

浙江大学

ZHEJIANG UNIVERSITY

软件需求规格说明书

垂直搜索引擎

组长：马梓睿

组员：程梦烨 毕予然 沈乐明 李楠

修改历史

| 修订日期 | 版本号 | 作者 | 修改内容 | 审核者 |
|-----------|-------|--------------------|------|-----|
| 2021.7.11 | 1.0.0 | 马梓睿、程梦烨、毕予然、沈乐明、李楠 | 创建 | 沈乐明 |
| 2021.7.12 | 1.1.0 | 马梓睿、程梦烨、毕予然、沈乐明、李楠 | 需求变更 | 马梓睿 |
| 2021.7.12 | 1.1.1 | 马梓睿、程梦烨、毕予然、沈乐明、李楠 | 修正错误 | 程梦烨 |

目录

| | |
|-------------------------|----|
| 修改历史 | 2 |
| 1 引言 | 6 |
| 1.1 编写目的 | 6 |
| 1.2 项目背景 | 6 |
| 1.3 预期的读者和阅读建议 | 6 |
| 1.4 参考资料 | 7 |
| 2 总体描述 | 8 |
| 2.1 产品基本信息 | 8 |
| 2.2 产品前景 | 8 |
| 2.3 市场分析 | 8 |
| 2.4 用户类及其特征 | 9 |
| 2.5 产品功能概述 | 10 |
| 2.5.1 普通搜索功能模块 | 10 |
| 2.5.2 高级搜索功能模块 | 10 |
| 2.5.3 搜索结果显示模块 | 10 |
| 2.5.4 更新关键词功能模块 | 10 |
| 2.5.5 更新知识图谱模块 | 10 |
| 2.5.6 更新爬虫网站源功能模块 | 10 |
| 2.6 设计和实现上的约束 | 11 |
| 2.6.1 数据存储 | 11 |
| 2.6.2 网络服务吞吐 | 11 |
| 2.6.3 数据安全 | 11 |
| 2.7 假设和依赖 | 11 |
| 2.8 开发、测试和运行环境 | 11 |
| 2.8.1 开发环境 | 11 |
| 2.8.2 测试环境 | 12 |
| 2.8.3 运行环境 | 12 |
| 2.9 用户文档 | 12 |
| 2.9.1 描述类文档 | 12 |

| | | |
|--------|------------------|----|
| 2.9.2 | 过程类文档 | 12 |
| 2.9.3 | 参考类文档 | 12 |
| 2.10 | 产品开发项目设计人员 | 12 |
| 3 | 系统需求分析概述 | 14 |
| 3.1 | 需求优先级分析 | 14 |
| 3.2 | 系统模块架构图 | 16 |
| 4 | 功能需求 | 17 |
| 4.1 | 用户进行普通搜索 | 17 |
| 4.1.1 | 关键词搜索 | 17 |
| 4.1.2 | 关键词过滤 | 19 |
| 4.1.3 | 拼音搜索 | 21 |
| 4.1.4 | 结果显示 | 23 |
| 4.1.5 | 图片、视频显示 | 25 |
| 4.1.6 | 结果相关度排序 | 27 |
| 4.1.7 | 结果销量排序 | 29 |
| 4.1.8 | 结果价格排序 | 31 |
| 4.1.9 | 同类商品推荐 | 33 |
| 4.1.10 | 搜索热榜 | 35 |
| 4.2 | 用户进行高级搜索 | 37 |
| 4.2.1 | 关键词并集 | 37 |
| 4.2.2 | 关键词交集 | 39 |
| 4.2.3 | 优先补全 | 41 |
| 4.2.4 | 搜索提示 | 43 |
| 4.2.5 | 关键词条件筛选 | 45 |
| 4.2.6 | 关键词模糊搜索 | 47 |
| 4.3 | 用户管理关键词 | 49 |
| 4.3.1 | 用户查看关键词 | 49 |
| 4.3.2 | 用户增加关键词 | 51 |
| 4.4 | 用户管理知识图谱 | 53 |
| 4.4.1 | 用户更新知识图谱 | 53 |
| 4.5 | 用户管理网站源 | 55 |

| | | |
|-------|-----------------|----|
| 4.5.1 | 用户查看爬虫网站源 | 55 |
| 4.5.2 | 用户更新爬虫网站源 | 57 |
| 5 | 数据流图 | 59 |
| 6 | 外部接口需求 | 60 |
| 6.1 | 硬件接口 | 60 |
| 6.1.1 | 服务器 | 60 |
| 6.1.2 | 客户端 | 60 |
| 6.2 | 软件接口 | 60 |
| 6.2.1 | 服务器 | 60 |
| 6.2.2 | 客户端 | 60 |
| 6.3 | 通信接口 | 60 |
| 7 | 非功能性需求 | 61 |
| 7.1 | 性能需求 | 61 |
| 7.2 | 输入要求 | 61 |
| 7.3 | 数据传输与并发要求 | 61 |
| 7.4 | 数据管理要求 | 61 |
| 7.5 | 权限与安全需求 | 61 |
| 7.6 | 可视化需求 | 61 |
| 7.7 | 防护性需求 | 61 |
| 7.8 | 软件质量属性 | 61 |
| 7.9 | 其他需求 | 61 |
| 8 | 数据字典 | 62 |
| 8.1 | 数据流定义表 | 62 |
| 8.2 | 数据元素定义表 | 62 |
| 8.3 | 数据精度表 | 62 |
| 9 | 业务规则与业务算法 | 63 |
| 9.1 | 业务规则 | 63 |
| 9.2 | 业务算法 | 63 |
| 10 | 附录 | 64 |
| 10.1 | 术语表 | 64 |
| 10.2 | 待确定问题清单 | 99 |

1 引言

1.1 编写目的

本软件需求规格说明书对垂直搜索引擎的实现做了细致全面的用户需求分析，针对消费者、商家等多个不同的用户群体，清晰地描述了系统的各项需求，包括功能性需求和非功能性需求。

本软件需求规格说明书旨在明确我们需要开发的垂直搜索引擎应该具有的基本概貌、功能要求、性能分析和运行要求等等，并且尽可能完整地描述垂直搜索引擎预期的外部行为和用户可视化行为，使得项目经理和软件开发人员以及系统分析师可以清楚地了解不同用户群体的不同需求，并在此基础上进一步提出概要设计说明书和完成后续设计与开发的工作。

1.2 项目背景

垂直搜索引擎是相对通用搜索引擎的信息量大、查询不准确、深度不够等提出来的新的搜索引擎服务模式。通过针对某一特定需求、某一特定人群或某一特定领域提供的有一定价值的信息和相关服务。

商业智能化（BI）是由加特纳集团于 1996 年提出的一种概念。其旨在通过一种智能化的方法，基于已有事实来辅助商业决策的制定。通过对数据的整合分析，BI 系统能从中提取到有用的信息，为企业提供更优的商业决策参考。

如今，包括微软、IBM、Oracle 等多个大型企业均提供一定程度上的商业智能解决方案，BI 市场年均增长率达到 27%，说明这是一个快速发展的行业。

商业智能数据涵盖广泛。从企业自身的经营数据到行业的大数据，再到每一件商品的销售数据等等都会为商业决策提供一定的参考。这种数据源的广泛性也代表了数据的多样性，以及商业智能市场亟待挖掘更多的机会，说明商业智能是一个广阔的市场。

商品购物是消费者的日常需要。本项目旨在开发一个商品购物的搜索引擎，为消费者提供从多种在线零售网站中进行商品价格、网站信誉、购物方便性等方面的比较资料。其作用不仅表现在为在线消费者提供方便，也为在线销售上推广产品提供了机会，获得的信息更加集中更全面。该购物搜索引擎的检索结果来自于网上购物网站，如淘宝、京东、亚马逊等，这样当用户检索某个商品时，所有销售该商品的网站上的产品记录都会被检索出来，用户可以根据产品价格、对网站的信任和偏好等因素进入所选择的网上购物网站购买产品，而一般来说，购物搜索引擎本身并不出售这些商品。

1.3 预期的读者和阅读建议

- 项目经理：项目经理可以根据本文档了解预期产品的功能，并据此进行了系统设计和项目管理
- 设计人员：系统的设计人员可以根据本文档对需求进行分析，并设计出系统，包含系统架构的设计和数据库表的设计

- 程序人员：程序人员可以根据本文档配合《系统总体设计计划》，了解系统功能，并编写《用户手册》
- 销售人员：销售人员可以根据本文档了解目标产品的各项功能和性能
- 测试人员：测试人员可以根据本文档编写测试用例，并对软件产品进行功能性的测试和非功能性的测试
- 用户：用户可以根据本文档了解本产品的功能和性质，并和分析人员进行需求的讨论和协商
- 其他人员：其他人员比如部门领导、公司领导、政府机关领导等领导可以根据本文档了解产品的功能和性能

在阅读本软件需求规格说明书的时候，首先要了解产品的功能概貌，然后可以根据自身的需要对每一功能进行实当的了解。

1.4 参考资料

- 《软件设计文档国家标准》
- 《软件工程项目开发文档范例》
- 《软件需求》刘伟琴、刘洪涛译
- 《“项目实训”课程案例 垂直搜索引擎 20210709》刑卫
- 《[G10]“垂直搜索引擎”项目计划书》

2 总体描述

2.1 产品基本信息

- **产品名称**
商品垂直搜索引擎
- **任务提出者**
浙江大学项目实训任课老师——邵健
- **开发者**
浙江大学 软件工程项目实训 G10 小组
- **用户**
有检索商品信息需求的供应商
- **功能**
根据关键词检索商品、支持高级搜索、展示搜索结果、筛选搜索结果、排序搜索结果、用户添加关键词、网页源、查看并更新知识图谱等
- **性能**
能够支撑至少 1000 名用户的使用需求，允许 20 人同时下载资源，平均速度达到 5Mbps
- **完成期限**
7 月 23 日（项目截止日期）前

2.2 产品前景

在商业竞争中，信息的不对称性一直是获利的重要来源。商家如果能从繁杂纷乱的信息中分析出对自己有利的信息，则会极大的提升自己在市场竞争中的优势，进而产生垄断效应，大幅提升商业竞争力。

比较购物是一种非常流行的购物方式。消费者综合判断价格、质量、售后服务等多个维度，进行理性判断后挑选出令人满意的商品。比较购物由于促进了商家间的竞争，推动了便于比较购物的网络购物核心竞争力的发展，降低了商家和消费者间信息的不对称性。比较购物知识图谱的构建，可以为消费者提供更加优质的购物体验。

基于以上两点，购物场景下的知识图谱垂直搜索引擎的构建有其商业价值。

2.3 市场分析

对比通用搜索引擎：通用搜索引擎除了商品有其他，对于商品具体属性涉及较弱，信息量大、查询不准确、深度不够；本搜索引擎聚焦商品，具有商品的深度信息，并结合消费者、供应商对于商品的买、卖两个方向的需求。

对比淘宝等单一网站：只有单一网站、单一信息源，本搜索引擎可以提供多个网站，并且可以满足用户自主选择网站源的需要；并且与购买渠道分割开来，更具有可信度。

PowerBI：优势：微软旗下，功能强大，类 SQL 语法可以实现复杂分析，云平台加持，可以收集多种数据源，同时可视化较为精美；劣势：价格昂贵，二维表模式对非专业用户不够友好；特色功能：通过编程实现复杂分析

WolframAlpha: 优势：人工智能技术加持，拥有世界最大的知识图谱系统，通过自然语言处理技术可以分析自然语言并对其进行问答；劣势：对购物场景涉猎较少，同时不支持中文问答；特色功能：自然语言问答

云听 CEM: 优势：实时采集多个电商平台的价格，可以进行智能价格分析，提供可视化图表相关功能；劣势：只有价格维度，较为单一，缺乏自定义分析功能，缺乏自定义网站添加功能，价格昂贵；特色功能：智能分析价格变化

Chrome 插件: 优势：与购物网站集成，无缝购物体验，无需安装；劣势：功能较为单一，无交互相关功能，网站较为固定，缺乏长期记录功能，可替代性强，难以形成用户粘性；特色功能：获得隐藏优惠券

一淘: 优势：阿里旗下，与淘宝生态深度集成，提供一站式购物体验，用户界面较为精美，可以快速收集大量促销信息；劣势：非第三方，数据有不客观的嫌疑，同时缺乏数据分析和比价功能，只能被动接受促销信息；特色功能：正在促销的商家

ShopSavvy: 优势：通过条形码获取商品信息，便于线下场景的购物比价；劣势：缺乏分析功能；特色功能：条形码获取商品数据

可见，目前市面上绝大多数竞品都存在以下几个问题：

1. 缺少多维度分析功能或使用较为复杂
2. 目标网站较为固定
3. 获取的信息品种较为单一

我们的产品通过以下几个方面来对这些问题进行改进。

选择面更广：由于不局限于单一网站源，消费者选择的面更广，更有可能买到最心仪的商品；每个商品都附有购买链接（如指向淘宝）这个属性，也方便消费者的购买。

选择方式更便捷：高级搜索功能支持关键词交集、并集、条件筛选、模糊搜索等，方便消费者更方便地快速筛选。

自由度更大：可以自定义关键词、自定义网站源和爬取方式，对于需要什么商品、需要什么来源的信息有着极高的选择权与自由度。

2.4 用户类及其特征

表 2-1 用户类及其特征

| 使用者 | 主要特征 | 备注 |
|-----|---------------|---------------------------|
| 用户 | 可以通过关键词搜索商品信息 | 支持高级搜索，指布尔运算等，支持类型筛选等筛选功能 |

| | | |
|--|--------------|------------------------------|
| | 可以查看搜索结果 | 搜索结果应包括商品名、价格、销量、购买链接等 |
| | 可以对搜索结果进行筛选 | 筛选可以根据价格、销量等 |
| | 可以对搜索结果进行排序 | 排序可以根据价格、销量等 |
| | 可以查看结果的基本信息 | 基本信息包括商品品牌、产地、支付方式等 |
| | 可以查看商品的评分与评价 | 可以看到客观的评分与五条买家评论 |
| | 可以添加关键词 | 在搜索不到某商品的情况下可以自行添加该关键词到关键词库中 |
| | 可以添加网站 | 除了淘宝，用户还可自己添加其他网站作为网站源进行搜索 |
| | 可以查看并更新知识图谱 | 增加灵活度 |

2.5 产品功能概述

2.5.1 普通搜索功能模块

- 用户进入搜索界面，在输入框输入关键词，页面返回搜索到的商品列表。
- 用户还可以对商品价格、销量等进行排序。
- 用户还可以看到同类商品推荐、查看搜索热榜等。

2.5.2 高级搜索功能模块

- 用户可以根据关键词筛选进行高级搜索，包括利用关键词交集、并集、条件筛选和模糊搜索，利用优先补全、搜索提示。

2.5.3 搜索结果显示模块

- 用户进入搜索界面，在输入框输入关键词后，可以查看页面返回搜索到的商品列表结果。
- 用户可看到每一个商品的具体属性，如价格、销量、库存、颜色分类、运费等，并可点击图片或视频进行查看，也可进入购买链接进行查看。

2.5.4 更新关键词功能模块

- 用户若在搜索框里搜索不到某商品，可以向关键词库中加入新的关键词。

2.5.5 更新知识图谱模块

- 用户可以请求重新进行爬取商品信息、重新更新图数据库，此时爬虫操作会对于关键词库中的关键词、网站源库中的网站进行逐一爬取。

2.5.6 更新爬虫网站源功能模块

- 除了淘宝，用户若想获取京东或其他网站上的商品信息，可自行添加爬虫网站源，并需要同时添加该网站爬取脚本。

2.6 设计和实现上的约束

2.6.1 数据存储

利用统一的数据库，构建图数据库。

采用的数据库：OrientDB

2.6.2 网络服务吞吐

提供一定网络服务吞吐量的对外服务的能力。

吞吐数量：确保同时为至少 10000 名用户提供服务的能力。

2.6.3 数据安全

为确保经过网络传输和交换的数据不会发生增加、修改、丢失和泄露等，我们需要确保数据的可用性、完整性和保密性。

- 可用性：让产品的设计能够符合使用者的习惯与需求。在设计网站方面，使用户在浏览的过程中不会产生压力或感到挫折，并能让使用者在使用网站功能时，能用最少的努力发挥最大的效能；在对数据的获取方面，采用身份验证的方法，避免因系统数据泄露而使得合法使用者无法接触可用数据。
- 完整性：指在传输、存储信息或数据的过程中，确保信息或数据不被未授权的篡改或在篡改后能够被迅速发现。采用安全框架和加密技术。
- 保密性：是指个人或团体的信息不为其他不应获得者获得。采用加密技术，对数据进行加密和权限设置，使得只有被授权者获取正确的密码才能访问和使用。

2.7 假设和依赖

表 2-2 假设和依赖

| 假设和依赖编号 | 假设和依赖内容 |
|---------|--|
| AS1 | 平台开发人员具备足够的开发能力和学习能力 |
| AS2 | 平台开发人员具有明确的职责与分工，在多次迭代过程中的工作分配覆盖所有工作任务 |
| AS3 | 平台开发时间足够、经费充足，开发工具可获取 |
| AS4 | 平台用户具有足够的电脑操作能力 |
| AS5 | 平台具有足够的服务器支持 |
| AS6 | 平台可以访问搜索结果信息相关数据库，获取所需数据等 |
| AS7 | 平台具备实时更新信息的能力 |

2.8 开发、测试和运行环境

2.8.1 开发环境

- Windows10 Pro 操作系统
- PyCharm, IntelliJ IDEA, Flask, Selenium, Vue, OrientDB

- Linux Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-31-generic x86_64)

2.8.2 测试环境

- 联网的 PC、天翼云服务器、相关开源测试工具（如 Apache JMeter, LoadRunner）

2.8.3 运行环境

- 联网的 PC
- 浏览器：Google Chrome, Mozilla Firefox, Microsoft Edge, Safari

2.9 用户文档

用户文档是软件开发人员为软件用户了解、使用、操作和维护等提供的详细资料。用户文档的编制需要一下几个要求：

- 描述规范准确：需要对用户有较强的实用和指导意义，用词准确、规范、统一；
- 叙述简练生动：由于用户文档的阅读对象是普通非计算机专业人员，因此需要简洁精炼，少专业术语，力求图文并茂，方便用户学习、理解和掌握实用方法。
- 内容完整：应覆盖软件整体结构、功能、安装、运行和操作的相关知识，并组织成有系统性、层次性的文档。

本垂直搜索引擎中的用户文档主要帮助用户可以了解本系统并快速上手，并在遇到实际问题时通过文档查阅快速解决问题。

我们将用户文档分为三类，分别进行说明。

2.9.1 描述类文档

- 目的：帮助用户概括性地了解本垂直搜索引擎平台具备的功能及其使用方法。
- 内容：功能、属性、接口、特性。

2.9.2 过程类文档

- 目的：引导用户在第一次使用系统时的操作。
- 内容：通过用户在第一次登录系统时以及第一次使用某种功能的引导式教学

2.9.3 参考类文档

- 目的：帮助用户解决问题，指导某具体功能的详细操作细节。
- 内容：可能遇到的问题及其解决方式，各功能的细节操作流程。

2.10 产品开发项目设计人员

表 2-1 项目开发角色分配

| 角色 | 职责 | 人员 |
|------|-------------------------------------|-----|
| 项目经理 | 在预算范围内按时优质地领导项目小组完成全部项目工作内容，并使客户满意。 | 马梓睿 |

| | | |
|--------|--|------|
| 产品经理 | 负责市场调查并根据用户的需求，确定开发何种产品，选择何种技术、商业模式等，根据产品的生命周期，协调研发、营销、运营等，确定和组织实施相应的产品策略。 | 马梓睿 |
| 设计总监 | 建立系统框架；数据库设计；概要设计；参加技术评审； | 程梦烨 |
| 测试经理 | 组织编写测试计划和测试方案，组织系统测试；参加技术评审； | 毕予然 |
| 美工 | 设计网站原型 | 程梦烨 |
| 质量经理 | 带领软件质量监督组成员制定质量保证计划，对监督组反映的质量问题进行汇总与产品经理、项目经理进行交流，当新的问题出现时最终由质量经理决定处理方式。 | 沈乐明 |
| 开发人员 | 负责进行编码工作与单元测试，进行系统集成，及时解决测试时出现的问题 | 全体成员 |
| 测试人员 | 编写测试方案和测试用例，进行系统测试，向开发组反馈 BUG。 | 全体成员 |
| 软件质量监督 | 实时对质量经理以及项目经理提供项目进度与项目实际开发时的差异提出报告，指出差异原因和改进方法。 | 李楠 |

3 系统需求分析概述

根据产品功能概述我们划分了系统的模块 并从 每个模块中挑选出了必要的需求作为用例进行分析。本部分我们将对这些需求进行优先级分析并给出系统模块架构图。

3.1 需求优先级分析

本需求优先级分析模块依据风险 价值四象限矩阵进行评级，即按照表 3-1 所示四象限对需求进行优先级分类。

表 3-1 需求优先级风险-价值矩阵

| | 高价值 | 低价值 |
|-----|------|------|
| 高风险 | 首先处理 | 避免 |
| 低风险 | 其次处理 | 最后处理 |

考虑到实际情况的限制，以下估算仅作为策略性指导，随着项目的进行可能会进行调整。具体的优先级分析如表 3-2 所示。

表 3-2 用例的需求优先级分析

| 用例编号 | 用例名称 | 风险 | 价值 | 优先级 |
|----------|---------|----|----|-----|
| SR-GS-01 | 关键词搜索 | 中 | 高 | 高 |
| SR-GS-02 | 关键词过滤 | 中 | 中 | 中 |
| SR-GS-03 | 拼音搜索 | 中 | 高 | 高 |
| SR-GS-04 | 结果显示 | 中 | 高 | 高 |
| SR-GS-05 | 图片、视频展示 | 中 | 中 | 中 |
| SR-GS-06 | 结果相关度排序 | 高 | 高 | 中 |
| SR-GS-07 | 结果销量排序 | 高 | 高 | 中 |
| SR-GS-08 | 结果价格排序 | 高 | 高 | 中 |
| SR-GS-09 | 同类商品推荐 | 中 | 高 | 高 |
| SR-GS-10 | 搜索热榜 | 中 | 中 | 中 |
| SR-AS-01 | 关键词交集 | 中 | 中 | 中 |
| SR-AS-02 | 关键词并集 | 中 | 中 | 中 |
| SR-AS-03 | 优先补全 | 中 | 中 | 中 |
| SR-AS-04 | 搜索提示 | 低 | 中 | 中 |
| SR-AS-05 | 关键词条件筛选 | 中 | 中 | 中 |
| SR-AS-06 | 关键词模糊搜索 | 低 | 中 | 中 |
| UD-KW-01 | 用户查看关键词 | 中 | 高 | 高 |
| UD-KW-01 | 用户增加关键词 | 低 | 中 | 中 |

| | | | | |
|----------|-----------|---|---|---|
| UD-KM-01 | 用户更新知识图谱 | 低 | 中 | 中 |
| UD-WB-01 | 用户查看爬虫网站源 | 中 | 高 | 高 |
| UD-WB-02 | 用户更新爬虫网站源 | 低 | 中 | 中 |

