 图片地址
private String imageURL;
/**
* 商品规格
* 格式: kid[0]-vid[0], kid[1]-vid[1]...kid[n]-vid[n]
* 例如: 20000-3275069, 1753146-3485013
private String properties;
/**
* 商品货号(商家为商品设置的外部编号)
private String itemNo;
/**
* 价格,单位分
private Integer price;
* 库存数量
*/
private Integer quantity;
* 商品在付款减库存的状态下,该Sku上未付款的订单数量
private Integer withHoldQuantity;
/**
* 销量
*/
private Integer soldNum;
/**
* 创建时间
private Date createTime;
* 更新时间
private Date updateTime;
```

uniqueCode ,SKU 唯一编码,格式为 \${shop1d}\${sku1d}。注意,在分销场景下,引入商品的 sku1d相同,而 shop1d 不同。通过该字段,保证唯一引入。

skuld ,SKU 编号,自增,非唯一,参见分销场景。

itemId , Item 编号, N: 1 指向对应的 Item 。

status ,SKU 状态。编辑商品时,当规格属性发生变化时,优先重用(保留)正常状态的 SKU ,标记删除移除的 SKU ,例如:

。 颜色: 红; 尺寸: X、L;

- 。【4】蓝-L properties ,商品规格,字符串拼接格式。

。【3】蓝−X

- 绝大多数情况下,数据库里的该字段,不存在检索的需求,更多的时候,是查询整体记录,在内存中解析使用。
- 。 少部分情况,灵活的检索,使用 Elasticsearch 进行解决。

quantity ,库存数量。在分销场景下,因为 skuld 相同,所以根据 skuld 可以批量修改,解决分销库存的同步问题。

withHoldQuantity ,商品在付款减库存的状态下(例如秒杀场景),该 SKU 上未付款的订单数量

itemNo ,商品货号。商家为商品设置的外部编号,例如,ERP 系统。通过该字段,打通不同系统的信息。

3.3 ItemSkuProperty

<u>ItemSkuProperty</u>,Item SKU 规格属性。

```
public class ItemSkuProperty {

/**

* 属性编号

*/

private Integer id;

/**

* 属性文本

*/

private String name;

/**

* 添加时间

*/

private Date addTime;

}
```

id ,属性编号。

name ,属性文本。注意,当规格名(PropertyKey)和规格值(PropertyValue)使用相同文本 ,对应的编号相同。例如,

 PropertyKey 与 PropertyValue 通过 「3.3.2 ItemSkuPropertyValueReference」 关联。

有赞的规格属性,这样的设定感觉有丢丢怪怪的,笔者后面调研了淘宝和微店的 Item SKU 规格属性的设计。

微店

- ▼ 3: {attr_title: "正面颜色", attr_values: [{attr_i attr_title: "正面颜色" ▼ attr_values: [{attr_id: 18932134, attr_value: ▼ 0: {attr_id: 18932134, attr_value: "黄色"} attr_id: 18932134
 - attr_id: 18932134 attr_value: "黄色"
- ▼ 4: {attr_title: "反面颜色", attr_values: [{attr_i attr_title: "反面颜色"
 - ▼ attr_values: [{attr_id: 18932142, attr_value: ▼ 0: {attr_id: 18932142, attr_value: "黄色"}

attr_id: 18932142 attr_value: "黄色"

分成 ItemSkuPropertyKey 和 ItemSkuPropertyValue 两个表,并且 ItemSkuPropertyValue 相同字符串时,不同编号。

淘宝

不同类目固定可选的商品规格名,并且部分商品规格不能自定义规格值。

男装>>背心/马甲

宝贝规格



☑ 选择或输入主色

宝贝规格

* 机身颜色						
选择标准颜色可增加搜索/导购机	会 ,标准颜色					
☑ 乳白色						
选择或输入主色						
* 套餐类型 不可编辑,	且不能					
□ 官方标配	□ 套餐一					
套餐四	套餐五					
□ 套餐八						
* 存储容量						
☐ 1.5GB	☐ 128MB					
256MB	288MB					
☐ 384MB	☐ 4MB					
☐ 64MB	768MB					
□ 无	☐ 4GB					

不同于有赞、微店,淘宝、京东是支持全平台进行检索,而有赞、微店是没有这方面的需求。



全部商品分类

京东服饰

美牧馆

品牌:阿迪达斯(adidas) ×

> "adidas"



adidas官方旗舰店

主营品牌: 阿迪达斯 (Adio

店铺简介: adidas is all in

运动鞋包:	休闲鞋		板鞋		
运动服饰:	运动裤		羽绒服		
尺码:	9	9.5	10	10.5	
适用人群:	男士	情侣	女	±	儿童
高级选项:	款式~	闭合方	式~	上市时间	

那么规格和类目做关联一定合适么? 笔者觉得不一定。类目和规格的关联整理是非常大的工作量 ,一旦整理缺失,反倒给上架带来不好的体验。而且,有赞、微店存在较多微商,非标的商品较多 ,需要更多灵活的空间。

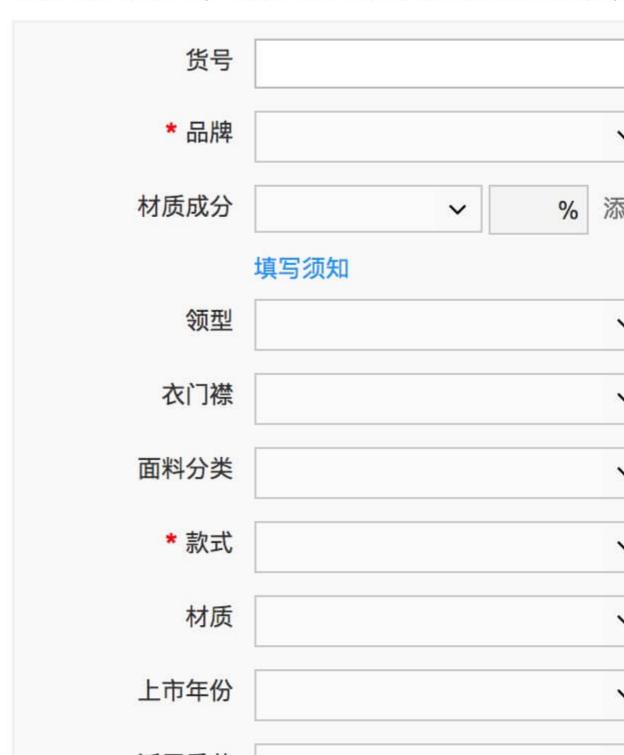
另外,淘宝商品的属性拆分的很细,猜测分成如下:

是否是规格属性是否是搜索属性是否是基本属性 … TODO 6003 淘宝属性研究

https://open.taobao.com/docs/api.htm?spm=a219a.7629065.0.0.6yhiZN&apild=4

同时,属性的类型除了包括选项完,也可以是文本、数值、百分比等等。

宝贝属性 错误填写宝贝属性,可能会引起宝贝下架或搜索流量减少,



推荐阅读 <u>《B2C电子商务系统研发——商品SKU分析和设计(一)》</u>,里面对SKU属性的管理做了很有借鉴性的思考。

3.3.1 ItemSkuPropertyKeyReference

<u>ItemSkuPropertyKeyReference</u> , Item SKU 规格名引用。用于和店铺关联,不同店铺有不同的规格名引用数据。

