|  |
| --- |
|  |
| **商品支持系统概要设计说明书** |
|  |
|  |
| V1.0 |
| **孔祥帅** |
| **2016-2-08** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 修订历史 | | | | | |
| 版本 | 类型 | 日期 | 简要描述 | 更改人 | 审核人 |
| 1.0 | 创建 | 2016-02-08 | 创建文档 | 孔祥帅 |  |

[1 概述 6](#_Toc27747)

[1.1 业务背景 6](#_Toc12801)

[1.2 业务目标 6](#_Toc17922)

[1.3 术语及定义 6](#_Toc15266)

[1.4 参考资料 6](#_Toc25934)

[2 业务架构 6](#_Toc10834)

[3 系统上下文 6](#_Toc12312)

[4 应用架构 6](#_Toc3969)

[4.1 应用功能架构 7](#_Toc4490)

[4.2 应用场景 7](#_Toc4951)

[4.3 应用功能 7](#_Toc31780)

[4.3.1 商品管理支持Admin，admin-sdc 7](#_Toc9548)

[该模块负责品类信息的增删改查，渠道划分。包括对品类上下级关系、品类属性信息的管理、属性值信息的管理。为了实现跨品类检索，品类属性、品类属性值按照字典模式设计。为了方便用户操作，界面提供拖拽功能，修改品类所属关系。 8](#_Toc13465)

[该模块负责对品牌的查询和品牌信息的维护。包括修改品牌中文名称，品牌英文名称。 8](#_Toc1557)

[该模块负责提供商品供应商查询。 8](#_Toc29799)

[4.3.2 商品服务支持inner,inner-sdc,outer,outer-sdc 8](#_Toc7761)

[1、 外网提供服务outer：服务对象为电商网站 8](#_Toc10851)

[2、 内网提供服务inner：服务对象为导购pad，销售系统,搜索。 8](#_Toc28519)

[4.3.3 商品同步支持Syn 9](#_Toc4253)

[而商品同步支持Sync模块就是负责将来自MDM等外部系统的商品信息同步到本系统。与此同时商品同步支持Sync模块还担负着将本系统的业务数据推送回外部系统的职责。 9](#_Toc6247)

[在服务的实现方面，我们采用消息处理机制（借助MMS）。当MDM等外部系统需要推送信息到商品管理系统时，推送者需要将相关信息以消息的形式推送至消息系统，而商品同步支持Sync模块则读取消息并将其中的信息同步到本系统。 9](#_Toc18999)

[在存储的设计上，商品相关的信息直接保存至Oracle数据库中（在实现上需处理Redis中缓存数据的同步问题）。如果需要通知外部系统，此模块还需通过适当的机制将相应的业务数据推送到相应的外部系统。 例如：当库存处理服务被调用后，同步服务通过写库或调用接口的形式，将业务数据推送回其它的外部系统。 9](#_Toc30451)

[5 数据架构 9](#_Toc4392)

[5.1 数据总体架构 9](#_Toc27774)

[5.2 概念模型 9](#_Toc31501)

[5.3 逻辑模型 10](#_Toc21084)

[6 技术架构 10](#_Toc21866)

[6.1 系统分层设计 10](#_Toc15226)

[系统在设计上采用分层的原则，再结合接口实现分离的的原则，可以非常灵活的决定选择具体使用的实现方式（在接口不变的前提下）。 10](#_Toc23822)

[6.2 日志系统 10](#_Toc22714)

[系统采用logback日志系统，可以非常灵活的决定日志的记录方式，如日志文件、数据库或者其他等等。 11](#_Toc29043)

[7 系统集成架构 11](#_Toc15682)

[7.1 系统集成上下文环境 11](#_Toc11350)

[7.2 集成技术架构 11](#_Toc27207)

[本系统在设计上采用Restful的架构风格，所有的服务以Restful Service的方式开放API接口。这极大的方便了系统的集成。 11](#_Toc18140)

[另外，采用了基于消息系统的集成方案，极大的减少了系统集成需要开发的接口的数量。即使接口发生了变化，也可以通过消息转换功能轻松应对这种变化，而不会影响其它业务系统。 11](#_Toc1555)

[7.3 主要交互接口清单 11](#_Toc16632)

[7.3.1 商品相关接口 12](#_Toc90)

[7.3.2 重要业务时序图 14](#_Toc9458)

[8 逻辑及物理部署架构 16](#_Toc13851)

[8.1 逻辑部署架构 16](#_Toc20546)

[8.2 物理部署架构 16](#_Toc31130)

[8.3 运行环境定义 16](#_Toc13336)

[系统运行环境包括：开发环境、SIT环境和生产环境。从而实现开发测试、SIT及生产环境在物理层面的完全隔离。 16](#_Toc18243)