表 3-3 QFD 图 (9 分计)

| 用例编号 | 用例名称 | 价值 | 未完成带来的损失 | 成本 | 风险 | 优先级 |
|----------|-----------|----|----------|----|----|-----|
| SR-GS-01 | 关键词搜索 | 9 | 9 | 5 | 5 | 高 |
| SR-GS-02 | 关键词过滤 | 8 | 6 | 7 | 8 | 中 |
| SR-GS-03 | 拼音搜索 | 8 | 6 | 8 | 7 | 高 |
| SR-GS-04 | 结果显示 | 9 | 9 | 4 | 5 | 高 |
| SR-GS-05 | 图片、视频展示 | 8 | 7 | 5 | 6 | 中 |
| SR-GS-06 | 结果相关度排序 | 8 | 8 | 6 | 6 | 中 |
| SR-GS-07 | 结果销量排序 | 7 | 7 | 6 | 6 | 中 |
| SR-GS-08 | 结果价格排序 | 7 | 7 | 6 | 6 | 中 |
| SR-GS-09 | 同类商品推荐 | 6 | 6 | 8 | 7 | 高 |
| SR-GS-10 | 搜索热榜 | 6 | 6 | 7 | 7 | 中 |
| SR-AS-01 | 关键词交集 | 8 | 6 | 7 | 8 | 中 |
| SR-AS-02 | 关键词并集 | 8 | 6 | 7 | 8 | 中 |
| SR-AS-03 | 优先补全 | 6 | 7 | 7 | 7 | 中 |
| SR-AS-04 | 搜索提示 | 6 | 4 | 8 | 8 | 中 |
| SR-AS-05 | 关键词条件筛选 | 6 | 4 | 7 | 7 | 中 |
| SR-AS-06 | 关键词模糊搜索 | 6 | 4 | 8 | 8 | 中 |
| UD-KW-01 | 用户查看关键词 | 8 | 8 | 6 | 6 | 高 |
| UD-KW-01 | 用户增加关键词 | 6 | 6 | 7 | 7 | 中 |
| UD-KM-01 | 用户更新知识图谱 | 6 | 6 | 7 | 7 | 中 |
| UD-WB-01 | 用户查看爬虫网站源 | 8 | 8 | 6 | 6 | 高 |
| UD-WB-02 | 用户更新爬虫网站源 | 6 | 6 | 7 | 7 | 中 |

3.2 系统模块架构图

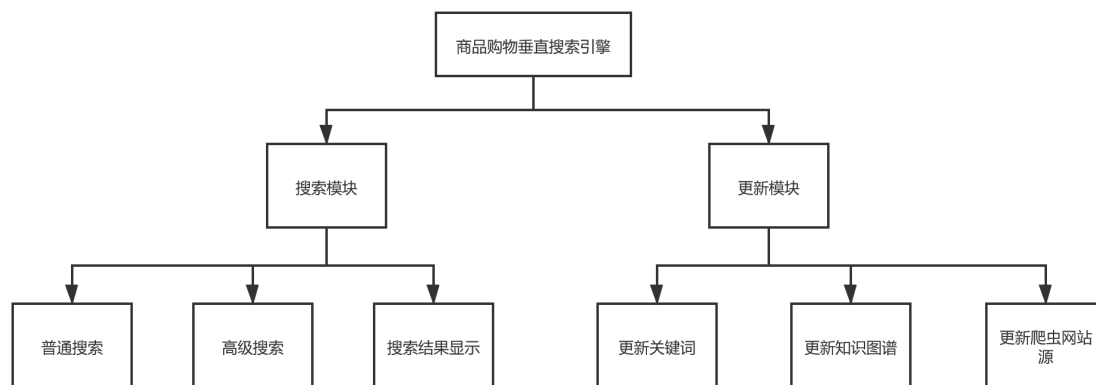


图 3-1 系统模块架构图

4 功能需求

本部分以用户类为区分对每个功能需求用例做了详细的描述并给出了时序图。特别说明，由于存在一些共性的用户行为，我们特别使用“用户”这一用户类来包括教师、助教、学生、游客以及管理员这五类用户的组合，它们的共同特征与需求都在“用户”这一用户类中阐述。

4.1 用户进行普通搜索

4.1.1 关键词搜索

表 4-1-1 关键词搜索用例描述

| 用例编号 | SR-GS-01 | 用例名称 | 关键词搜索 |
|-------|---|--------|-----------|
| 创建人 | 毕予然 | 最后修改人 | 毕予然 |
| 创建日期 | 2021/7/10 | 最后修改日期 | 2021/7/10 |
| 角色 | 用户 | 需求来源 | 用户代表 |
| 主要参与者 | 用户 | | |
| 描述 | 用户在搜索界面的文本框中输入关键词，点击搜索按钮可以进行搜索 | | |
| 前置条件 | 1. 系统正常运行 2. 用户可正常打开相关页面 | | |
| 触发器 | 用户需要进入搜索页面 | | |
| 后置条件 | 用户搜索记录写入网站日志 | | |
| 输入信息 | 搜索内容 | | |
| 主干过程 | 1. 用户进入搜索页面 2. 用户输入关键词 3. 用户点击搜索按钮 4. 搜索成功后跳转至结果显示界面 | | |
| 分支过程 | 无 | | |
| 异常 | 1. 用户输入危险信息（发生于主干过程 2） 处理：系统提示“查询错误，请重新输入” 系统结束用例 2. 搜索后页面出现异常（发生于主干过程 4） 处理：根据网站的联系方式联系系统管理员 系统结束用例 | | |
| 假设 | 无异常发生 | | |
| 输入 | 关键词、语句 | | |
| 输出 | 搜索结果显示页面 | | |
| 包括用例 | 搜索过滤、搜索并集、搜索交集 | | |
| 优先级 | 高 | | |
| 使用频率 | 高 | | |
| 备注与问题 | 无 | | |

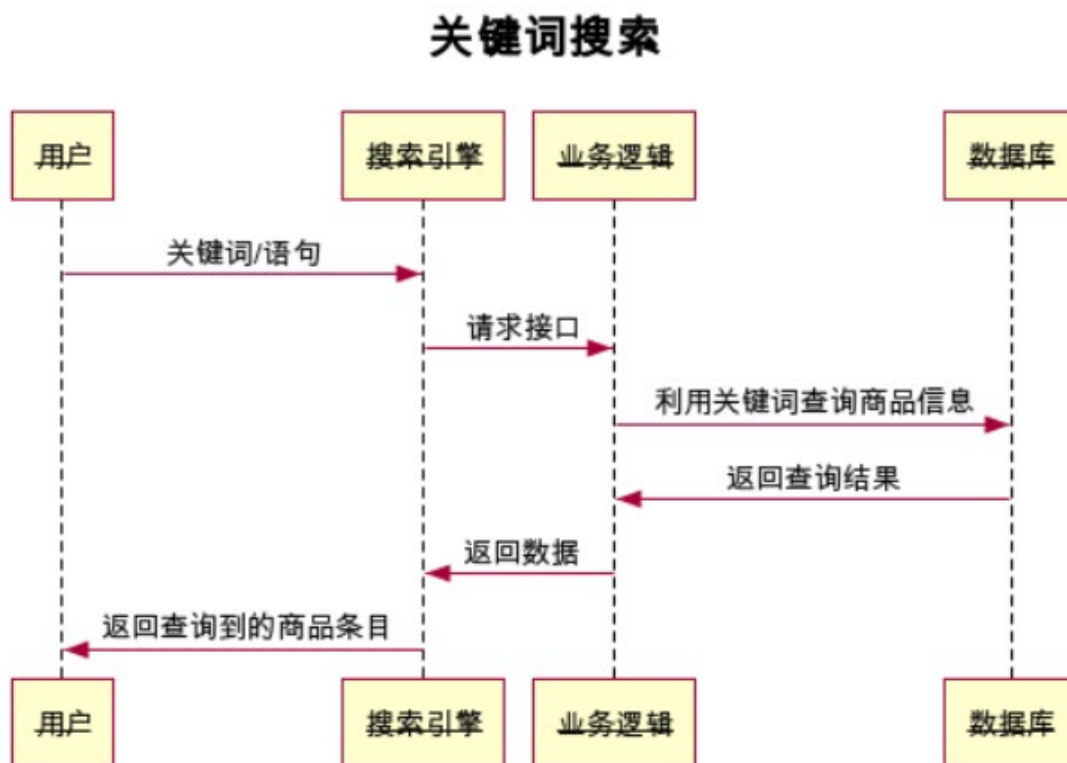


图 4-1-1 关键词搜索时序图

4.1.2 关键词过滤

表 4-1-2 关键词过滤用例描述

| 用例编号 | SR-GS-02 | 用例名称 | 关键词过滤 |
|-------|---|--------|-----------|
| 创建人 | 毕予然 | 最后修改人 | 毕予然 |
| 创建日期 | 2021/7/10 | 最后修改日期 | 2021/7/10 |
| 角色 | 用户 | 需求来源 | 用户代表 |
| 主要参与者 | 用户 | | |
| 描述 | 用户在搜索界面的文本框中输入带有[]或者[]的关键词或语句后，点击搜索按钮可以进行搜索 | | |
| 前置条件 | 1. 系统正常运行 2. 用户可正常打开相关页面 | | |
| 触发器 | 用户需要进入搜索页面 | | |
| 后置条件 | 用户搜索记录写入网站日志 | | |
| 输入信息 | 搜索内容 | | |
| 主干过程 | 1. 用户进入搜索页面 2. 用户输入带有[]或者[]的关键词或语句 3. 用户点击搜索按钮 4. 搜索成功后跳转至结果显示界面 | | |
| 分支过程 | 无 | | |
| 异常 | 1. 用户输入危险信息（发生于主干过程 2） 处理：系统提示“查询错误，请重新输入” 系统结束用例 2. 搜索后页面出现异常（发生于主干过程 4） 处理：根据网站的联系方式联系系统管理员 系统结束用例 | | |
| 假设 | 无异常发生 | | |
| 输入 | 关键词 | | |
| 输出 | 搜索结果显示页面 | | |
| 包括用例 | 搜索过滤、搜索交集 | | |
| 优先级 | 中 | | |
| 使用频率 | 低 | | |
| 备注与问题 | 无 | | |

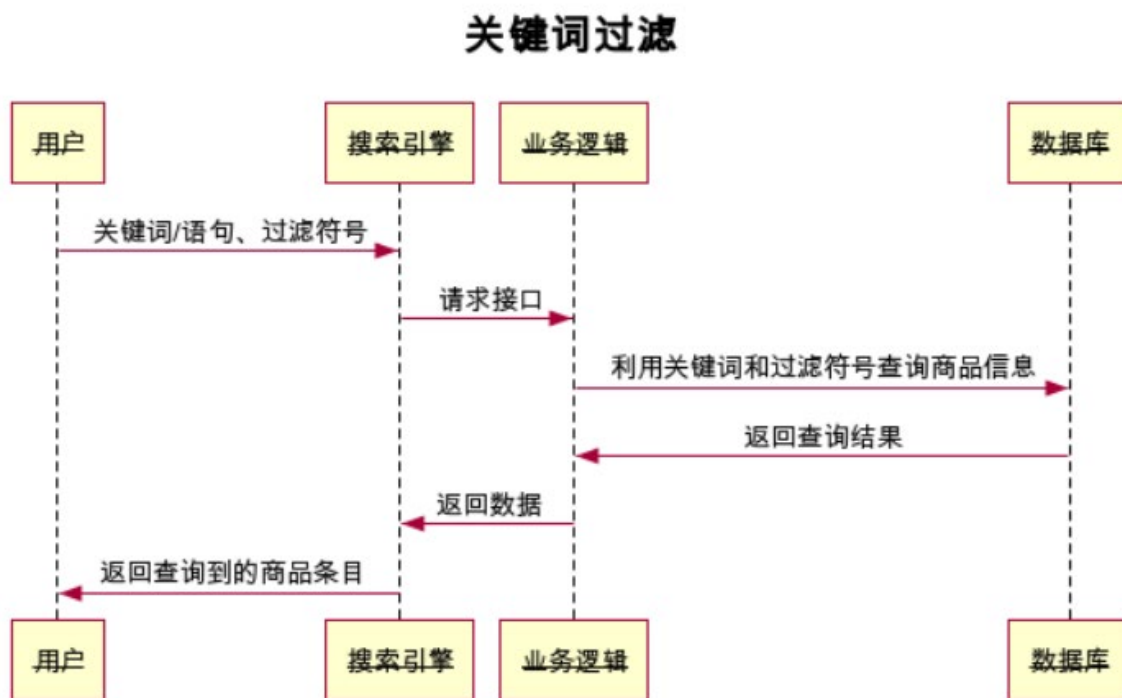


图 4-1-2 关键词过滤

4.1.3 拼音搜索

表 4-1-3 拼音搜索用例描述

| 用例编号 | SR-GS-03 | 用例名称 | 拼音搜索 |
|-------|---|--------|-----------|
| 创建人 | 毕予然 | 最后修改人 | 毕予然 |
| 创建日期 | 2021/7/10 | 最后修改日期 | 2021/7/10 |
| 角色 | 用户 | 需求来源 | 用户代表 |
| 主要参与者 | 用户 | | |
| 描述 | 用户在搜索界面的文本框中输入拼音后，点击搜索按钮可以进行搜索 | | |
| 前置条件 | 1. 系统正常运行 2. 用户可正常打开相关页面 | | |
| 触发器 | 用户需要进入搜索页面 | | |
| 后置条件 | 用户搜索记录写入网站日志 | | |
| 输入信息 | 搜索内容 | | |
| 主干过程 | 1. 用户进入搜索页面 2. 用户输入拼音 3. 用户点击搜索按钮 4. 搜索成功后跳转至结果显示界面 | | |
| 分支过程 | 无 | | |
| 异常 | 1. 用户输入危险信息（发生于主干过程 2） 处理：系统提示“查询错误，请重新输入” 系统结束用例 2. 搜索后页面出现异常（发生于主干过程 4） 处理：根据网站的联系方式联系系统管理员 系统结束用例 | | |
| 假设 | 无异常发生 | | |
| 输入 | 拼音 | | |
| 输出 | 搜索结果显示页面 | | |
| 包括用例 | 无 | | |
| 优先级 | 高 | | |
| 使用频率 | 低 | | |
| 备注与问题 | 无 | | |

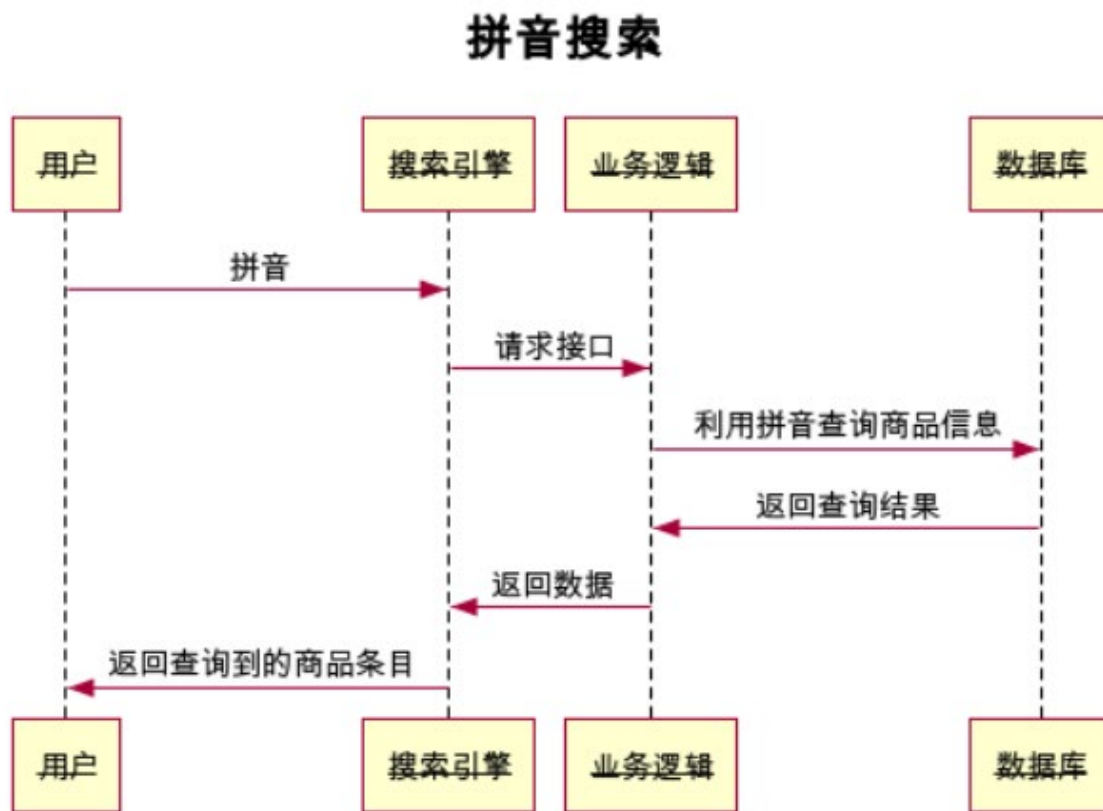


图 4-1-3 拼音搜索时序图

4.1.4 结果显示

表 4-1-4 结果显示用例描述

| 用例编号 | SR-GS-04 | 用例名称 | 结果显示 |
|-------|---|--------|-----------|
| 创建人 | 毕予然 | 最后修改人 | 毕予然 |
| 创建日期 | 2021/7/10 | 最后修改日期 | 2021/7/10 |
| 角色 | 用户 | 需求来源 | 用户代表 |
| 主要参与者 | 用户 | | |
| 描述 | 用户完成搜索准备，点击搜索按钮后，系统可以显示搜索结果 | | |
| 前置条件 | 1. 系统正常运行 2. 用户可正常打开相关页面 | | |
| 触发器 | 用户需要进入搜索页面 | | |
| 后置条件 | 用户搜索记录写入网站日志 | | |
| 输入信息 | 搜索内容 | | |
| 主干过程 | 1. 用户完成搜索输入 2. 用户点击搜索按钮 3. 结果页面按照默认相关度排序，显示搜索结果卡片列表 | | |
| 分支过程 | 无 | | |
| 异常 | 1. 搜索后页面出现异常（发生于主干过程 3） 处理：根据网站的联系方式联系系统管理员 系统结束用例 | | |
| 假设 | 无异常发生 | | |
| 输入 | 关键词/语句 | | |
| 输出 | 搜索结果显示页面 | | |
| 包括用例 | 无 | | |
| 优先级 | 高 | | |
| 使用频率 | 高 | | |
| 备注与问题 | 无 | | |

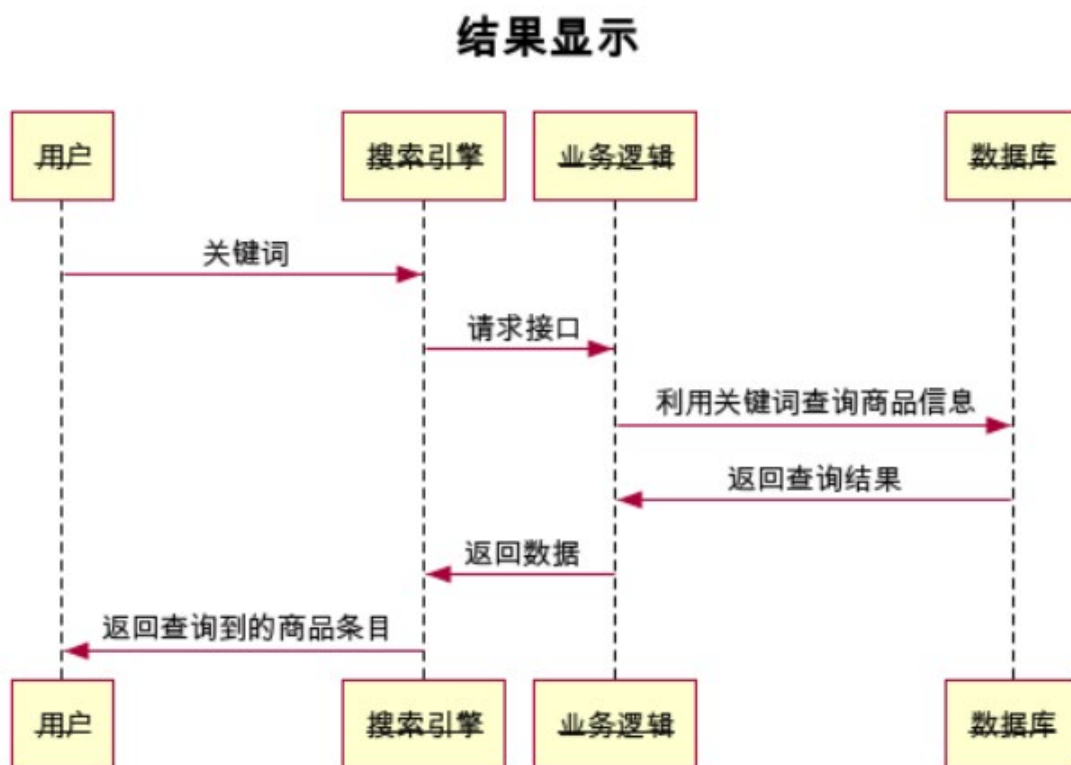


图 4-1-4 结果显示

4.1.5 图片、视频显示

表 4-1-4 图片、视频显示用例描述

| 用例编号 | SR-GS-05 | 用例名称 | 图片、视频显示 |
|-------|--|--------|-----------|
| 创建人 | 毕予然 | 最后修改人 | 毕予然 |
| 创建日期 | 2021/7/10 | 最后修改日期 | 2021/7/10 |
| 角色 | 用户 | 需求来源 | 用户代表 |
| 主要参与者 | 用户 | | |
| 描述 | 用户进入搜索后，点击选中商品的图片或视频 | | |
| 前置条件 | 1. 系统正常运行 2. 用户可正常打开相关页面 | | |
| 触发器 | 用户需要搜索完毕 | | |
| 后置条件 | 用户搜索记录写入网站日志 | | |
| 输入信息 | 搜索内容 | | |
| 主干过程 | 1. 用户完成搜索过程 2. 用户点击选中商品的图片、视频 3. 显示图片、视频 | | |
| 分支过程 | 无 | | |
| 异常 | 1. 搜索后页面出现异常（发生于主干过程 1） 处理：根据网站的联系方式联系系统管理员 系统结束用例 | | |
| 假设 | 无异常发生 | | |
| 输入 | 无 | | |
| 输出 | 图片、视频显示页面 | | |
| 包括用例 | 无 | | |
| 优先级 | 中 | | |
| 使用频率 | 中 | | |
| 备注与问题 | 无 | | |