```
/**

* 编号

*/
private Integer id;
/**

* 店铺编号

*/
private Integer shopId;
/**

* 属性键编号

*

* {@link ItemSkuProperty#id}

*/
private Integer keyId;
/**

* 添加时间

*/
private Date addTime;
```

3.3.2 ItemSkuPropertyValueReference

<u>ItemSkuPropertyValueReference</u> , Item SKU 规格值引用。用于和店铺关联,不同店铺有不同的规格值引用数据。

```
/**
* 编号
*/
private Integer id;
* 店铺编号
private Integer shopld;
* 关联编号
* {@link ItemSkuPropertyKeyReference#id}
private Integer referenceld;
/**
* 属性键编号
* {@link ItemSkuProperty#id}
private Integer keyld;
/**
* 属性值编号
* {@link ItemSkuProperty#id}
```

```
*/
private Integer valueld;
/**

* 添加时间

*/
private Date addTime;
```

4. API

基于 <u>有赞云提供的商品API</u> ,整理如下 API 类。

4.1 ItemAPI

```
<u>ItemAPI</u> ,商品 API 。
 public interface ItemAPI {
      /**...*/
+
      YouzanItemCreateResult.ItemDetailOpenModel creat
      /**..*/
+
      YouzanItemUpdateResult update(YouzanItemUpdatePa
      /**...*/
+
      Boolean updateListing(Integer[] itemIds);
      /**..*/
+
      Boolean updatedeListing(Integer[] itemIds);
      /**...*/
+
      Boolean updateListing(Integer itemId);
      /**...*/
+
      Boolean updateDelisting(Integer itemId);
      /**...*/
+
      YouzanItemDeleteResult delete(Integer itemId);
      /**..*/
```

4.2 ItemSkuAPI

ItemSkuAPI , 商品 SKU API 。

```
public interface ItemSkuAPI {
    /**...*/
    List<YouzanSkusCustomGetResult.ItemSku> customGe
    /**...*/
    Boolean update(Integer itemId, Integer skuId, Integer
```

666. 彩蛋

第一次尝试写这种类型的文章,如果内容有错误的地方,烦请斧正。如果有写的不详细或者不明白 的地方,欢迎一起探讨。

推荐与参考文章:

```
《产品 SKU 是什么意思?与之相关的还有哪些?》
《关于电商网站数据库的设计?》
《B2C电子商务系统研发——商品模块E-R图建模》
《B2C电子商务系统研发——商品SKU分析和设计(一)》
《B2C电子商务系统研发——商品SKU分析和设计(二)》
```

文章目录

- 1. 1. 机迷
- 2. 2. 2. 背景了解
 - 1. 2.1. 2.1 名词定义
 - 2. 2. 2. 2. 2 界面
- 3. 3. 3. 数据库实体
 - 1. 3.1. 3.1 Item
 - 1. 3.1.1. 3.1.1 基础字段
 - 2. 3.1.2. 3.1.2 基本信息
 - 3. 3.1.3. 3.1.3 价格库存
 - 4. 3.1.4. 3.1.4 运费信息
 - 5. 3.1.5. 3.1.5 其他信息

- 6. 3.1.6. 3.1.6 ItemEtd
- 7. <u>3.1.7.</u> <u>3.1.7 ItemFenxiao</u>
- 8. 3.1.8. 3.1.8 ItemPurchaseRight
- 2. <u>3.2.</u> <u>3.2 ItemSku</u>
- 3. 3. 3. 3 ItemSkuProperty
 - 1. 3.3.1. 3.3.1 ItemSkuPropertyKeyReference
 - 2. 3.3.2. 3.3.2 ItemSkuPropertyValueReference
- 4. <u>4. 4. API</u>
 - 1. <u>4.1.</u> 4.1 ItemAPI
 - 2. <u>4.2. 4.2 ItemSkuAPI</u>
- 5. 5. 666. 彩蛋

2014 - 2023 芋道源码 | 总访客数 次 && 总访问量 次 回到首页