[1. 开发环境 16](#_Toc21315)

[2. SIT环境 16](#_Toc30315)

[3. 生产环境 17](#_Toc12390)

[9 架构决策 17](#_Toc27980)

[9.1 架构设计原则 17](#_Toc13963)

[王府井商务电子化架构设计原则： 17](#_Toc22957)

[1. 松耦合设计原则：各个系统独立部署，便于后期扩展 17](#_Toc14230)

[2. 基于服务化设计原则：各业务系统提供符合标准规范的服务接口，使得多业务渠道的整合和拓展成为可能 17](#_Toc15389)

[3. 数据库按照业务系统进行水平拆分：可以实现灵活地按照业务系统的需求对业务数据库进行水平拆分以满足扩展性和高性能的需求 17](#_Toc14130)

[4. 基于消息系统的系统交互：通过采用消息系统，各个业务系统可以很灵活的在同、异步模式之间进行选择，并且系统之间的交互接口数量不会随着业务系统的增加而呈现指数式增加；并且，各个业务系统可以很方便得对处理容量进行扩充 17](#_Toc16806)

[5. 通过系统冗余、分布式多级的缓存及负载均衡等技术，最大限度提供系统的负载能力、减少单点故障出现的几率，从而提高了用户的购物体验 17](#_Toc11809)

[6. 基于J2EE标准，采用开放架构和成熟的轻量级组件技术，从而提供了优秀的系统性能和灵活开速的二次开发能力 17](#_Toc2139)

# 概述

## 业务背景

随着O2O流行，统一销售平台，做到线上线下一体化。(O2O即Online To Offline（在线离线/线上到线下），是指将线下的商务机会与互联网结合，让互联网成为线下交易的平台)

## 业务目标

* 统一的商品信息维护管理服务
* 高性能的商品CRUD处理接口
* 外部系统数据同步

## 术语及定义

|  |  |  |
| --- | --- | --- |
| **缩略语/术语** | **全称** | **说明** |
| SOA | Service Oriented Architecture  面向服务的体系结构 | 面向服务的架构 |
| PCM | Product Common Management  产品共同管理 | 商品管理系统 |

## 参考资料

# 业务架构

王府井商务电子化的目标是资源的整合，包括商品、价格、库存、订单、结算等等的整合，以及营促销、订单交付等的一体化运作。

商品支持系统涵盖了产品/商品从上架、销售直到下架的整个生命周期管理所需的服务，包括了产品的上架、运作，直到产品下架。而且，商品支持系统包含了运作中所需的价格、库存、供应商等管理所需的服务。

# 系统上下文

# 应用架构

## 应用功能架构

商品支持系统PCM提供的服务按照功能可以分为三大模块：

1. 商品管理支持Admin,admin-sdc
2. 商品服务支持Service（服务）
3. 商品同步支持Syn（同步）
4. 商品对内支持 inner（对内），inner-sdc
5. 商品对外支持 outer（对外），outer-sdc

D:\快盘\王府井\开发系统\商品系统\拓扑图\商品系统架构图.emf

## 应用场景

## 应用功能

### 商品管理支持Admin，admin-sdc

提供管理商品主数据信息，品类信息，品牌信息等管理界面。用于业务人员日常维护用，

1. **品类管理**

该模块负责品类信息的增删改查，渠道划分。包括对品类上下级关系、品类属性信息的管理、属性值信息的管理。为了实现跨品类检索，品类属性、品类属性值按照字典模式设计。为了方便用户操作，界面提供拖拽功能，修改品类所属关系。

1. **品牌管理**

该模块负责对品牌的查询和品牌信息的维护。包括修改品牌中文名称，品牌英文名称。

1. **供应商管理**

该模块负责提供商品供应商查询。

1. **商品单品管理**

该模块提供相应的接口支持对商品的信息编辑，包括添加商品描述，添加包装清单，添加商品运营信息(是否自营，是否支持价保等)，设定商品的上下架时间，商品的品类设置，标签设置，图片设置等。

### 商品服务支持inner,inner-sdc,outer,outer-sdc

商品服务模块，PCM系统的核心，有两部分组成:

1. 外网提供服务outer：服务对象为电商网站
2. 内网提供服务inner：服务对象为导购pad，销售系统,搜索。

将服务分成两部分，是为了提高系统的可用性。如果外部系统因压力过大而出现问题，为内部系统提供的服务可以不受影响。

该模块为外部系统提供了：商品信息查询、商品库存查询、商品品类查询以及商品品牌查询等服务接口。

为内部服务提供了：商品信息查询、商品库存查询、商品品类查询、商品品牌查询以及库存修改等服务接口。

在具体实现上，该模块采用Redis + mysql存储模型。对于查询相关的服务接口，在实现上采用了先Redis缓存后mysql的原则。这样就能在很大程度上降低此类服务对于Oracle数据库的压力。