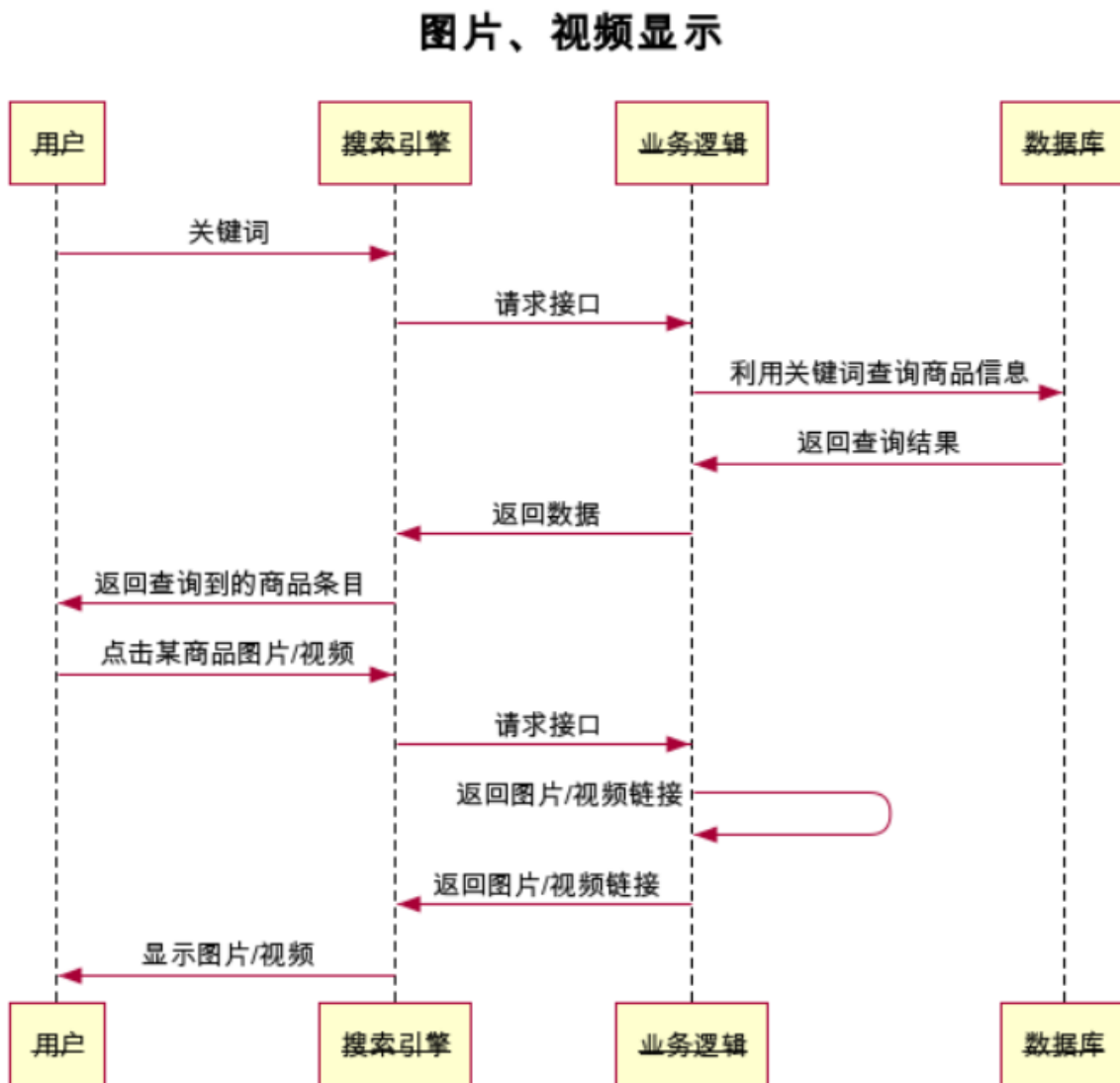


图 4-1-5 图片、视频显示

4.1.6 结果相关度排序

表 4-1-6 结果相关度排序用例描述

| 用例编号 | SR-GS-06 | 用例名称 | 结果相关度排序 |
|-------|---|--------|-----------|
| 创建人 | 毕予然 | 最后修改人 | 毕予然 |
| 创建日期 | 2021/7/10 | 最后修改日期 | 2021/7/10 |
| 角色 | 用户 | 需求来源 | 用户代表 |
| 主要参与者 | 用户 | | |
| 描述 | 系统默认按照相关度排序显示搜索结果；用户可以点击相关度按钮，将搜索结果按照相关度排序 | | |
| 前置条件 | 1. 系统正常运行 2. 用户可正常打开相关页面 3. 搜索结果正常 | | |
| 触发器 | 用户完成搜索 | | |
| 后置条件 | 用户搜索记录写入网站日志 | | |
| 输入信息 | 搜索内容 | | |
| 主干过程 | 1. 用户完成搜索输入 2. 用户点击搜索按钮 3. 用户点击按照相关度排序 4. 结果页面显示排序后的搜索结果卡片列表 | | |
| 分支过程 | 无 | | |
| 异常 | 1. 搜索后页面出现异常（发生于主干过程 3） 处理：根据网站的联系方式联系系统管理员 系统结束用例 | | |
| 假设 | 无异常发生 | | |
| 输入 | 无 | | |
| 输出 | 按照相关度排序的搜索结果显示页面 | | |
| 包括用例 | 无 | | |
| 优先级 | 中 | | |
| 使用频率 | 高 | | |
| 备注与问题 | 无 | | |

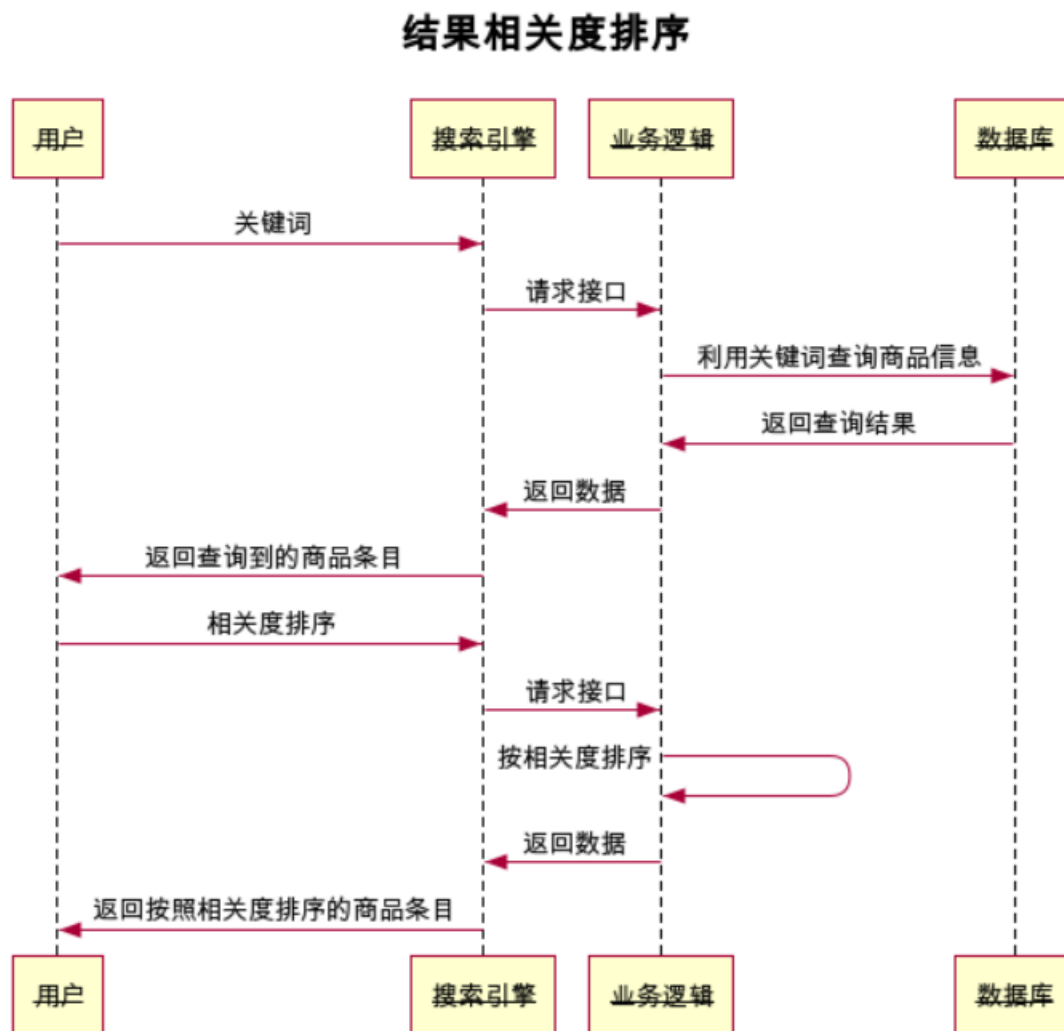


图 4-1-6 结果相关度排序

4.1.7 结果销量排序

表 4-1-7 结果销量排序用例描述

| 用例编号 | SR-GS-07 | 用例名称 | 结果销量排序 |
|-------|--|--------|-----------|
| 创建人 | 毕予然 | 最后修改人 | 毕予然 |
| 创建日期 | 2021/7/10 | 最后修改日期 | 2021/7/10 |
| 角色 | 用户 | 需求来源 | 用户代表 |
| 主要参与者 | 用户 | | |
| 描述 | 用户可以点击销量按钮，将搜索结果按照销量排序 | | |
| 前置条件 | 1. 系统正常运行 2. 用户可正常打开相关页面 3. 搜索结果正常 | | |
| 触发器 | 用户完成搜索 | | |
| 后置条件 | 用户搜索记录写入网站日志 | | |
| 输入信息 | 搜索内容 | | |
| 主干过程 | 1. 用户完成搜索输入 2. 用户点击搜索按钮 3. 用户点击按照销量排序 4. 结果页面显示排序后的搜索结果卡片列表 | | |
| 分支过程 | 无 | | |
| 异常 | 1. 搜索后页面出现异常（发生于主干过程 3） 处理：根据网站的联系方式联系系统管理员 系统结束用例 | | |
| 假设 | 无异常发生 | | |
| 输入 | 无 | | |
| 输出 | 按照销量排序的搜索结果显示页面 | | |
| 包括用例 | 无 | | |
| 优先级 | 中 | | |
| 使用频率 | 中 | | |
| 备注与问题 | 无 | | |

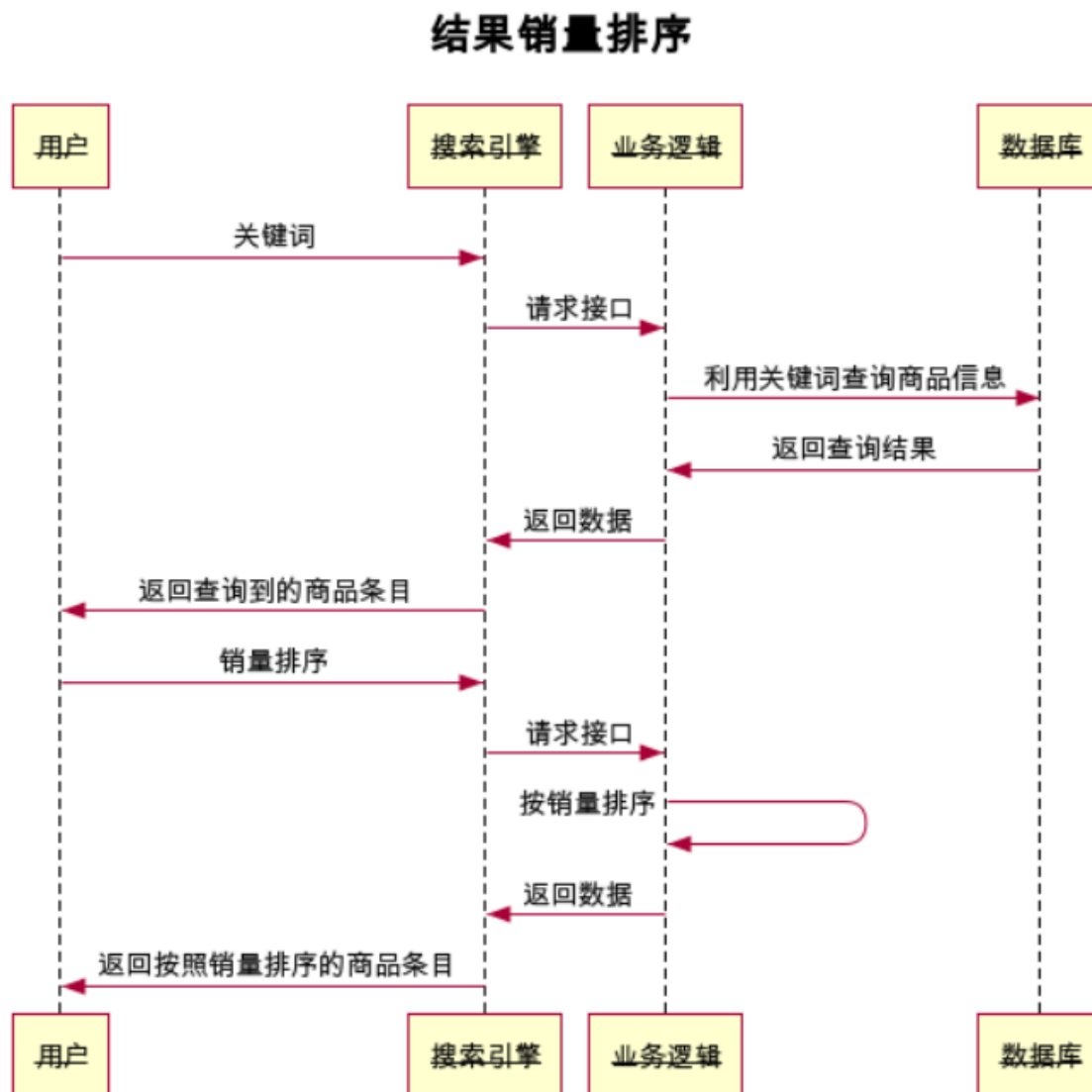


图 4-1-7 结果销量排序

4.1.8 结果价格排序

表 4-1-6 结果价格排序用例描述

| 用例编号 | SR-GS-08 | 用例名称 | 结果价格排序 |
|-------|--|--------|-----------|
| 创建人 | 毕予然 | 最后修改人 | 毕予然 |
| 创建日期 | 2021/7/10 | 最后修改日期 | 2021/7/10 |
| 角色 | 用户 | 需求来源 | 用户代表 |
| 主要参与者 | 用户 | | |
| 描述 | 用户可以点击价格按钮，将搜索结果按照价格排序 | | |
| 前置条件 | 1. 系统正常运行 2. 用户可正常打开相关页面 3. 搜索结果正常 | | |
| 触发器 | 用户完成搜索 | | |
| 后置条件 | 用户搜索记录写入网站日志 | | |
| 输入信息 | 搜索内容 | | |
| 主干过程 | 1. 用户完成搜索输入 2. 用户点击搜索按钮 3. 用户点击按照价格排序 4. 结果页面显示排序后的搜索结果卡片列表 | | |
| 分支过程 | 无 | | |
| 异常 | 1. 搜索后页面出现异常（发生于主干过程 3） 处理：根据网站的联系方式联系系统管理员 系统结束用例 | | |
| 假设 | 无异常发生 | | |
| 输入 | 无 | | |
| 输出 | 按照价格排序的搜索结果显示页面 | | |
| 包括用例 | 无 | | |
| 优先级 | 中 | | |
| 使用频率 | 中 | | |
| 备注与问题 | 无 | | |

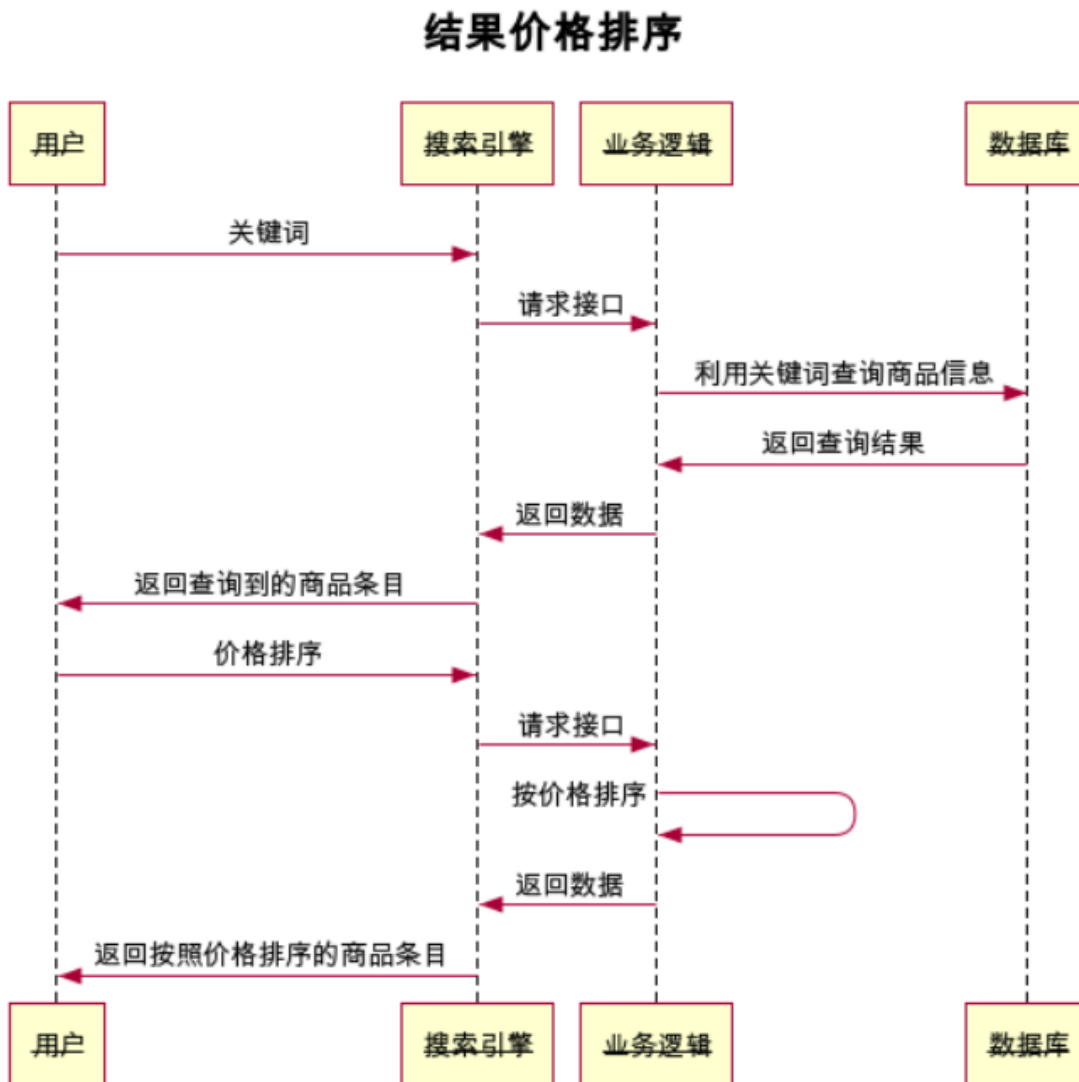


图 4-1-8 结果价格排序

4.1.9 同类商品推荐

表 4-1-9 同类商品推荐用例描述

| 用例编号 | SR-GS-09 | 用例名称 | 结果价格排序 |
|-------|---|--------|-----------|
| 创建人 | 毕予然 | 最后修改人 | 毕予然 |
| 创建日期 | 2021/7/10 | 最后修改日期 | 2021/7/10 |
| 角色 | 用户 | 需求来源 | 用户代表 |
| 主要参与者 | 用户 | | |
| 描述 | 用户完成搜索后，系统自动进行同类商品搜索，并在页面上展示 | | |
| 前置条件 | 1. 系统正常运行 2. 用户可正常打开相关页面 | | |
| 触发器 | 用户完成搜索 | | |
| 后置条件 | 用户搜索记录写入网站日志 | | |
| 输入信息 | 搜索内容 | | |
| 主干过程 | 1. 用户进入网站搜索页 2. 用户输入关键词 3. 用户点击搜索按钮 4. 搜索成功后跳转至结果显示界面 | | |
| 分支过程 | 无 | | |
| 异常 | 1. 用户输入危险信息（发生于主干过程 2） 处理：系统提示“查询错误，请重新输入” 系统结束用例 2. 搜索后页面出现异常（发生于主干过程 4） 处理：根据网站的联系方式联系系统管理员 系统结束用例 | | |
| 假设 | 无异常发生 | | |
| 输入 | 无 | | |
| 输出 | 显示同类商品推荐页面 | | |
| 包括用例 | 搜索过滤、搜索并集、搜索交集 | | |
| 优先级 | 高 | | |
| 使用频率 | 高 | | |
| 备注与问题 | 无 | | |

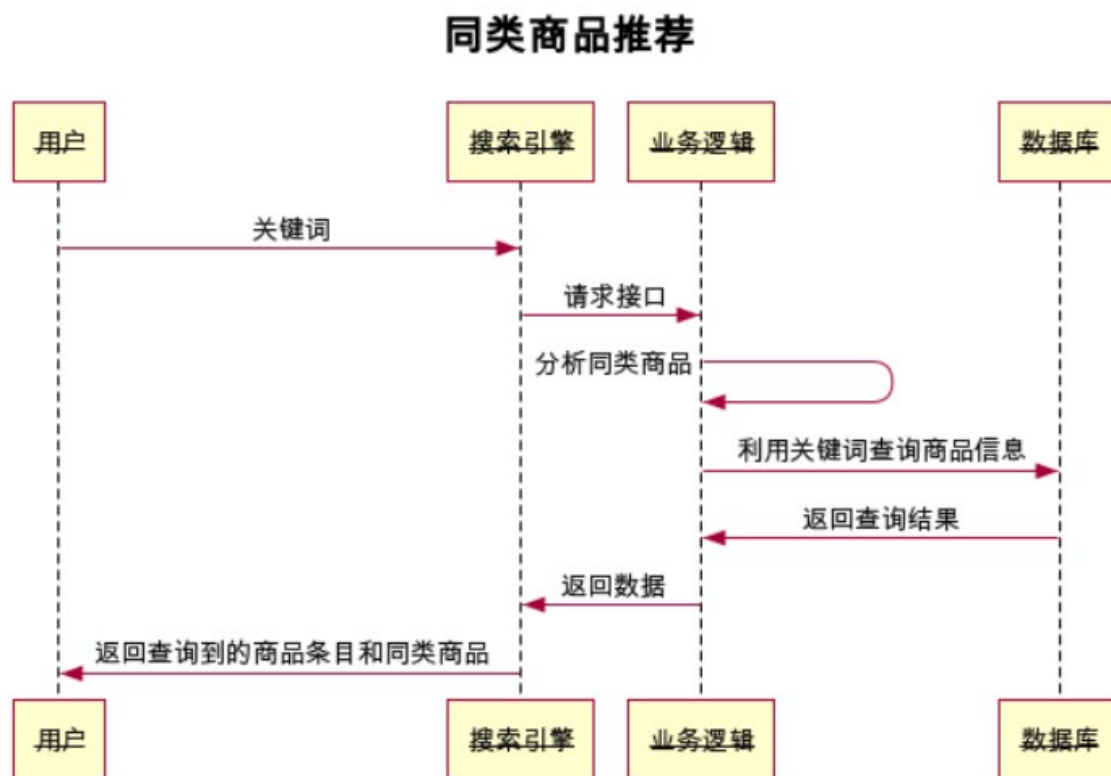


图 4-1-9 同类商品推荐

4.1.10 搜索热榜

表 4-1-10 搜索热榜用例描述

| 用例编号 | SR-GS-10 | 用例名称 | 搜索热榜 |
|-------|--|--------|-----------|
| 创建人 | 毕予然 | 最后修改人 | 毕予然 |
| 创建日期 | 2021/7/10 | 最后修改日期 | 2021/7/10 |
| 角色 | 用户 | 需求来源 | 用户代表 |
| 主要参与者 | 用户 | | |
| 描述 | 系统根据网站日志自动形成搜索热榜，用户点击热榜可将被搜索次数最多的商品依次排序在页面上展示 | | |
| 前置条件 | 1. 系统正常运行 2. 用户可正常打开相关页面 | | |
| 触发器 | 用户点击热榜 | | |
| 后置条件 | 无 | | |
| 输入信息 | 无 | | |
| 主干过程 | 1. 用户进入网站搜索页 2. 用户点击热榜按钮 3. 搜索成功后跳转至搜索热榜界面 | | |
| 分支过程 | 无 | | |
| 异常 | 1. 点击后页面出现异常（发生于主干过程 3） 处理：根据网站的联系方式联系系统管理员 系统结束用例 | | |
| 假设 | 无异常发生 | | |
| 输入 | 无 | | |
| 输出 | 显示搜索热榜页面 | | |
| 包括用例 | 无 | | |
| 优先级 | 中 | | |
| 使用频率 | 低 | | |
| 备注与问题 | 无 | | |

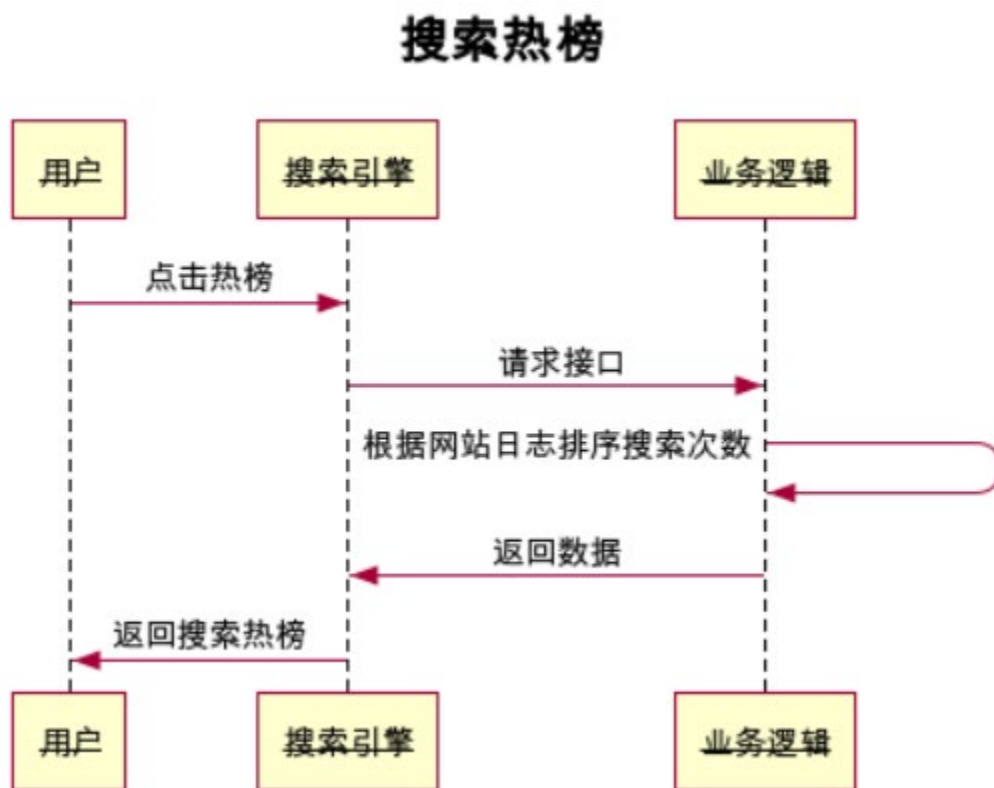


图 4-1-10 搜索热榜推荐

4.2 用户进行高级搜索

4.2.1 关键词并集

表 4-2-1 关键词并集用例描述

| 用例编号 | SR-AS-01 | 用例名称 | 关键词并集 |
|-------|---|--------|-----------|
| 创建人 | 程梦烨 | 最后修改人 | 程梦烨 |
| 创建日期 | 2021/7/10 | 最后修改日期 | 2021/7/10 |
| 角色 | 用户 | 需求来源 | 用户代表 |
| 主要参与者 | 用户 | | |
| 描述 | 用户在高级搜索界面的文本框中选择关键词并集搜索，点击搜索按钮可以进行搜索 | | |
| 前置条件 | 3. 系统正常运行 4. 用户可正常打开相关页面 | | |
| 触发器 | 用户需要进入搜索页面 | | |
| 后置条件 | 用户搜索记录写入网站日志 | | |
| 输入信息 | 搜索内容 | | |
| 主干过程 | 1. 用户进入搜索页面 2. 用户输入关键词，选择关键词搜索并集 3. 用户点击搜索按钮 4. 搜索成功后跳转至结果显示界面 | | |
| 分支过程 | 无 | | |
| 异常 | 3. 用户输入危险信息（发生于主干过程 2） 处理：系统提示“查询错误，请重新输入” 系统结束用例 4. 搜索后页面出现异常（发生于主干过程 4） 处理：根据网站的联系方式联系系统管理员 系统结束用例 | | |
| 假设 | 无异常发生 | | |
| 输入 | 关键词、搜索方式 | | |
| 输出 | 搜索结果显示页面 | | |
| 包括用例 | 搜索过滤、搜索交集 | | |
| 优先级 | 中 | | |
| 使用频率 | 低 | | |
| 备注与问题 | 无 | | |

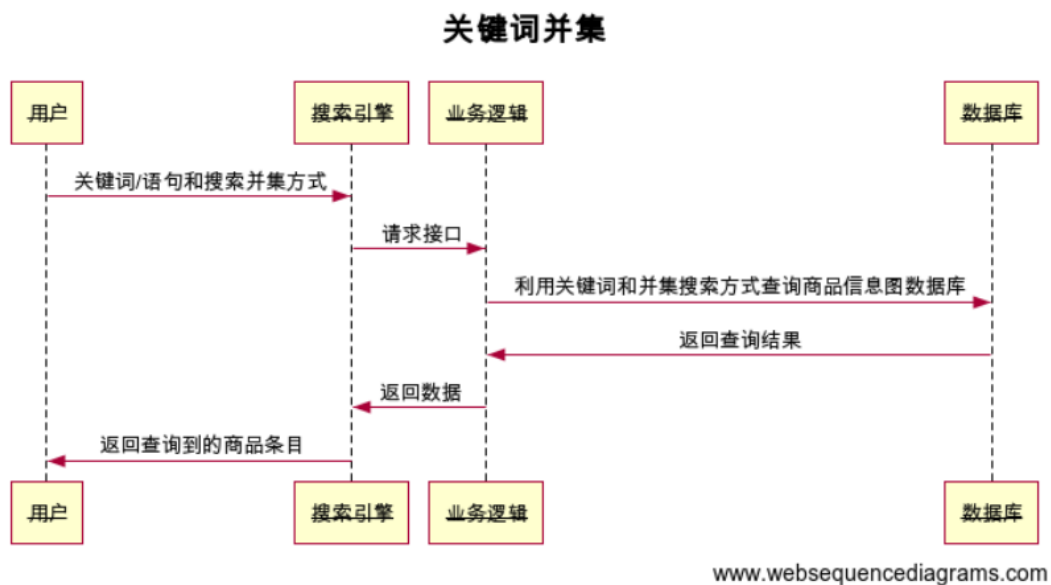


图 4-2-1 关键词并集时序图

4.2.2 关键词交集

表 4-2-2 关键词交集用例描述

| 用例编号 | SR-AS-01 | 用例名称 | 关键词交集 |
|-------|---|--------|-----------|
| 创建人 | 程梦烨 | 最后修改人 | 程梦烨 |
| 创建日期 | 2021/7/10 | 最后修改日期 | 2021/7/10 |
| 角色 | 用户 | 需求来源 | 用户代表 |
| 主要参与者 | 用户 | | |
| 描述 | 用户在高级搜索界面的文本框中选择关键词交集搜索，点击搜索按钮可以进行搜索 | | |
| 前置条件 | 1. 系统正常运行 2. 用户可正常打开相关页面 | | |
| 触发器 | 用户需要进入搜索页面 | | |
| 后置条件 | 用户搜索记录写入网站日志 | | |
| 输入信息 | 搜索内容 | | |
| 主干过程 | 1. 用户进入搜索页面 2. 用户输入关键词，选择关键词搜索交集 3. 用户点击搜索按钮 4. 搜索成功后跳转至结果显示界面 | | |
| 分支过程 | 无 | | |
| 异常 | 1. 用户输入危险信息（发生于主干过程 2） 处理：系统提示“查询错误，请重新输入” 系统结束用例 2. 搜索后页面出现异常（发生于主干过程 4） 处理：根据网站的联系方式联系系统管理员 系统结束用例 | | |
| 假设 | 无异常发生 | | |
| 输入 | 关键词、搜索方式 | | |
| 输出 | 搜索结果显示页面 | | |
| 包括用例 | 搜索过滤、搜索并集 | | |
| 优先级 | 中 | | |
| 使用频率 | 低 | | |
| 备注与问题 | 无 | | |

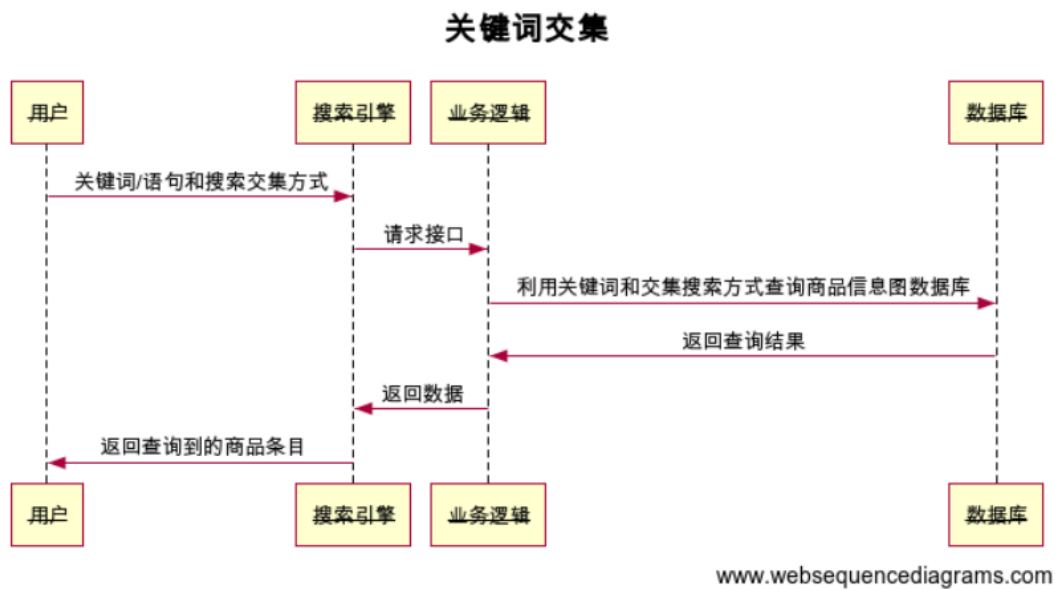


图 4-2-2 关键词交集时序图

4.2.3 优先补全

表 4-2-3 优先补全用例描述

| 用例编号 | SR-AS-01 | 用例名称 | 优先补全 |
|-------|---|--------|-----------|
| 创建人 | 程梦烨 | 最后修改人 | 程梦烨 |
| 创建日期 | 2021/7/10 | 最后修改日期 | 2021/7/10 |
| 角色 | 用户 | 需求来源 | 用户代表 |
| 主要参与者 | 用户 | | |
| 描述 | 用户在搜索文本框输入内容时，系统会优先补全已搜索过的内容 | | |
| 前置条件 | 1. 系统正常运行 2. 用户可正常打开相关页面 | | |
| 触发器 | 用户需要进入搜索页面 | | |
| 后置条件 | 用户搜索记录写入网站日志 | | |
| 输入信息 | 搜索内容 | | |
| 主干过程 | 1. 用户进入搜索页面 2. 用户输入关键词，系统进行优先补全 3. 用户点击搜索按钮 4. 搜索成功后跳转至结果显示界面 | | |
| 分支过程 | 无 | | |
| 异常 | 1. 用户输入危险信息（发生于主干过程 2） 处理：系统提示“查询错误，请重新输入” 系统结束用例 2. 搜索后页面出现异常（发生于主干过程 4） 处理：根据网站的联系方式联系系统管理员 系统结束用例 | | |
| 假设 | 无异常发生 | | |
| 输入 | 关键词、语句 | | |
| 输出 | 搜索结果显示页面 | | |
| 包括用例 | 搜索过滤、搜索交集、搜索并集 | | |
| 优先级 | 中 | | |
| 使用频率 | 中 | | |
| 备注与问题 | 无 | | |

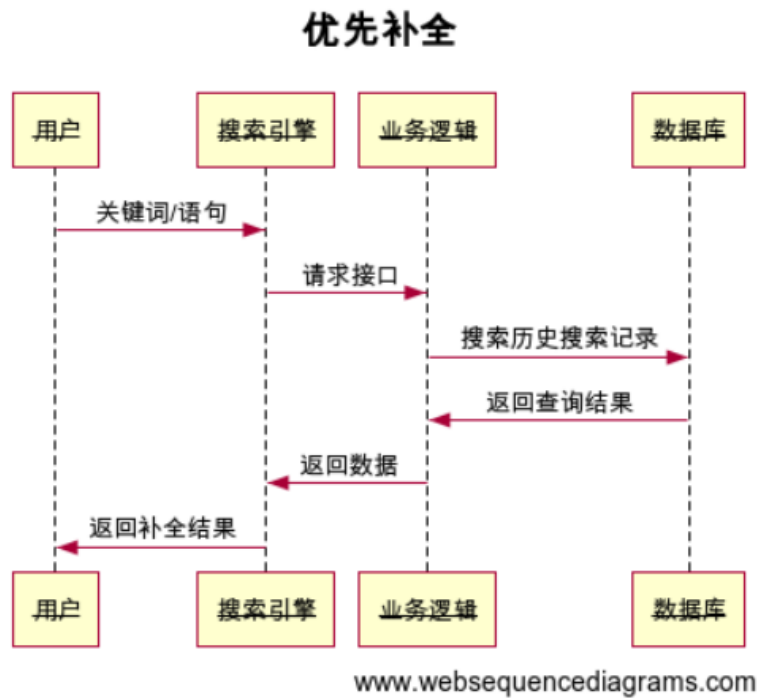


图 4-2-3 优先补全时序图

4.2.4 搜索提示

表 4-2-4 搜索提示用例描述

| 用例编号 | SR-AS-01 | 用例名称 | 搜索提示 |
|-------|---|--------|-----------|
| 创建人 | 程梦烨 | 最后修改人 | 程梦烨 |
| 创建日期 | 2021/7/10 | 最后修改日期 | 2021/7/10 |
| 角色 | 用户 | 需求来源 | 用户代表 |
| 主要参与者 | 用户 | | |
| 描述 | 用户在搜索文本框输入内容时，系统显示最近两条搜索记录 | | |
| 前置条件 | 1. 系统正常运行 2. 用户可正常打开相关页面 | | |
| 触发器 | 用户需要进入搜索页面 | | |
| 后置条件 | 用户搜索记录写入网站日志 | | |
| 输入信息 | 搜索内容 | | |
| 主干过程 | 1. 用户进入搜索页面 2. 用户输入关键词，系统提示最近两条搜索记录 3. 用户点击搜索按钮 4. 搜索成功后跳转至结果显示界面 | | |
| 分支过程 | 无 | | |
| 异常 | 1. 用户输入危险信息（发生于主干过程 2） 处理：系统提示“查询错误，请重新输入” 系统结束用例 2. 搜索后页面出现异常（发生于主干过程 4） 处理：根据网站的联系方式联系系统管理员 系统结束用例 | | |
| 假设 | 无异常发生 | | |
| 输入 | 关键词、语句 | | |
| 输出 | 搜索结果显示页面 | | |
| 包括用例 | 搜索过滤、搜索交集、搜索并集 | | |
| 优先级 | 中 | | |
| 使用频率 | 中 | | |
| 备注与问题 | 无 | | |