对于数据更新类服务接口，在实现的时候需要处理Redis中缓存数据的失效问题。更新操作直接更新Oracle中的数据，然后让Redis中的缓存数据失效以确保Redis缓存中的数据与Oracle中的数据是一致的。

### 商品同步支持Syn

商品信息是从其它系统创建的，然后同步到MDM（主数据）。因此商品管理系统需要提供服务接口，将来自MDM的商品信息同步到商品管理系统。

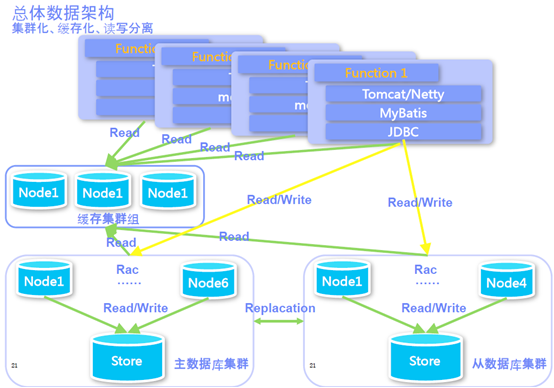
而商品同步支持Sync模块就是负责将来自MDM等外部系统的商品信息同步到本系统。与此同时商品同步支持Sync模块还担负着将本系统的业务数据推送回外部系统的职责。

在服务的实现方面，我们采用消息处理机制（借助MMS（信息服务））。当MDM等外部系统需要推送信息到商品管理系统时，推送者需要将相关信息以消息的形式推送至消息系统，而商品同步支持Sync（同步）模块则读取消息并将其中的信息同步到本系统。

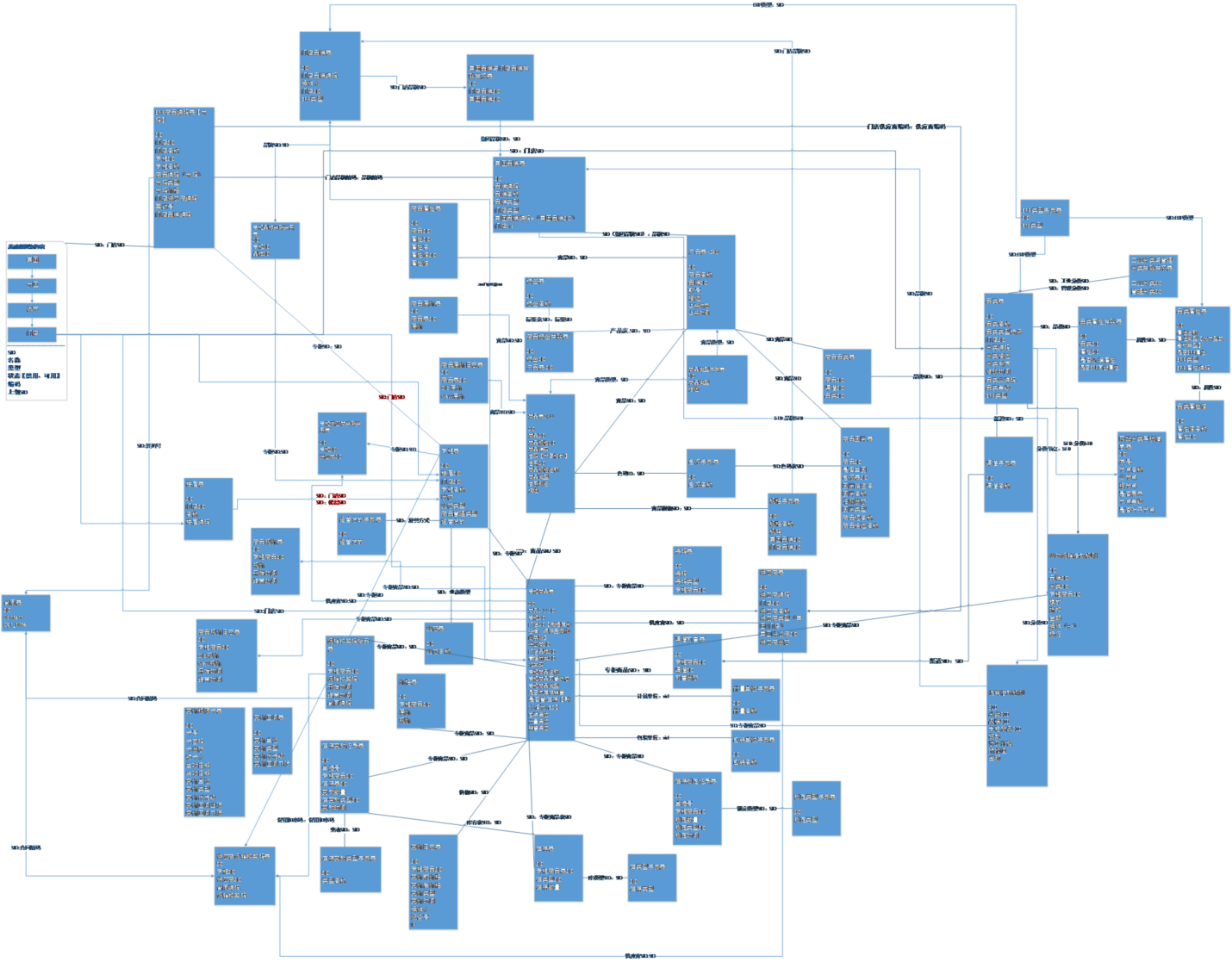
在存储的设计上，商品相关的信息直接保存至Oracle数据库中（在实现上需处理Redis中缓存数据的同步问题）。如果需要通知外部系统，此模块还需通过适当的机制将相应的业务数据推送到相应的外部系统。 例如：当库存处理服务被调用后，同步服务通过写库或调用接口的形式，将业务数据推送回其它的外部系统。