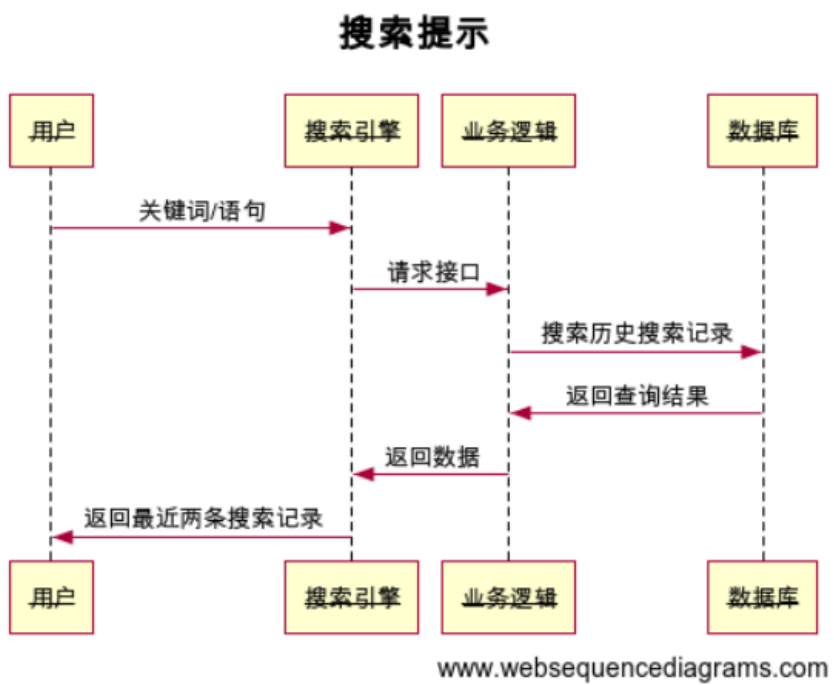


图 4-2-4 搜索提示时序图

4.2.5 关键词条件筛选

表 4-2-5 关键词条件筛选用例描述

| 用例编号 | SR-AS-01 | 用例名称 | 关键词条件筛选 |
|-------|---|--------|-----------|
| 创建人 | 程梦烨 | 最后修改人 | 程梦烨 |
| 创建日期 | 2021/7/10 | 最后修改日期 | 2021/7/10 |
| 角色 | 用户 | 需求来源 | 用户代表 |
| 主要参与者 | 用户 | | |
| 描述 | 用户在高级搜索界面的文本框中选择关键词条件筛选，点击搜索按钮可以进行搜索 | | |
| 前置条件 | 1. 系统正常运行 2. 用户可正常打开相关页面 | | |
| 触发器 | 用户需要进入搜索页面 | | |
| 后置条件 | 用户搜索记录写入网站日志 | | |
| 输入信息 | 搜索内容 | | |
| 主干过程 | 1. 用户进入搜索页面 2. 用户输入关键词，选择关键词条件筛选 3. 用户点击搜索按钮 4. 搜索成功后跳转至结果显示界面 | | |
| 分支过程 | 无 | | |
| 异常 | 1. 用户输入危险信息（发生于主干过程 2） 处理：系统提示“查询错误，请重新输入” 系统结束用例 2. 搜索后页面出现异常（发生于主干过程 4） 处理：根据网站的联系方式联系系统管理员 系统结束用例 | | |
| 假设 | 无异常发生 | | |
| 输入 | 关键词、语句 | | |
| 输出 | 搜索结果显示页面 | | |
| 包括用例 | 搜索过滤、搜索交集、搜索并集 | | |
| 优先级 | 中 | | |
| 使用频率 | 中 | | |
| 备注与问题 | 无 | | |

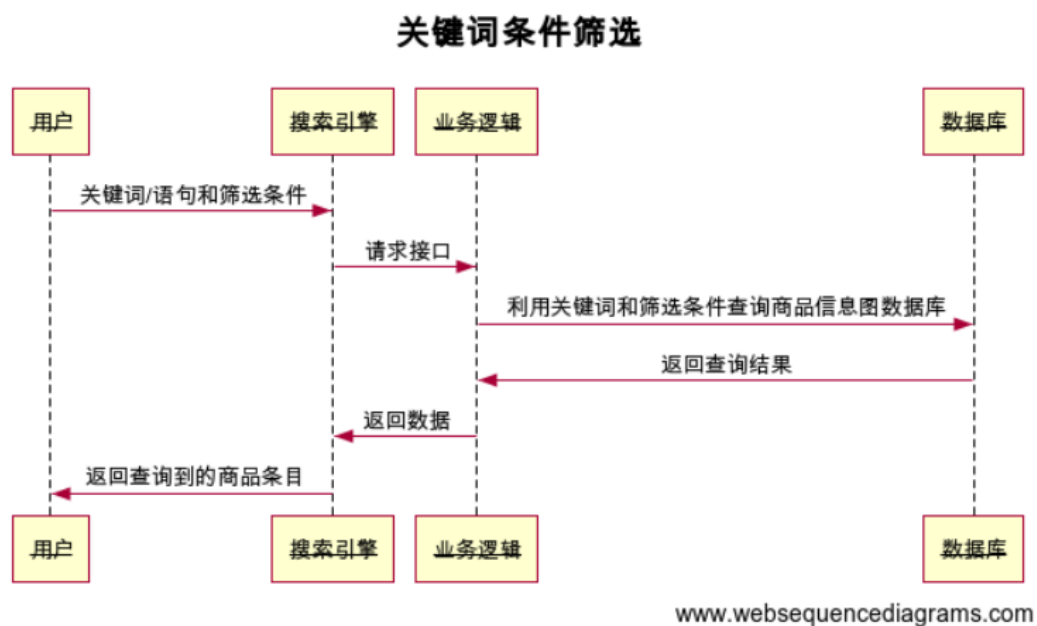


图 4-2-5 关键词条件筛选时序图

4.2.6 关键词模糊搜索

表 4-2-6 关键词模糊搜索用例描述

| 用例编号 | SR-AS-01 | 用例名称 | 关键词模糊搜索 |
|-------|---|--------|-----------|
| 创建人 | 程梦烨 | 最后修改人 | 程梦烨 |
| 创建日期 | 2021/7/10 | 最后修改日期 | 2021/7/10 |
| 角色 | 用户 | 需求来源 | 用户代表 |
| 主要参与者 | 用户 | | |
| 描述 | 用户在高级搜索界面的文本框中输入带有*或?的关键词或语句后，点击搜索按钮可以进行搜索 | | |
| 前置条件 | 1. 系统正常运行 2. 用户可正常打开相关页面 | | |
| 触发器 | 用户需要进入搜索页面 | | |
| 后置条件 | 用户搜索记录写入网站日志 | | |
| 输入信息 | 搜索内容 | | |
| 主干过程 | 1. 用户进入搜索页面 2. 用户输入关键词，选择关键词模糊并集 3. 用户点击搜索按钮 4. 搜索成功后跳转至结果显示界面 | | |
| 分支过程 | 无 | | |
| 异常 | 1. 用户输入危险信息（发生于主干过程 2） 处理：系统提示“查询错误，请重新输入” 系统结束用例 2. 搜索后页面出现异常（发生于主干过程 4） 处理：根据网站的联系方式联系系统管理员 系统结束用例 | | |
| 假设 | 无异常发生 | | |
| 输入 | 关键词、语句 | | |
| 输出 | 搜索结果显示页面 | | |
| 包括用例 | 搜索过滤、搜索交集、搜索并集 | | |
| 优先级 | 中 | | |
| 使用频率 | 低 | | |
| 备注与问题 | 无 | | |

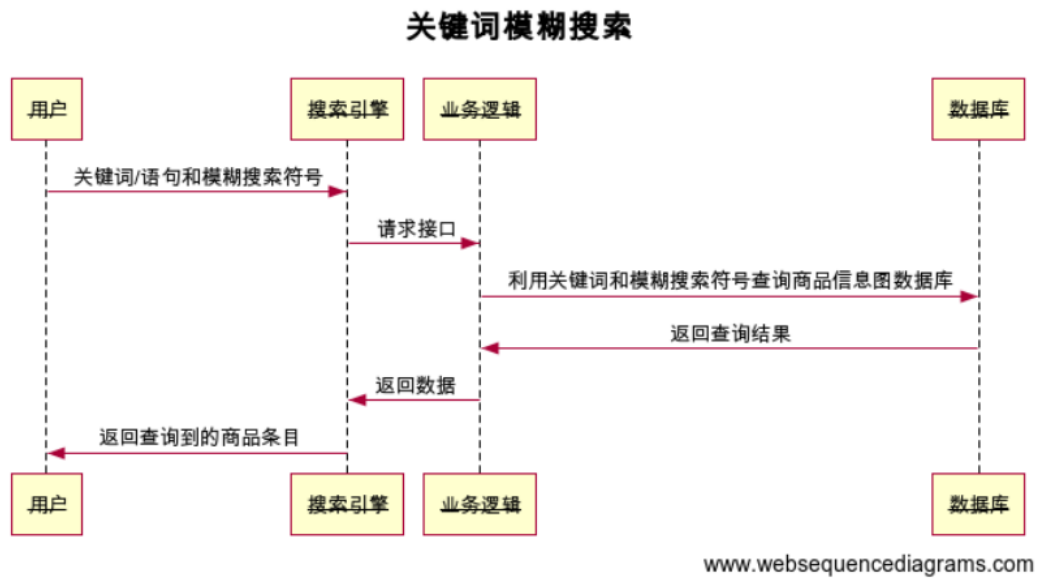


图 4-2-6 关键词模糊搜索时序图

4.3 用户管理关键词

4.3.1 用户查看关键词

表 4-3-1 用户查看关键词用例描述

| 用例编号 | UD-KW-01 | 用例名称 | 用户增加关键词 |
|-------|--|--------|------------|
| 创建人 | 沈乐明 | 最后修改人 | 沈乐明 |
| 创建日期 | 2021-07-09 | 最后修改日期 | 2021-07-09 |
| 角色 | 普通用户 | 需求来源 | 普通用户 |
| 主要参与者 | 普通用户 | | |
| 描述 | 用户在使用普通搜索功能时，可能会有些想要搜索的关键词未收录到服务器的知识图谱中，因此当用户输入了一个新的关键词后，系统会自动将该关键词加入到一个特定的列表中。而该列表中的关键词会被服务器逐个取出进行爬虫，因此会需要一定的时间，当用户需要查看某个关键词是否已经爬取并更新知识图谱完毕时，用户可以点击“查看关键词状态”进行查看。 | | |
| 前置条件 | <ul style="list-style-type: none"> 系统正常运行 普通用户可以正常打开相关页面 服务器爬虫服务端正常运行 | | |
| 触发器 | 普通用户点击了“查看关键词状态”按钮 | | |
| 后置条件 | 前端会显示一个所有关键词状态的列表，用户可以进行筛选或搜索，主要分为“爬虫爬取中”，“知识图谱构建中”和“已完成更新”这三个状态 | | |
| 输入信息 | 无 | | |
| 主干过程 | <ol style="list-style-type: none"> 普通用户在搜索界面中点击“查看关键词状态”的按钮 界面显示一个所有关键词列表，每个关键词都附带一个状态属性值，主要分为“爬虫爬取中”，“知识图谱构建中”和“已完成更新”这三个状态 | | |
| 分支过程 | 无 | | |
| 异常 | <ul style="list-style-type: none"> 未能显示或未能正确显示相应的页面（主干过程 1） 处理：联系管理员或技术人员或再次尝试 无法显示或无法正常显示关键词状态列表（主干过程 2） 处理：刷新页面或联系管理员或技术人员 | | |
| 假设 | 无异常发生 | | |
| 输入 | 无 | | |
| 输出 | <ul style="list-style-type: none"> 关键词状态列表 | | |
| 包括用例 | 无 | | |
| 优先级 | 高 | | |
| 使用频率 | 高 | | |
| 备注与问题 | 无 | | |

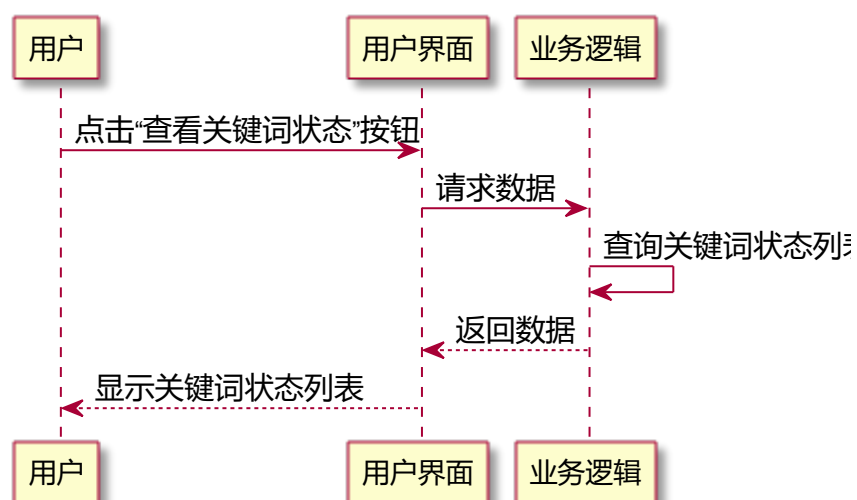


图 4-3-1 用户查看关键词时序图

4.3.2 用户增加关键词

表 4-3-2 用户增加关键词用例描述

| 用例编号 | UD-KW-02 | 用例名称 | 用户增加关键词 |
|-------|---|--------|------------|
| 创建人 | 沈乐明 | 最后修改人 | 沈乐明 |
| 创建日期 | 2021-07-09 | 最后修改日期 | 2021-07-09 |
| 角色 | 普通用户 | 需求来源 | 普通用户 |
| 主要参与者 | 普通用户 | | |
| 描述 | 用户在使用普通搜索功能时，可能会有些想要搜索的关键词未收录到服务器的知识图谱中，因此当用户输入了一个新的关键词后，系统会自动将该关键词加入到一个特定的列表中，当系统轮询到当前关键词后会启动爬虫端自动进行爬虫，更新已有的知识图谱，从而达到自我的不断更新。 | | |
| 前置条件 | <ul style="list-style-type: none"> 系统正常运行 普通用户可以正常打开相关页面 服务器爬虫服务端正常运行 | | |
| 触发器 | 普通用户输入了暂未收录进入知识图谱的关键词后，点击了搜索按钮。 | | |
| 后置条件 | <ul style="list-style-type: none"> 前端提示用户“您当前搜索的关键词未被收录，是否立刻加入知识图谱？否则我们将一天后自动收录。” 用户可以选择是否立即搜索。 | | |
| 输入信息 | <ul style="list-style-type: none"> 关键词 | | |
| 主干过程 | <ol style="list-style-type: none"> 普通用户在搜索框内输入想要搜索的关键词。 系统自动进行检测，可能会发现该关键词尚未被收录进入知识图谱。 系统界面提示用户“您当前搜索的关键词未被收录，是否立刻加入知识图谱？否则我们将一天后自动收录。” 用户进行选择。 如果用户选择“是”，则转到用例 SE-05。 如果用户选择“否”，则系统将该新关键词加入到后端等待队列中。 当该关键词从等待队列中被取出后，系统会自动通知爬虫端进行自动化的爬虫和系统知识图谱的更新。 一天后会自动更新，可以查到关键词。 | | |
| 分支过程 | <ol style="list-style-type: none"> 分支一：用户选择“是”，则转到用例 SE-05。 如果用户选择“否”，则系统将该新关键词加入到后端等待队列中。后续将会由后端自动处理。 | | |

| | |
|-------|---|
| 异常 | <ul style="list-style-type: none"> 未能显示或未能正确显示相应的页面（主干过程 1） 处理：联系管理员或技术人员或再次尝试 一天后仍然无法查询到关键词（主干过程 8） 处理：刷新页面或联系管理员或技术人员 |
| 假设 | 无异常发生 |
| 输入 | <ul style="list-style-type: none"> 关键词：key_word |
| 输出 | <ul style="list-style-type: none"> 提醒用户“您当前搜索的关键词未被收录，是否立刻加入知识图谱？否则我们将一天后自动收录。”的弹框。 更新后能进行搜索的界面。 |
| 包括用例 | 无 |
| 优先级 | 高 |
| 使用频率 | 高 |
| 备注与问题 | 无 |

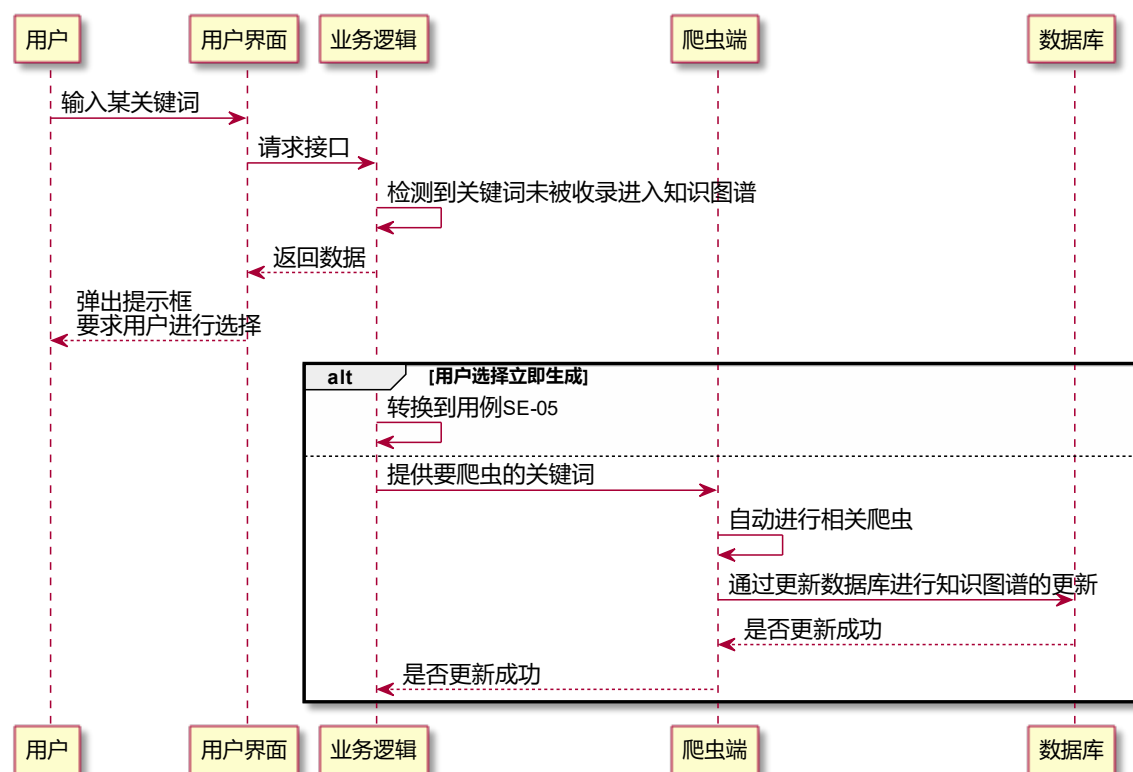


图 4-3-2 用户增加关键词时序图

4.4 用户管理知识图谱

4.4.1 用户更新知识图谱

表 4-4-1 用户更新知识图谱

| 用例编号 | UD-KM-01 | 用例名称 | 用户更新知识图谱 |
|-------|--|--------|------------|
| 创建人 | 马梓睿 | 最近修改人 | 马梓睿 |
| 创建日期 | 2021/07/10 | 最后修改日期 | 2021/07/10 |
| 角色 | 包括买家、卖家在内的所有垂直搜索引擎系统使用者 | 需求来源 | 买家、卖家等用户代表 |
| 主要参与者 | 买家、卖家 | | |
| 描述 | 用户试图更新知识图谱 | | |
| 前置条件 | 1. 系统运行正常 2. 用户使用设备运行正常 | | |
| 触发器 | 用户点击更新知识图谱按钮 | | |
| 后置条件 | 1. 日志记录本次请求 | | |
| 输入信息 | 无 | | |
| 主干过程 | 1. 用户进入网页主界面，点击头部的“关于”链接跳转到关于页面 2. 用户点击更新知识图谱 3. 后台爬取所有网站源 | | |
| 分支过程 | 无 | | |
| 异常 | 1. 网页无反应（发生于主干过程步骤 1、2） 处理：根据网站的联系方式联系系统管理员 系统结束用例 2. 爬虫模块报异常（发生于主干过程步骤 3） 处理：根据网站的联系方式联系系统管理员 系统结束用例 | | |
| 假设 | 无异常 | | |
| 输入 | 无 | | |
| 输出 | 无 | | |
| 包括用例 | 无 | | |
| 优先级 | 中 | | |
| 使用频率 | 低（预估为每天一次） | | |
| 备注 | 无 | | |

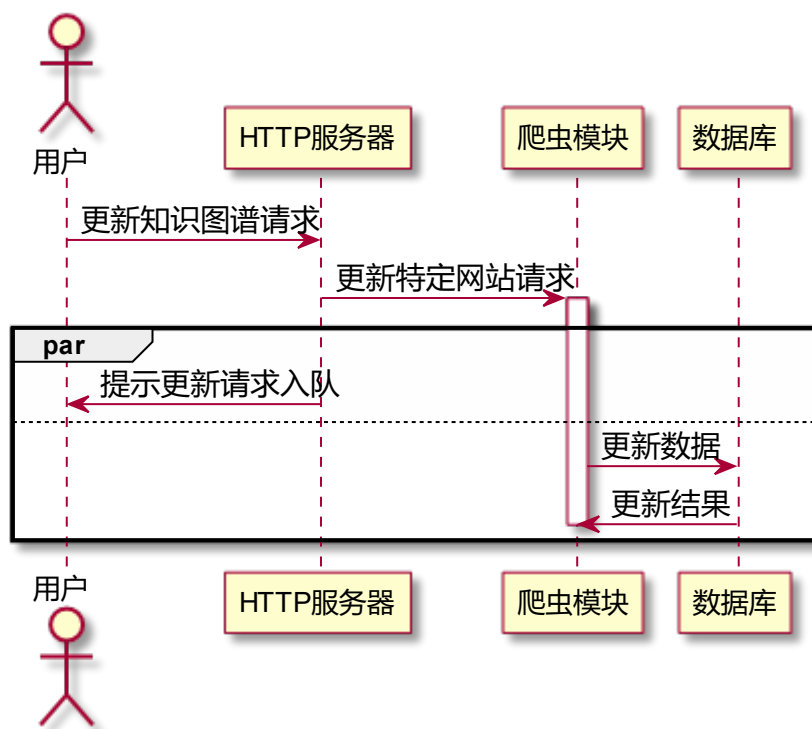


图 4-5-2 用户更新知识图谱时序图

4.5 用户管理网站源

4.5.1 用户查看爬虫网站源

表 4-5-1 用户查看爬虫网站源用例描述

| 用例编号 | UD-WB-01 | 用例名称 | 用户查看爬虫网站源 |
|-------|--|--------|------------|
| 创建人 | 李楠 | 最近修改人 | 李楠 |
| 创建日期 | 2021/07/10 | 最后修改日期 | 2021/07/10 |
| 角色 | 包括买家、卖家在内的所有垂直搜索引擎系统使用者 | 需求来源 | 买家、卖家等用户代表 |
| 主要参与者 | 买家、卖家 | | |
| 描述 | 用户试图查看网站信息源 | | |
| 前置条件 | 3. 系统运行正常 4. 用户使用设备运行正常 | | |
| 触发器 | 用户点击查看爬虫网站源按钮 | | |
| 后置条件 | 2. 日志记录本次请求 | | |
| 输入信息 | 无 | | |
| 主干过程 | 4. 用户进入网页主界面，点击头部的“关于”链接跳转到关于页面 5. 用户点击查看爬虫网站源 6. 后台返回爬虫网站源，前台显示 | | |
| 分支过程 | 无 | | |
| 异常 | 3. 网页无反应（发生于主干过程步骤 1、2） 处理：根据网站的联系方式联系系统管理员 系统结束用例 4. 爬虫模块报异常（发生于主干过程步骤 3） 处理：根据网站的联系方式联系系统管理员 系统结束用例 | | |
| 假设 | 无异常 | | |
| 输入 | 无 | | |
| 输出 | 爬虫网站源列表 | | |
| 包括用例 | 无 | | |
| 优先级 | 高 | | |
| 使用频率 | 中（预估为每日登陆人数的 0.1 倍） | | |
| 备注 | 无 | | |

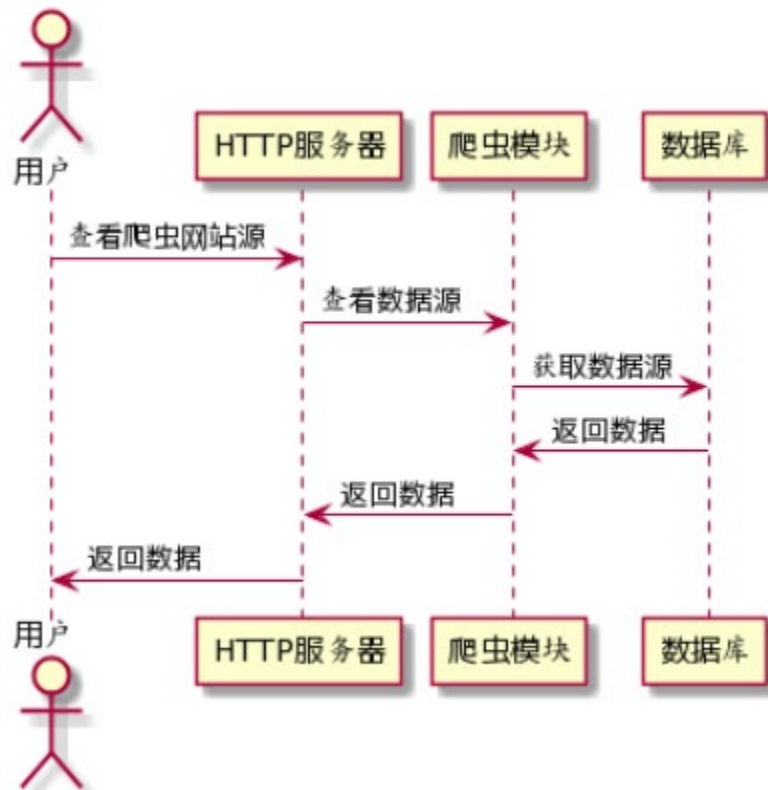


图 4-5-2 用户查看爬虫网站源时序图

4.5.2 用户更新爬虫网站源

表 4-5-2 用户更新爬虫网站源用例描述

| 用例编号 | UD-WB-02 | 用例名称 | 用户更新爬虫网站源 |
|-------|--|--------|------------|
| 创建人 | 李楠 | 最近修改人 | 李楠 |
| 创建日期 | 2021/07/10 | 最后修改日期 | 2021/07/10 |
| 角色 | 包括买家、卖家在内的所有垂直搜索引擎系统使用者 | 需求来源 | 买家、卖家等用户代表 |
| 主要参与者 | 买家、卖家 | | |
| 描述 | 用户发现存在爬虫网站源的缺失即存在购物网站未收录，请求更新爬虫网站源 | | |
| 前置条件 | 1. 系统运行正常 2. 用户使用设备运行正常 | | |
| 触发器 | 用户点击更新爬虫网站源按钮 | | |
| 后置条件 | 1. 日志记录本次请求 | | |
| 输入信息 | 爬虫网站源及爬取脚本及映射配置 | | |
| 主干过程 | 1. 用户进入网页主界面，点击头部的“关于”链接跳转到关于页面 2. 用户阅读更新爬虫网站源的说明 3. 用户填写相关输入 4. 用户点击更新爬虫网站源按钮 5. 后台返回成功信息，前台显示 | | |
| 分支过程 | 无 | | |
| 异常 | 1. 爬虫模块报异常（发生于主干过程步骤 5） 处理：根据网站的联系方式联系系统管理员，系统结束用例 2. 用户输入不合法（如爬取脚本存在语法错误，发生于主干过程步骤 5） 处理：返回错误信息，前台显示，系统结束用例 3. 网站源已存在（发生于主干过程步骤 5） 处理：返回错误信息，前台显示，系统结束用例 | | |
| 假设 | 无异常 | | |
| 输入 | 1. 网站源 website，类型为 string 2. 爬取脚本（基于 JavaScript）websiteSpiderScript，类型为 string 3. 映射配置 websiteMap，类型为 Object | | |
| 输出 | 系统提示“更新网站源”成功 | | |
| 包括用例 | 无 | | |
| 优先级 | 中 | | |
| 使用频率 | 低（预估为每周一次） | | |
| 备注 | 无 | | |

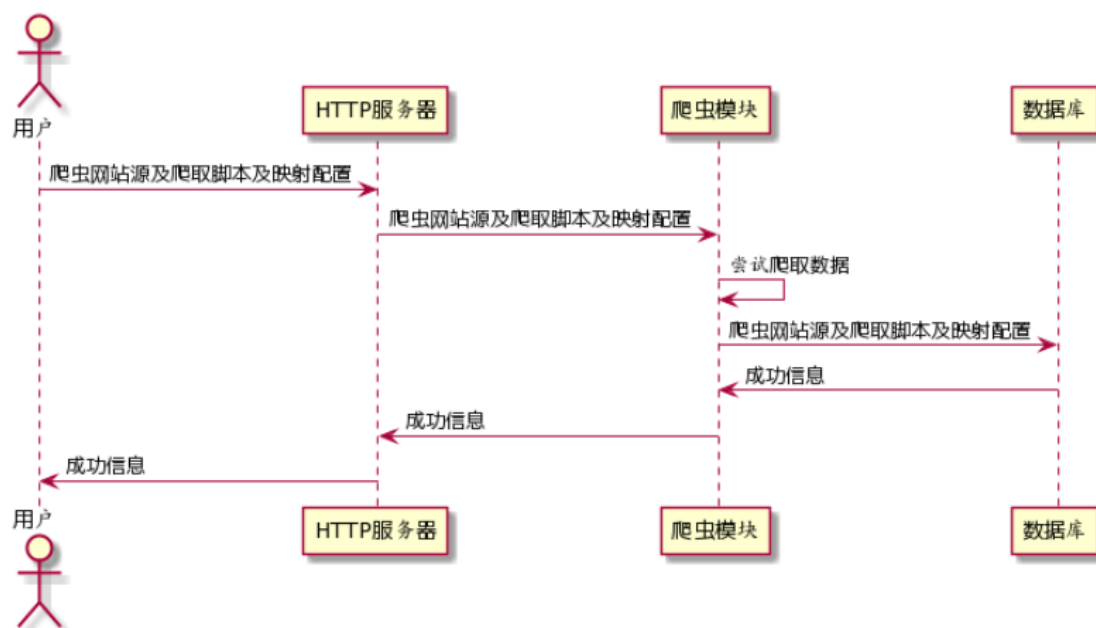


图 4-5-2 用户更新爬虫网站源时序图

5 数据流图

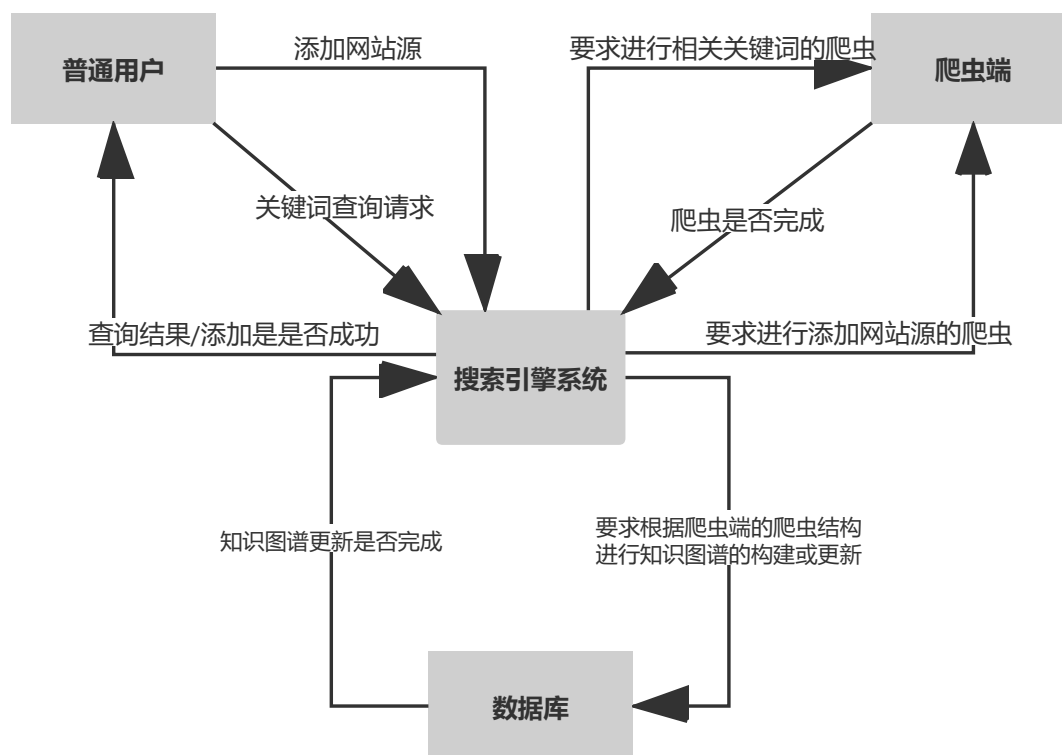


图 5-1 系统数据流图

6 外部接口需求

6.1 硬件接口

6.1.1 服务器

| 资源编号 | 资源名称 | 说明 |
|--------|------|--------------|
| RS-HS1 | 处理器 | 至强 E5 以上 |
| RS-HS2 | 内存 | DDR4 32GB 以上 |
| RS-HS3 | 硬盘 | 2TB 以上 |
| RS-HS4 | 网络 | 100Mbps 对称带宽 |

6.1.2 客户端

| 资源编号 | 资源名称 | 说明 |
|--------|------|-------------|
| RS-HC1 | 处理器 | 酷睿 i3 以上 |
| RS-HC2 | 内存 | DDR4 8GB 以上 |
| RS-HC3 | 硬盘 | 80GB 以上 |
| RS-HC4 | 网络 | 4Mbps 下行带宽 |

6.2 软件接口

6.2.1 服务器

| 资源编号 | 资源名称 | 说明 |
|--------|--------|--------------------------|
| RS-SS1 | 操作系统 | Ubuntu 20.04 |
| RS-SS2 | 前端支持环境 | Nginx |
| RS-SS3 | 后端支持环境 | Python, OrientDB, Docker |

6.2.2 客户端

| 资源编号 | 资源名称 | 说明 |
|--------|------|------------------|
| RS-SC1 | 操作系统 | Windows 10 |
| RS-SC2 | 浏览器 | Chrome 或 Firefox |

6.3 通信接口

| 资源编号 | 资源名称 | 说明 |
|-------|--------|----------|
| RS-P1 | 内网穿透信道 | SSH 反向代理 |

7 非功能性需求

7.1 性能需求

1. 系统能承受 1000 人左右的总用户数量
2. 系统能承受 500 人以上同时访问网站
3. 在同时访问人数不超过 100 人时，系统反应时间（以加载页面计）应少于 1s
4. 在同时访问人数不超过 400 人时，系统反应时间（以加载页面计）应少于 3s

7.2 输入要求

1. 系统能应对用户测试序列，在合理输入范围内（长度低于 40 字符的输入序列）不应导致系统出现超出预期的结果

7.3 数据传输与并发要求

1. 系统在合理网站访问范围内（可承受用户数量输入可接受的输入），系统应能够正确应对并发请求
2. 系统负载（以用户数量计）高于 80%时，带宽利用率不应低于 90%

7.4 数据管理要求

1. 系统提供数据备份的能力，有能力支持每周备份的操作，并提供恢复接口
2. 在进行每周备份时，可采用完整备份的方式备份全部数据，保证上一次可用的备份距当前时间不超过 1 周

7.5 权限与安全需求

1. 系统提供基本的安全能力，能够抵抗基本的 MITM 攻击和 XSS 攻击
2. 服务器操作系统提供面向管理员的安全审计接口，包括服务器登录日志、网站访问日志等
3. 系统提供基本的资料保护功能，包括对用户的资料采取加密措施等

7.6 可视化需求

1. 系统各主要功能入口在主流分辨率下（以 1920x1080 计）均不需滚动即可点击
2. 在主流分辨率下，页面主体内容宽度不超过总宽度的 70%，主体内容居中或偏左

7.7 防护性需求

暂无防护性需求

7.8 软件质量属性

在资源不足时，应优先满足性能和安全性需求。

7.9 其他需求

暂无

8 数据字典

8.1 数据流定义表

表 8-1 数据流定义表

| 编号 | 数据流名 | 来源 | 去向 | 组成 | 说明 |
|----|--------|------|------|------------|------|
| L1 | 关键词查询 | 普通用户 | 搜索引擎 | E-01, E-02 | 搜索 |
| L2 | 添加网站源 | 普通用户 | 搜索引擎 | E-03 | 添加网站 |
| L3 | 爬取关键词 | 搜索引擎 | 爬虫 | E-04 | 关键词 |
| L4 | 查询结果 | 搜索引擎 | 用户 | E-05 | 搜索结果 |
| L5 | 添加成功与否 | 搜索引擎 | 用户 | E-06 | 添加结果 |

8.2 数据元素定义表

表 8-2 数据元素定义表

| 编号 | 数据元素名 | 类型 | 值域 | 说明 |
|------|--------|-----|------------|-------------|
| E-01 | 搜索字符串 | 字符串 | / | 普通搜索字符串 |
| E-02 | 高级搜索描述 | 字符串 | JSON 字符串 | 高级搜索相关信息 |
| E-03 | 网站源 | 字符串 | 合法 URL | 需要爬取的网站 |
| E-04 | 爬取脚本 | 字符串 | 合法脚本 | 爬取网站需要的描述脚本 |
| E-05 | 查询结果 | 字符串 | JSON 字符串 | 搜索后返回的结果 |
| E-06 | 是否成功信息 | 布尔 | true/false | 关键词是否添加成功 |

8.3 数据精度表

表 8-3 数据精度表

| 编号 | 数据元素名 | 类型 | 精度要求 | 说明 | 示例 |
|------|--------|-----|------------|-------------|---|
| E-01 | 搜索字符串 | 字符串 | 最多 40 位字符 | 普通搜索字符串 | 雨伞 |
| E-02 | 高级搜索描述 | 字符串 | JSON 字符串 | 高级搜索相关信息 | {"name": "雨伞"} |
| E-03 | 网站源 | 字符串 | 合法 URL | 需要爬取的网站 | https://www.taobao.com |
| E-04 | 爬取脚本 | 字符串 | 合法脚本 | 爬取网站需要的描述脚本 | / |
| E-05 | 查询结果 | 字符串 | JSON 字符串 | 搜索后返回的结果 | / |
| E-06 | 是否成功信息 | 布尔 | true/false | 关键词是否添加成功 | true |

9 业务规则与业务算法

9.1 业务规则

表 9-1 业务规则

| 标识符 | 规则定义 | 规则类型 | 静态/动态 | 来源 |
|-------|---------------------------|--------|-------|------|
| BR-01 | 用户仅能搜索相关商品的信息 | 约束 | 静态 | 数据限制 |
| BR-02 | 高级搜索栏目支持与或非逻辑运算 | 动作触发规则 | 动态 | 用户 |
| BR-03 | 用户仅能通过高级搜索栏目显示的搜索范围进行高级搜索 | 约束 | 静态 | 管理策略 |
| BR-04 | 用户搜索时语言仅限于中文（含拼音）及英文 | 约束 | 静态 | 管理策略 |
| BR-05 | 搜索结果列表默认按搜索相关度排序 | 约束 | 静态 | 管理策略 |
| BR-06 | 搜索框默认显示用户最近七次搜索的词条 | 约束 | 静态 | 管理策略 |

9.2 业务算法

暂无

10 附录

10.1 术语表

表 10-1 术语表

| 术语名称 | 解释 |
|---------------------------------------|--|
| abstract class 抽象类 | 提供一组子类共有行为的类，但它本身并不具有实例。抽象类表示一个概念，从中派生的类代表对这一概念的实施。 |
| Abstraction 抽象 | 对视图或模型的创建，其中忽略了不必要的细节，以便专注于一组特定的相关细节。 |
| access modifier 存取权限 | 对类、方法或属性进行访问控制的关键字。Java 中的存取权限可以是公有、私有、保护和包装（默认）。 |
| accessor methods 存取器方法 | 由对象提供的、用于定义连接该对象实例变量的方法。用来返回实例变量值的存取器方法被称为获取方法；用来为实例变量指定值的存取器方法被称为设置方法。 |
| Acceptance 验收 | 客户接受软件产品（作为部分或完整履行合同的结果）所有权的操作。 |
| action 动作 | 对构成计算过程抽象的可执行语句的规范。动作通常会导致系统状态发生变化，这是通过向一个对象发送消息或是更改链接或属性值来实现。 |
| action sequence 动作序列 | 解析为一系列先后发生的动作的表达式。 |
| action state 动作状态 | 表示不可分动作的执行状态，通常指的是调用一个操作。 |
| Activation 激活 | 动作的执行。 |
| active class 主动类 | 表示系统中控制线程的类。请参见主动对象。 |
| Activity 活动 | 要求角色执行的工作单元。 |
| active object 主动对象 | 拥有线程并可发起控制活动的对象。主动类的实例。 |
| activity graph 活动图 | 状态机的特例，用于对涉及一个或多个分类器的进程建模。对比：状态图 (statechart diagram)。同义词：活动图 (activity diagram)。 |
| Actor 主角 | 在系统之外与系统交互的某人或某事物。 |
| actor class 主角类 | 定义一组主角实例，其中每个主角实例相对于系统而言都担任着同样的角色。 在与用例交互时这些用例的用户所担任的一组紧密相关的角色。主角为每个要与其通信的用例都准备了一个角色。 |
| actor-generalization 主角泛化关系 | 从一个主角类(后代)到另一个主角类（祖先）的主角泛化关系，表示后代将继承祖先在用例中所能担任的角色。 |

| | |
|---|--|
| actual parameter 实参 | 以值传递的参数 |
| ad hoc review 临时评审 | 是一种非正式的评审，在这样的评审中，过程的执行者可随时请求其同事帮助发现所正在开发之构件中的问题。 |
| Advanced Program-to-Program Communication (APPC) 高级程序间通信 | 主要在 IBM 环境下使用的通信协议。 |
| aggregate class 聚合类 | 在聚合（整体与部分）关系中表示"整体"的类。请参见聚合关系。 |
| Aggregation 聚合关系 | 用于在聚合（整体）及其部分之间建立整体与部分关系的一种关联。 一种特殊形式的关联关系，它确定了介于聚合（整体）和组成部分之间的整体与部分关系。请参见组装。 |
| American Standard Code for Information Interchange (ASCII) 美国国家信息交换标准代码 | 美国国家信息交换标准代码。由绝大多数 PC 和 UNIX 系统使用的 8 位字符解码方案。它取代了早期的 7 位 ASCII 标准。 |
| analysis 分析 | 在软件开发过程中，主要任务在于确立问题领域的模型的流程部分。分析侧重于做什么，而设计则侧重于怎样做。请参见设计。 |
| analysis class 分析类 | 由设计元素在系统中所担任的角色的抽象，通常处于用例实现的环境之下。分析类可以提供若干个角色的抽象，表示这些角色的共有行为。分析类通常会演变为一个或多个设计元素，例如设计类和/或封装体，或者是设计子系统。 |
| analysis & design 分析设计 | 软件开发的前期阶段，目的在于显示如何通过实施实现系统需求。有关分析设计活动的结果，请参见设计模型。 |
| analysis mechanism 分析机制 | 在设计过程初期，即在发现和确定关键类和子系统期间所使用的一种结构机制。通常，分析机制记录了解决方案的主要方面，其中并未考虑实施的影响。分析机制一般与问题领域无关，它是一个"计算机科学"的概念。它们为与领域相关的类或构件提供特定的行为，或者对应于类和/或构件之间协作的实施。它们也可作为框架实施。例如处理永久性、进程间通信、错误或故障处理、通知和消息传递等的机制，其他例子不胜枚举。 |
| analysis pattern 分析模式 | 表示业务建模中的共有构架的一组概念。它可能只与一个领域有关，也可能跨越多个领域。 |
| Analyst 分析员 | 项目团队成员，负责获取并解释涉众需求，并将这些需要传达给整个团队。 |