# 数据架构

## 数据总体架构



## 概念模型



## 逻辑模型

# 技术架构

## 系统分层设计

系统在设计上采用分层的原则，再结合接口实现分离的的原则，可以非常灵活的决定选择具体使用的实现方式（在接口不变的前提下）。

## 日志系统

系统采用logback日志系统，可以非常灵活的决定日志的记录方式，如日志文件、数据库或者其他等等。

# 系统集成架构

## 系统集成上下文环境

## 集成技术架构

本系统在设计上采用Restful的架构风格，所有的服务以Restful Service的方式开放API接口。这极大的方便了系统的集成。

Restful的架构风格：主要用于客户端和服务器交互类的软件。基于这个风格设计的软件可以更简洁，更有层次，更易于实现缓存等机制。

另外，采用了基于消息系统的集成方案，极大的减少了系统集成需要开发的接口的数量。即使接口发生了变化，也可以通过消息转换功能轻松应对这种变化，而不会影响其它业务系统。

## 主要交互接口清单

### 商品相关接口

#### 商品相关接口

7‑1商品相关接口

#### 产品相关接口

7‑2产品相关接口

#### 品类相关接口

7‑3品类相关接口

#### 品牌相关接口

7‑4品牌相关接口

#### 库存相关接口

7‑5库存相关接口

### 重要业务时序图

#### 商品主数据信息同步

7‑6MDM同步到PCM

#### 商品信息维护

商品信息编辑

7‑7商品信息维护

#### 商品上下架

商品上架流程

7‑8商品上架申请流程图

商品确认上架流程图

7‑9商品确认上架流程图

库存同步流程图

7‑10商品上架库存同步流程图

#### 商品库存操作

商品占库流程

7‑11商品占库流程图

# 逻辑及物理部署架构

## 逻辑部署架构

## 物理部署架构

## 运行环境定义

系统运行环境包括：开发环境、SIT环境和生产环境。从而实现开发测试、SIT及生产环境在物理层面的完全隔离。

1. 开发环境

操作系统： Linux

数据库：mysql

缓存系统：Redis

1. SIT环境

操作系统： Linux

数据库：mysql

缓存系统：Redis

1. 生产环境

操作系统： Linux

数据库：mysql

缓存系统：Redis

# 架构决策

## 架构设计原则

王府井商务电子化架构设计原则：

1. 松耦合设计原则：各个系统独立部署，便于后期扩展
2. 基于服务化设计原则：各业务系统提供符合标准规范的服务接口，使得多业务渠道的整合和拓展成为可能
3. 数据库按照业务系统进行水平拆分：可以实现灵活地按照业务系统的需求对业务数据库进行水平拆分以满足扩展性和高性能的需求
4. 基于消息系统的系统交互：通过采用消息系统，各个业务系统可以很灵活的在同、异步模式之间进行选择，并且系统之间的交互接口数量不会随着业务系统的增加而呈现指数式增加；并且，各个业务系统可以很方便得对处理容量进行扩充
5. 通过系统冗余、分布式多级的缓存及负载均衡等技术，最大限度提供系统的负载能力、减少单点故障出现的几率，从而提高了用户的购物体验
6. 基于J2EE标准，采用开放架构和成熟的轻量级组件技术，从而提供了优秀的系统性能和灵活开速的二次开发能力