| | |
|--|---|
| application programming interface (API) 应用程序编程接口 | 使应用程序能够相互通信的软件接口。API 是一组编程语言结构或语句，可编制为程序代码以获取由基础操作系统或服务程序提供的特定功能和服务。 |
| APPC | 高级程序间通信 |
| Appraisal 评估 | 审查组对所审查的产品质量的一种集体判定，包含在跟踪阶段对所要求的返工验证种类的判定。 |
| architectural baseline 构架基线 | 处于精化阶段末期的基线，此时系统的基础结构和行为已较稳定。 |
| architectural mechanism 构架机制 | 构架机制表示对于常见问题的通用具体解决方案。它们可以指结构模式、行为模式，也可以指这两者。在 Rational Unified Process 中，构架机制是一个涵盖分析机制、设计机制和实施机制的术语。 |
| architectural pattern 构架模式 | 构架模式表示软件系统的基本结构组织方案。它提供了一组预定义的子系统、指定它们的职责，并且包括用于组织其间关系的规则和指导。可以进一步解释如下：构架模式是一个特定范围的模式（即解决方案模板），并且也是具体软件构架的模板。它涉及整个系统范围内的特征，并且通常涉及子系统范围内（而非类级别）的关系。尽管从原则上说不是不可能以此方式使构架模式更为细化，构架模式就其本质来说似乎与应用程序领域无关（特定领域的词汇似乎与模式说明无关）。请与分析模式作比较。软件构架文档将提供在系统中使用的构架模式。 |
| architectural view 构架视图 | 以某一特定角度看到的系统构架，注重于表示结构、模块度、核心构件和主要控制流。 |
| Architecture 构架 | <p>系统在其所处环境中的最高层次的概念。软件系统的构架是通过接口交互的重要构件（在特定时间点）的组织或结构，这些构件又由一些更小的构件和接口组成。</p> <p>系统的组织结构。构架可以递归解构为通过接口交互的部件、连接部件的关系以及组装部件的一些限制条件。通过接口交互的部件有类、构件和子系统。</p> |
| Artifact 工件 | 由软件开发过程所生成或使用的文档、模型、说明或软件。同义词：产品 (product)。 |
| artifact guidelines 工件指南 | 有关如何使用特定工件的说明，包括如何创建和修订工件的说明在内。 |
| artifact set 工件集 | 体现系统一个方面的一组有关工件。由于在若干核心工作流程中都要用到一些工件，例如风险列表、软件构架文档和迭代计划。 |
| ASP | 活动服务器页 |
| Association 关联关系 | <p>对多个实例间的双向语义连接进行建模的关系。</p> <p>在两个或多个分类器（指定其实例间的连接）之间的语义关系。</p> |

| | |
|--|---|
| association class 关联类 | 同时具有关联关系和类特征的模型元素。关联类可被视作一种具有类特征的关联关系，或是一个还具有关联关系特征的类。 |
| association end 关联关系端 | 关联关系的端点，它连接关联关系和分类器。 |
| asynchronous action 异步动作 | 发送对象没有暂停来等待结果的请求。对比：同步动作 (synchronous action)。 |
| asynchronous review 异步评审 | 一种不需开会，只需评审者在不同时间独立地完成评语的同级评审。 |
| Attribute 属性 | 由类定义的特征，表示类或其对象的指定特征。属性具有类型，用于定义其实例的类型。 |
| base class 基类 | 其他类或 Bean 要从其中派生的类。基类本身也可由其他基类派生而来。请参见抽象类。 |
| Baseline 基线 | 指在特殊的节点已被同意或批准、可作为后继开发工作的基础的一份文档、一段程序或其他工件。只能通过正式程序，例如：变更管理和配置控制才能进行更改。 |
| Bean | 可用于构建应用程序的小构件。请参见 JavaBean 。 |
| BeanInfo | Bean 的伴生类，确定可访问的方法集，这些方法可以检索有关 Bean 的特征、事件和方法。 |
| Behavior 行为 | 操作或事件（包括其结果在内）的可见效果。 |
| behavioral feature 行为特性 | 模型元素的动态特性，如操作或方法。 |
| behavioral model aspect 模型的行为侧重面 | 强调系统中实例行为的模型侧重面，包括其方法、协作和状态历史记录。 |
| beta testing Beta 测试 | 预先发布的测试，目的在于让一部分抽样选取的目标用户来试用产品。 |
| binary association 二元关联关系 | 在两个类之间的关联关系。是多元关联关系的特例。 |
| Binding 绑定 | 通过为模板参数提供实参来从模板创建模型元素。 |
| boundary class 边界类 | 用于在系统环境和其内部运作之间建立通信模型的类。 |
| break point 断点 | 计算机程序中执行中断的点。 |
| build 工作版本 | 工作版本既可以是系统的可操作版本，也可以是展示要在最终产品中提供的部分功能的部分系统。 |
| business actor 业务主角 | 在业务之外与业务交互的某人或某事。 |
| business actor class 业务主角类 | 定义一个业务主角实例集，其中各个业务主角实例相对于业务而言都担任着相同的角色。 |
| business creation 业务创建 | 其目标为创建一个新业务过程、新业务或新组织的业务工程。 |

| | |
|--|--|
| business engineering 业务工程 | 公司用于根据特定目标来设计其业务的一组方法。业务工程方法可用于业务重建、业务改进，也可用于业务创建。 |
| business entity 业务实体 | 表示业务角色处理或使用的"事物"的业务实体。 |
| business improvement 业务改进 | 执行业务工程，但其中的变更工作仅限于局部而并不波及整个业务。它涉及到削减成本、缩短交付周期以及监督服务和质量。 |
| business object model 业务对象模型 | 说明业务用例实现的对象模型。 |
| business modeling 业务建模 | 包含您可用于对业务进行可视化建模的所有建模方法。这些是您可用于执行业务工程的方法的子集。 |
| business process 业务过程 | 一组逻辑相关活动，目的在于使用组织资源为支持组织目标提供规定的结果。在 Rational Unified Process 中，我们使用业务用例（显示业务的预期行为）和业务用例实现（通过业务角色和业务实体来显示行为是如何实现的）来定义业务过程。另请参见过程。 |
| business process engineering 业务过程工程 | 请参见业务工程。 |
| business reengineering 业务重建 | 执行业务重建，其中的变更工作涉及以全面的观点来看待整个现有业务，并彻底想通为什么要这样做。您将置疑所有现存业务过程并努力通过重新构建寻找合理改进的全新方法。同一含义的其他名称还有业务过程重建 (BPR) 和过程创新。 |
| business rule 业务规则 | 在业务之中必须满足的策略或条件的声明。 |
| business use case 业务用例 | 定义一组业务用例实例的业务用例，其中每个实例都是业务执行的一个操作序列，将生成对特定业务主角来说"有价值的可见结果"。业务用例类包含所有与生成"有价值的可见结果"有关的主工作流程和备用工作流程。 |
| business use-case instance 业务用例实例 | 由生成对特定对象来说有价值的可见结果的业务所执行的动作序列。 |
| business use-case model 业务用例模型 | 业务既定功能的模型。业务用例模型被用作一种基本输入，用于确定组织的各个角色和可交付工件。 |
| business use-case package 业务用例包 | 业务用例包是业务用例、业务主角、关系、图和其他包的集合，通过将其划分为若干个较小部分来建立业务用例模型。 |
| business use-case realization 业务用例实现 | 业务用例实现按照协作的业务对象描述了在业务对象模型中如何实现特定业务用例的工作流程。 |
| business worker 业务角色 | 业务角色表示业务中的一个或一组角色。参与业务用例实现时，一个业务角色和其他业务角色交互并控制业务实体。 |

| | |
|--|--|
| Capsule 封装体 | 一种特定的设计模式，代表系统中已封装的控制线程。封装体是一个已赋予构造型的类，该类具有一组特定的并且是必需和限定性的关联关系和特征。 |
| Cardinality 基数 | 元素集内的元素数目。对比：多重性(multiplicity)。 |
| causal analysis 因果分析 | 追查问题的产生原因，并确定解决办法。 |
| CBD | 基于构件的开发 |
| CCB | 变更控制委员会 |
| CDR | 关键设计评审 |
| CGI | 公共网关接口 |
| change control board (CCB) 变更控制委员会 | CCB 的作用是提供集中的控制机制，以确保妥当地考虑、批准和协调每个变更请求。 |
| change management 变更管理 | 控制和跟踪工件变更的活动。另请参见范围管理。 |
| change request (CR) 变更请求 | 对涉众提出的要变更工件或过程的任何请求的统称。在变更请求中记录的信息是有关当前问题、提议解决方案及其成本的起源和影响的信息。另请参阅扩展请求、缺陷。 |
| checklist | 检查表 |
| checkpoints 检查点 | 某种组织良好的工件应该具有的一组条件。也可采用应作肯定回答的问答形式。 |
| Class 类 | 对于一组具有共同属性、操作、方法、关系和语义的对象的描述。类可使用一组接口来指定它提供给它环境的操作集合。请参见接口。 |
| class diagram 类图 | 显示了一组说明性（静态）模型元素的图,例如类、类型及它们的内容和关系。 |
| class hierarchy 类分层结构 | 共享某一单继承的类之间的关系。所有 Java 类都从 Object（对象）类继承。 |
| class library 类库 | 类的集合。 |
| class method 类方法 | 请参见方法。 |
| Classifier 分类器 | 描述行为和结构特性的机制。分类器包括接口、类、数据类型和构件。 |
| Client 客户端 | 向其他分类器请求服务的分类器。对比：提供端 (supplier)。 |
| client/server 客户机/服务器 | 分布式数据处理中的交互模型，即某一位置的程序向另一位置的程序发出请求并等待响应。发出请求的程序称为客户程序，应答程序称为服务程序。 |
| collaboration 协作 | <ul style="list-style-type: none"> 对于为在某一环境中实施某种行为而交互的对象集的说明。它说明组合在一起以达到某种目的一组合作对象。 |

| | |
|--|---|
| | <ul style="list-style-type: none"> • 它为网络对象的消息交换中所发生的行为获取了一个更为全面的视图。 • 协作体现了计算的三个主要基础结构的统一：即数据结构、控制流和数据流。 • 协作具有动态和静态部分。其中的静态部分说明对象和链接在协作实例化中所担当的角色。而动态部分则由一个或多个动态交互组成，用于显示为执行计算而进行协作的整个过程中所传递的消息流。协作可以具有一组描述其动态行为的消息。 • 带有消息的协作就是交互。 |
| collaboration diagram 协作图 | <ul style="list-style-type: none"> • 协作图说明了对象间进行交互的模式，它通过对象之间的链接及其相互发送的消息显示了参与交互的对象。 • 它是一个包含分类器角色和关联关系角色而不是分类器和关联关系的类图。 • 协作图和序列图都显示了交互，但它们各有侧重。序列图明确显示了时间序列，但未明确显示对象关系。协作图明确显示了对象关系，但却必须从序列号中获取时间序列。 |
| COM | 构件对象模型 (Microsoft) |
| Comment 注释 | 附属于一个元素或一组元素的注释说明。注释不具有语义。对比：约束 (constraint)。 |
| Commit 提交 | 结束一个工作单元的一种操作，该操作将使它对资源（事务或数据）所作的更改永久化。 |
| Common Gateway Interface (CGI) 公共网关接口 | 一种标准协议，Web 服务器通过该协议可以执行在服务器计算机上运行的程序。CGI 程序是响应来自 Web 客户机浏览器的请求而执行的。 |
| Common Object Request Broker Architecture (CORBA) 公用对象请求代理程序体系结构 | 确定提供基础结构的软件总线，即对象请求代理程序 (ORB) 的中间件说明。 |
| communicate-association 通信关联关系 | 介于主角类和用例类之间的关联关系，表示在其实例间存在交互。关联关系的方向可指明通信的发起方。 |
| communication association 通信关联关系 | 在部署图中，表示通信的节点间的关联关系。请参见部署图。 |
| component 构件 | <p>系统中相当重要的、几乎是独立的可替换部分，它在明确定义的构架环境中实现确切的功能。符合并提供一组接口的物理实现的构件。</p> <p>系统中实际存在的可更换部分，它包含了实施，符合并提供一组接口的实现。构件代表系统中的一部分物理实施，包括软件</p> |

| | |
|---|--|
| | 代码（源代码、二进制代码或可执行代码）或其等价物（如脚本或命令文件）。 |
| component diagram 构件图 | 显示构件之间的组织和依赖关系的图。 |
| component model 构件模型 | 构架和 API，允许开发人员确定可组合在一起创建程序的可复用代码段。VisualAge for Java 使用 JavaBean 构件模型。 |
| component-based development (CBD) 基于构件的开发 | 对由构件组装的软件密集型系统的创建和部署，以及这种构件的开发和收集。 |
| composite aggregation 组装关系 | 同义词：组装 (composition)。 |
| composite bean 组合 Bean | 由其他 Bean 构成的 Bean。组合 Bean 可以包含可见 Bean、不可见 Bean 或两者都包括。另请参见 Bean、不可见 Bean 和可见 Bean。 |
| composite class 组装类 | 通过组装关系与一个或多个类相关的类。请参见组装。 |
| composite state 组合状态 | 包含并行（正交）子状态或串行（互斥）子状态的状态。请参见子状态。 |
| composite substate 组合子状态 | 可以和包含在同一组合子状态中的其他子状态并存的子状态。同义词：区域 (region)。请参见组合状态。 |
| composition 组装 | 一种聚合关系关联关系，它具有很强的归属关系，而且部分与聚合关系体的生存期恰巧相同。具有不固定的多重性部件可在组装本身之后创建，但这之后就与组装同生共死，即它们将具有同样的生命周期。这样的部件也可以在组装消亡之前明确删除。组装可以是递归的。同义词：组装关系 (composite aggregation)。 |
| Concrete 具体 | 配置中的实体，它满足最终使用要求，并且对于特定的引用，它可被唯一确定。 |
| concrete class 具体类 | 可以直接实例化的类。对比：抽象类 (abstract class)。 |
| concurrency 并行 | 在同一时间间隔中两个或多个活动同时发生的现象。并行可以通过交替或同时执行两个或多个线程来实现。请参见线程。 |
| concurrent substate 并行子状态 | 可以和包含在同一组合状态中的其他子状态并存的子状态。请参见组合子状态。对比：互斥子状态 (disjoint substate)。 |
| Configuration 配置 | (1) 一般：由其功能单元的性质、个数、主要特性所确定的系统或网络的安排，可应用于硬件或软件配置。 (2) 用于确定系统或系统构件的特定版本的需求、设计和实施。请参见配置管理。 |
| configuration item 配置项 | 配置中的实体，它满足最终使用要求，并且对于特定的引用，它可被唯一确定。 |
| configuration management 配置管理 | 一个支持过程，其目的是标识、确定项目并建立项目基线；控制这些项目的更改和发布；报告并记录这些项目和更改请求的 |

| | |
|--|---|
| | 状态；确保项目的完整性、一致性和正确性；控制存储；处理并交付这些项目。 |
| constraint 约束 | 语义条件或限制。特定约束已在 UML 中预定义，其他可由用户定义。约束是 UML 中的三个可扩展性机制之一。请参见标注值、构造型。 |
| construction 构建 | 软件开发过程的阶段，在该阶段中，软件从可执行构架基线前进到可准备向用户群过渡的这一点上。 |
| Constructor 构造函数 | 与类同名的特殊类方法，用于构建并可能初始化和它同属一个类的对象。 |
| Container 容器 | (1) 一个实例，用于包含其他实例，并为访问内容或进行内容迭代提供操作。（例如：数组、列表和集）。(2) 用于包含其他构件的构件。 |
| containment hierarchy 容器分层结构 | 包含模型元素和其间的包含关系的名字空间分层结构。容器分层结构形成一个非循环图。 |
| Context 环境 | 用于特定目的（如指定操作）的一组相关建模元素的视图。 |
| control chart 控制图 | 一种通过对某过程的单独执行情况进行观察，而表明该过程稳定性的图。 |
| control class 控制类 | 用于针对一个或多个用例的行为进行建模的类。 |
| Conversational 会话式 | 一种通信模型，两个分布式的应用程序在其中以会话形式交换信息。通常一个应用程序先开始（或分配）会话，发送一些数据，然后允许其他应用程序来发送一些数据。两个应用程序交替进行会话，直到一方决定结束为止（取消分配）。会话模型是通信的同步形式。 |
| Cookie | 由您的 Web 浏览器根据您所访问的 Web 站点的请求所创建的小文件，浏览器将在随后访问中将该文件的内容发送给相应站点。 |
| CORBA | 公用对象请求代理程序体系结构 |
| CR | 变更请求 |
| critical design review (CDR) 关键设计评审 | 在瀑布式生命周期中，详细设计结束时进行的主要评审。 |
| Customer 客户 | 生产组织之内或之外的个人或组织，要承担系统在财务方面的责任。在大型系统中，客户可能不是最终用户。他们是开发的产品及其工件的最终接受者。另请参见涉众。 |
| Cycle 周期 | 软件开发生命周期，如 RUP 包括：先启、精化、构建和产品化四个阶段。 |
| Database 数据库 | (1) 根据为一个或多个应用程序提供服务的方案存储的相关数据（带有冗余控制）的集合。 (2) 存储在系统中的所有数据文件。 (3) 存储在一起并由数据库管理系统管理的数据集。 |

| | |
|---|---|
| database management system (DBMS) 数据库管理系统 | 通过为有效访问、完整性、恢复、并行控制、隐私和安全性提供集中控制、数据独立和复杂物理结构等服务来管理数据的计算机程序。 |
| Datatype 数据类型 | 缺乏标识并且操作不带有副作用的一组值的描述符。数据类型包括初始预定义类型和用户可定义类型。预定义类型包括数字、字符串和时间。用户可定义类型可包括各种枚举。 |
| DBA | 数据库管理员 |
| DBCS | 双字节字符集 |
| DBMS | 数据库管理系统 |
| DCE | 分布式计算环境 |
| DCOM | 分布式对象模型 (Microsoft) |
| Deadlock 死锁 | 两个独立控制线程彼此阻塞，相互等待对方采取操作，这种状态称为死锁。死锁常常是由于添加了避免竞争状态的同步机制而引起的。 |
| decision rule 决策规则 | 指一组人员同意就某一议题给出集体决策的规则或技术，例如多数票决、一致通过、自发协议等。 |
| Defect 缺陷 | 缺陷是已交付产品中的异常情况或瑕疵。缺陷可以是在生命周期的初期发现的遗漏和缺点，也可以是已准备测试或运行的较成熟软件中存在的故障征兆。缺陷可以是您要跟踪并解决的任何种类的问题。另请参见变更请求。 |
| defect checklist 缺陷检查表 | 某类指定的工件中常见缺陷的列表。 |
| defect density 缺陷密度 | 每一个单位数量（每页代码或每千行代码）的被评审工作产品中所发现的缺陷的数量。 |
| defect log 缺陷日志 | 见"问题日志"。 |
| defining model 定义模型 | 储存库所基于的模型。任意数量的存储库都可以具有相同的定义模型。 |
| Delegation 委托 | 作为对于一个消息的响应，一个对象可以向另一对象发出消息的能力。委托可用作继承的备选方案。对比：(继承 inheritance)。 |
| Deliverable 可交付工件 | 过程中生成的、要交付给客户或其他涉众的具有值、材料或其他信息的输出。 |
| de-marshal 串行化 | 为使对象可编写为字节流而对对象进行的解构。是串行化 (flatten) 和串行化 (serialize) 的同义词。 |
| demilitarized zone (DMZ) 隔离带 | 此术语目前在业界常用于描述子网，通常用于使防火墙所保护的 Web 服务器免受外部 Internet 和公司内部网络的破坏。 |
| Dependency 依赖关系 | 介于两个建模元素之间的关系，其中，一个建模元素（独立元素）的变更将会影响到另一建模元素（依赖元素）。 |
| Deployment 部署 | 软件工程过程的一个阶段，其目的在于确保将开发的系统顺利转交给用户。其中包括工件，如培训材料和安装步骤。 |

| | |
|----------------------------------|---|
| deployment diagram 部署图 | 显示运行时处理节点以及在其上存活的构件、过程和对象的配置的一种图。构件是代码单元在运行时的具体表现形式。请参见构件图。 |
| deployment unit 部署单元 | 作为一个组分配到一个进程或一个处理器的一组对象或构件。分布单元可由运行时组装或聚合关系表示。 |
| deployment view 部署视图 | 一种构架视图，描述一个或多个系统配置；软件构件（任务、模块）到这些配置中的计算节点的映射。 |
| derived element 派生元素 | 可从其他元素计算得到的模型元素。尽管没有增加语义信息，但可能出于澄清的目的要显示派生元素，也可能出于设计目的要包含它。 |
| Deserialize 反串行化 | 从串行化状态构建一个对象。是反串行化 (marshal) 和反串行化 (resurrect) 的同义词。 |
| Design 设计 | 软件开发过程中的一个部分，其首要目的在于决定系统如何实施。在设计期间，制定战略和战术决策以便满足系统的功能和需求。请参见分析。 |
| design mechanism 设计机制 | <p>在设计过程以及完成细节设计期间中所使用的一种构架机制。它们与关联关系的分析机制有关，它们是所关联关系的分析机制的额外改进，而且它们还可以绑定一种或多种构架和设计模式。</p> <p>分析机制和设计机制之间在规模上不存在泾渭分明的区别，因而以下说法是可能成立的，即在分析层次上的永久性机制和在设计层次上的永久性机制是处于不同改进层次上的同一事物。设计机制假定了实施环境的一些细节，但它并不属于特定的实施（即实施机制）。</p> <p>例如，进程间通信的分析机制可以通过以下若干进程间通信 (IPC) 的设计机制来进一步改进：共享的内存、类似函数调用的 IPC、基于信号量的 IPC 等。</p> <p>每一设计机制都具有特定的优缺点，特定的设计机制是根据使用机制的对象特征来决定的。</p> |
| design model 设计模型 | 描述用例实现的对象模型，用作实施模型及其源代码的抽象。 |
| design package 设计包 | 设计包是由类、关系、用例实现、图和其他包组成的集合，它可用于将设计模型分成更小的部分，通过此方式来构建设计模型。它是实施子系统的逻辑模拟。 |
| design subsystem 设计子系统 | 具有包（可以包含其他元素）和类（具有行为）的语义的模型元素。子系统的行为由它所包含的类或其他子系统提供。子系统实现一个或多个接口，这些接口确定子系统可以执行的行为。对比：设计包 (design package)。 |
| Developer 开发人员 | 负责按照项目采用的标准和步骤开发所需功能的人员。其中可以包括在任意需求、分析设计、实施和测试工作流程中执行的活动。 |
| development case 开发案例 | 由执行组织所使用的软件工程过程。它是作为 Unified Process 产品的配置或定制来开发的，并且满足项目的需求。 |

| | |
|---|--|
| development process 开发过程 | 在软件开发中，为达到给定目的而执行的部分排序的一组步骤，例如构建模型或实施模型。 |
| Device 设备 | 为处理器提供支持功能的一种节点。尽管可以运行嵌入式程序（驱动程序），但是它无法执行通用的应用程序，它只是为处理器提供运行通用应用程序的服务。 |
| Diagram 图 | 对模型的部分或全部进行说明的图形。 |
| disjoint substate 互斥子状态 | 无法和包含在同一组合状态中的其他子状态同时保存的子状态。请参见组合状态。对比：并行子状态 (concurrent substate)。 |
| distributed Computing Environment (DCE) 分布式计算环境 | 分布式的计算环境。计算机行业用于分布式计算的既成标准。DCE 允许不同供应商的计算机透明通信并共享资源，例如网络中的计算能力、文件、打印机和其他对象。 |
| distributed processing 分布式处理 | 分布式处理是一种应用程序或系统模式，功能和数据在其中可以跨越 LAN 或 WAN 上连接的多个资源分布。请参见客户机/服务器计算。 |
| dLL | 动态链接库 |
| dMZ | 隔离带 |
| dNS | 域名服务 |
| Document 文档 | 文档是用于在纸张或在暗喻为纸张的介质上表示的一个信息集合。上述暗喻为纸张的介质包括概念上的纸张，但它既可具有内容的隐含序列，也可具有内容的明确序列。信息将采用文本或二维图片的形式。字处理器文档、电子表格、时间表、甘特图表或投射幻灯演示等形式都是纸介暗喻的示例。 |
| document description 文档说明 | 说明特定文档的内容。 |
| document template 文档模板 | 一个具体的工具模板，例如 Adobe? FrameMaker? 或 Microsoft? Word? 模板。 |
| Domain 领域 | 以一系列有关系统为其特点的知识或活动区域。 |
| domain model 领域模型 | 领域模型记录了在领域环境中最重要的对象类型。领域对象代表存在的实体或在系统运作所在的环境中发生的事件。领域模型是业务对象模型的子集。 |
| domain name server 域名服务器 | 用于翻译域名的系统，例如将 www.software.ibm.com 翻译为 Internet 协议地址，如 123.45.67.8。 |
| double-byte character set (DBCS) 双字节字符集 | 每个字符均由 2 个字节表示的字符集。例如汉语、日语、朝鲜语等语言，它们所包含的符号无法完全由 256 个代码点来表示，因而需要使用双字节字符集。可以和单字节字符集进行比较。 |
| dynamic classification 动态分类 | 泛化关系的语义变化形式，其中对象可以变更类型或角色。对比：静态分类 (static classification) 。 |

| | |
|--|--|
| dynamic information 动态信息 | 在用户请求它时创建的信息。动态信息不时变化，因而在用户每次查看动态信息时都将见到不同的内容。 |
| dynamic link library (DLL) 动态链接库 | 属于运行时程序(而非链接时程序)的可执行代码和数据的文件。C++ Access Builder 生成 Bean 和 C++ 包装程序，以允许 Java 程序可以访问 C++ DLL。 |
| e-Business 电子商务 | 或者指 (1) 通过 Internet 等电子介质进行的业务交易，或者指 (2) 在其内部业务流程（经由 Intranet）中、其业务关系（经由 Extranet）中，以及买卖货物、服务和信息（通过电子商务）的过程中使用 Internet 技术和网络计算的商务运作。 |
| eJB | |
| Elaboration 精化 | 软件开发过程中的阶段，产品的前景及其构架即在此阶段确定。 |
| Element 元素 | 模型的不可再分的最小成份。 |
| Encapsulation 封装 | 对于软件对象的内部表示的隐藏。该对象提供了查询并控制数据而无需暴露其基础结构的接口。 |
| enclosed document 附带文档 | 可由其他文档附带的文档，用于将一组文档组合为一个整体。附带文档和单个附件均可看做独立的工件。 |
| enhancement request 扩展请求 | 一种涉众请求，它说明系统的新特性或新功能。另请参见变更请求。 |
| entity class 实体类 | 用于为已由系统存储并与行为相关联的信息建模的类。一个普通类，可在众多用例中复用，并通常具有永久性特点。实体类定义了一组实体对象，这些对象参与若干用例并通常可在相应用例中存活。 |
| entry action 进入动作 | 在忽略要达到该状态所采取的转移的条件下，进入状态机中的状态时执行的动作。 |
| eRP | 企业资源规划 |
| Event 事件 | 对于所发生的重要事情（具有时间和空间位置）进行的说明。在状态图的环境下，事件就是可以触发转移的事情。 |
| event-to-method connection 事件-方法映射 | 从 Bean 所生成的事件到 Bean 的方法的连接。当连接的事件发生之后，即执行此方法。 |
| Evolution 演进 | 软件在其初始开发循环之后的生命期，在产品不断发展期间的任何后继循环。 |
| Evolutionary 演进方式 | 一种迭代式开发策略，该策略承认并未完全理解用户需要，因而对于需求的理解将在每一后继迭代（精化阶段）中不断完善。 |
| executable architecture 可执行构架 | 可执行构架是系统的部分实施，它建立用于演示选定的系统功能和特征，尤其是那些满足非功能性需求的功能和特征。该构架是在精化阶段中构建的，它用于降低与性能、吞吐量、容量、可靠性以及其他"某某性"方面有关的风险，从而可以在一个坚实的基础上，向构建阶段中添加系统的全部功能，而不必担心破坏系统。 |

| | |
|---|---|
| | 将可执行构架作为一个演进式原型来构建是 Rational Unified Process 的既定意图之一，这样做同时还带有保留有待于进一步开发（满足需求）并使之成为可交付系统的一部分的意图。 |
| exit action 退出动作 | 在忽略要达到该状态所采取的转移的条件下，退出状态机中的状态时执行的动作。 |
| exit criteria 准出条件 | 声明一个过程，如审查过程的执行已成功完成所必须满足的条件。 |
| Export 导出 | 在包的环境下，使元素在其所在的名字空间之外可见。请参见可见性。对比：导出 (export)，导入 (import)。 |
| Expression 表达式 | 对于特定类型的值求解的字符串。例如，表达式"(7 + 5 * 3)"。 |
| Extend 扩展 | 从扩展用例到基础用例的关系，说明为扩展用例定义的行为如何才可插入到为基础用例定义的行为中。 |
| extend-relationship 扩展关系 | 从用例类 A 到用例类 B 的扩展关系表示 B 的实例可以包括（受到在扩展中指定的特定条件的限制）由 A 指定的行为。 由单一目标用例的若干个扩展对象指定的行为可以在单一用例实例中发生。 |
| Façade 外观 | 子系统内一个特殊的包，构造型为 "facade"，用于组织和导出子系统的客户模块所需的所有信息。该包中包括接口（该子系统所特有的一些接口）、与子系统之外接口的实现关系，以及子系统的客户模块要使用该子系统所需的所有文档资料。 |
| factory 工厂 | 不可见的 Bean，能够动态创建指定 Bean 的新实例。 |
| Fault 故障 | 一种意外情况，它导致实施模型中的构件无法完成其预期行为。故障是导致一个或多个缺陷的根本原因。 |
| Feature 特性 | 一种从外部可见的服务，它由系统提供并用于直接满足涉众需要。 |
| Field 字段 | 请参见属性。 |
| file transfer protocol (FTP) 文件传输协议 | 在计算机之间实现文件传输的基本 Internet 功能。借助于它可从远程主机下载文件，也可将文件从本地计算机上载到远程主机。 |
| final state 最终状态 | 一种特殊状态，表示内含的组合状态或整个状态机已完成。 |
| Fire 击发 | 击发用于执行状态转移。请参见转移。 |
| Firewall 防火墙 | 具有可根据确定规则限制进出信息的相关软件的计算机或可编程设备。通常施行控制的依据是始发地址或目标地址，以及 TCP/IP 端口号。 |
| Flatten 串行化 | 串行化 (de-marshal) 的同义词。 |
| focus of control 控制焦点 | 序列图上的一个符号，表示对象直接或通过附属过程执行动作的时段。 |

| | |
|---|--|
| formal review 正式评审 | 一种具备下述所有或多数特征的同级评审方式：评审对象被事先定义，参与者是经过特定培训的团队，由一个训练有素的评审组长领导，定义参与者各自的角色和责任，有一个文档化的评审程序，结果向管理层报告，详细描述准入和准出条件，跟踪所有的缺陷直至其关闭，以及记录过程和质量数据。例如审查、小组评审。 |
| formal parameter 形参 | 同义词：参数 (parameter)。 |
| Framework 框架 | 微型构架，可为特定领域内的应用程序提供可扩展的模板。 |
| FTP | 文件传输协议 |
| HURPS | 功能性、使用性、可靠性、性能和可支持性。该缩写词代表评估产品质量所使用的类别。 |
| Gateway 网关 | 连接以不同语言进行通信的网络的主机；例如：网关可将公司的 LAN 连接到 Internet。 |
| generalizable element 可泛化元素 | 可参与泛化关系的模型元素。请参见泛化关系。 |
| Generalization 泛化关系 | 较为一般的元素与较为特殊的元素之间的一种分类关系。较为特殊的元素除与较为一般的元素完全一致外，还包含其他信息。 凡是可以使用较为一般元素的地方，较为特殊的元素的实例都适用。请参见继承。 |
| Generation 代 | 周期结束时的最终发布版。 |
| graphical user interface (GUI) 图形用户界面 | 一类界面，用户使用它可控制图形化的功能（而不是输入命令）来与程序进行通信。 通常，GUI 是图形、定位设备、菜单栏和其他菜单、重叠窗口以及图标的组合。 |
| green-field development 零起点开发 | "从无到有"的开发，相对于"现有系统的演进"或"遗留系统重建"。 该词源自于在杂草丛生的处女地上建立新工厂时所发生的转变。 |
| guard condition 警戒条件 | 要击发关联关系的转移而必须满足的条件。 |
| GUI | 图形用户界面 |
| home page 主页 | 请参见起始页。 |
| HTML | 超文本标记语言 |
| HTML browser HTML 浏览器 | 请参见 Web 浏览器。 |
| HTTP | 超文本传输协议 |
| HTTP request HTTP 请求 | 通过 Web 浏览器发出的事务，它遵循 HTTP。 |

| | |
|---|--|
| | 服务器通常以 HTML 数据响应，但也可以发送其他类型的对象。 |
| Hypertext 超文本 | 具有指向其他文本的隐藏链接的文档中的文本。使用鼠标单击超文本词语即可转至相应链接指向的文本。 在 Windows 帮助程序和 CD 百科全书中使用超文本，可跳转到位于同一文档的其他位置上的有关参考信息。 超文本的最奇妙之处还是在于：它能够通过 HTTP 链接 Web 上的任何 Web 文档。当然，要在 Web 上四处遨游还需单击鼠标。 |
| hypertext markup language (HTML) 超文本标记语言 | 用于构建万维网上的超文本文档的基本语言。 它在基本的、纯 ASCII 文本文档中使用。 但一经 Netscape 之类的 Web 浏览器解释（称为呈现），该文档即可显示格式化的文本、颜色、各种字体、图形图像、特殊效果、链接到 Internet 其他位置的超文本跳转，以及信息表格。 |
| IT | 信息技术 |
| IDE | 集成开发环境 |
| Idiom 代码模式 | 代码模式是一种编程语言专用的低级模式。代码模式说明如何利用给定语言的特性来实施构件的某些特定方面或实施构件之间的关系，亦称作实施模式。 例如采用 UML 表示的具体设计，并要在 Java 中实施它，则可能需要重现 UML 语言的实施模式。因而在设计和实施中都将使用代码模式。 |
| IEEE | 电气及电子工程师协会 |
| IIOp | |
| iMAP4 | 因特网报文访问协议第 4 版 |
| Implementation 实施 | 软件工程过程中的一个活动，包括编码和测试。 |
| implementation inheritance 实施继承 | 较为特殊的元素的实施的继承。包括对接口的继承。对比：接口继承 (interface inheritance)。 |
| implementation model 实施模型 | 实施模型是构件与包含这些构件的实施子系统的集合。 |
| implementation pattern 实施模式 | 请参见代码模式。 |
| implementation subsystem 实施子系统 | 构件和其他实施子系统的集合，通过细分实施模型对其进行构建。请注意：在 RUP 中，实施模型和实施子系统是实施视图中的目标，因此在开发阶段意义重大。 它是设计包的物理模拟。“实施子系统”这个名称反映了“子系统”这一术语的通常用法，即指比构件范围更大的对象。但是，在 UML 术语中，赋予它的构造型是包，而不是子系统。 |
| implementation view 实施视图 | 一种构架视图，它就打包和分层以及配置管理（所有权、发布策略等）等方面对开发环境中静态软件元素（代码、数据和其 |

| | |
|--|---|
| | 他补充工件)的组织进行说明。在 Unified Process 中, 实施视图是和实施模型有关的视图。 |
| Import 导入 | 在包的环境中的一种依赖关系, 表明那些其类可以在特定包(包含以递归方式嵌入到其中的包)中引用的包。对比: 导出。 |
| import-dependency 导入依赖关系 | 设计中的一种赋予构造型的依赖关系, 以某一设计包为源, 以另一设计包为目标。导入依赖关系允许目标包的公共内容可由源包引用。 |
| Inception 先启 | 软件开发的一个阶段。在此阶段中, 上一代的基本构想(方案征求)已至少在内部得到拨款, 准备工作就绪, 可以进入精化阶段。 |
| Include 包含 | 从基本用例到包含用例的关系, 指定如何将包含用例定义的行为插入到为基本用例定义的行为中。 |
| include-relationship 包含关系 | 包含关系是从基本用例到包含用例的关系, 它指定如何采用显式方式, 将为包含用例定义的行为插入到为基本用例定义的行为中。 |
| Increment 增量 | 在后续迭代结束后, 两个发布版之间存在的差异(差值)。 |
| Incremental 递增 | 一种符合迭代式开发策略的构建方式, 采用此方式构建系统时, 将在每次迭代过程中逐渐增加更多的功能。 |
| informal review 非正式评审 | 一种缺乏正式评审的多项特征的同级评审方式。例如临时评审、同级检查, 以及轮查等。 |
| Inheritance 继承 | 实现泛化关系的机制, 也是从单个类片段中提炼出完整类说明所采用的机制。请参见泛化关系。 |
| injection rate 缺陷率 | 一个工作产品在其开发过程中创建的每单元规模内(每页代码或每千行代码)的缺陷数量。 |
| Input 输入 | 流程所使用的工件。请参见静态工件。 |
| Inspection 审查 | 一种正式的评审方法, 此方法规定一些工件(模型、文档、软件)由创始人之外的个人或小组审核, 旨在检测故障、找出违反开发标准的情况以及其他问题。 |
| inspection effectiveness 审查有效性 | 由审查所发现的一个工作产品的缺陷占该工作产品固有的缺陷的百分比。 |
| inspection efficiency 审查效率 | 每个审查人工小时所发现的平均缺陷数量。 |
| inspection package 审查包 | 由工作产品的作者和评审组长在审查会之前分发给审查人员的一组材料, 包括被审查的工作产品及定义其规格要求的文档、标准、必要的表单、检查表或规则集, 以及测试文档等。 |
| inspection summary report 审查总结报告 | 一份关于被审查的工作产品描述、审查参与人员及其角色分配、对该工作产品的评价及关于审查持续时间、所耗费成本数据的总结报告。 |

| | |
|--|--|
| Inspector 审查者 | 在一个审查过程中担当检查工作产品缺陷的人。他（她）可能还担当诸如读者、记录者或评审组长等指定的角色。工作产品的作者也是一个审查者。 |
| Issue 问题 | 在开发过程中提出的尚未确定为缺陷的事项，如疑问、需澄清的注意点等。 |
| issue log 问题日志 | 所发现的可能缺陷及其改进建议、其他问题的列表 |
| Instance 实例 | 满足类或类型说明的单个实体。请参见对象。 |
| integrated development environment (IDE) 集成开发环境 | 包含编辑器、编译器和调试器的软件程序。 |
| Integration 集成 | 将单独的软件构件组装为可执行的整体系统的软件开发活动。 |
| integration build plan 集成构建计划 | 定义在特定的迭代中实施和集成构件所要采用的顺序。附带在迭代计划中。 |
| Interaction 交互 | 交互说明如何在实例间发送激励来执行特定任务。交互是在协作环境中定义的。请参见协作。 |
| interaction diagram 交互图 | 用于强调对象交互的若干种图的统称。这些图包括：协作图、序列图和活动图。 |
| Interface 接口/界面 | 用于指定类或构件服务的操作的集合。 |
| interface inheritance 接口继承 | 对较为特殊的元素的接口的继承。不包含对实施的继承。对比：实施继承。 |
| internal transition 内部转移 | 表示响应事件但并未改变对象状态的转移。 |
| Internet 互联网 | 使用 TCP/IP 协议的众多互连网络的集合，演变自二十世纪六十年代末、七十年代的 ARPANET。 |
| internet Inter-ORB Protocol (IIOP) Internet ORB 间协议 | 业界标准协议，定义如何通过 TCP/IP 网络交换通用 ORB 间协议 (GIOP) 消息。IIOP 使得 Internet 本身即可用作骨干 ORB，其他 ORB 可借此进行连接。 |
| internet Protocol (IP) Internet 协议 | 提供基本 Internet 功能的协议。 |
| internet protocol address Internet 协议地址 | 一个数字地址，可唯一标识每台连接网络的计算机。例如，123.45.67.8。 |
| Intranet 内部网 | 公司或组织内部的专用网络，使用与公用 Internet 相同种类的软件，不过只供内部使用。 随着 Internet 的日益普及，专用网中使用了许多 Internet 上使用的工具。例如，许多公司的 Web 服务器只供内部员工使用。 |
| IP | |

| | |
|--|--|
| IP number IP 号码 | 以唯一号码表示的 Internet 地址，由四段组成，各段间以圆点（有时称为点分四元组）分隔，例如：198.204.112.1。每台 Internet 计算机都有一个 IP 号码，而且多数计算机还具备一个或多个作为点分四元组映射的域名。 |
| IPSec | |
| IP Security Protocol (IPSec) IP 安全协议 | 提供网络层上的加密安全服务。 |
| ISAPI | |
| ISO | 国际标准化组织。 |
| ISP | 因特网服务提供商 |
| Iteration 迭代 | 带有已建立基线的计划和评估准则的独特活动序列，迭代生成发布版（内部或外部）。 |
| JAR | |
| java | |
| java archive (JAR) Java 档案文件 | 独立于平台的、可将多个文件归为一组的文件格式。JAR 文件用于压缩、缩短下载时间和确保安全性。JAR 格式是以 Java 编写的，因而 JAR 文件可完全扩展。 |
| JavaBean | javaBean 是一种构件，它可与其他单独开发的 Bean 一同集成到应用程序中。 这一应用程序可在浏览器内独立使用，还可用作 ActiveX 构件。对单个进程而言，JavaBean 应在本地运行，而且通常在运行时可见。 |
| JDBC | |
| JDK | |
| JFC | |
| JIT | 刚好、及时 |
| JVM | |
| key mechanism 关键机制 | 对如何根据系统元素间的交互模式实现构架模式的说明。详见软件构架文档。 |
| Keyword 关键字 | 为 Java 保留的预定义字，例如：return，此类字不能用作标识符。 |
| LAN | 局域网 |
| layer 层 | 对模型中同一抽象层次上的包进行分组的一种特定方式。层是对构架的横向划分，而分区是对构架的纵向划分。 对比：分区(partition)。 |
| LDAP | 轻量目录存取协议 |
| Link 链接 | 对象元组之间的语义连接。关联关系的一个实例。请参见关联关系。 |
| link end 链接端 | 关联关系端的一个实例。请参见关联关系端。 |

| | |
|--|--|
| local Area Network (LAN) 局域网 | 位于限定地理区域范围内的用户组织或机构处的计算机网络。LAN 通常由一台或多台服务器组成，可向多台客户机工作站提供服务。 |
| logical view 逻辑视图 | 用于对系统设计中的主要类进行说明的构架视图，包含：与业务相关的关键类，以及定义关键行为和结构机制（永久性、通信、容错、用户界面）的类。 在 Unified Process 中，逻辑视图是设计模型的视图。 |
| major defect 主要缺陷 | 一个工作产品中所存在的那些严重影响产品功能的正确表现、且若在产品开发周期的后期发现将可能比在产品开发前期发现更加难以改正的故障。 |
| Management 管理 | 软件工程过程中的核心支持工作流程，其目的在于计划和管理开发项目。 |
| Marshal 反串行化 | 反串行化 (deserialize) 的同义词。 |
| measurement dysfunction 测量混乱 | 由于人们改变了他们的行为以便使得对某过程的测量给出一些人们喜好或避免受到惩处的结果，使得对该过程的测量出现一种不期望的结果的情形。 |
| Mechanism 机制 | 机制是模式的一个实例。要成为特定模型中的协作，机制还需进一步的改进。因而可将机制看作是在单一环境中针对重复出现问题的一种特定解决方案。 可以说机制与模式是相适应或相符的。 任何协作均可称作机制，但该术语常指能够解决软件应用中频繁出现问题的那些协作。 例如，为处理永久性问题以及可应用模式提供解决方案的协作。 在分析设计中，可以将机制的概念用作“占位符”。举例来说，如果已确定需要永久性，分析员和设计员就可以说将要使用永久性机制，这样就可使问题得到系统且一致的解决。 |
| Message 消息 | 实例间进行信息传递的规约，并具有引起相应活动的期望。消息中可指定发出信号或调用操作。 |
| Messaging 消息传递 | 分布式应用程序借以相互传送消息进行通信所用的通信模型。消息通常是一个较短的信息包，它不一定需要答复。消息传递使用的是异步通信方法。 消息也可是类中的一段 Java 代码，它可以被激活并传递用于执行特定任务的一组参数。 |
| Metaclass 元类 | 实例均为类的一个类。元类通常用于构建元模型。 |
| meta-metamodel 元-元模型 | 定义表示元模型所用语言的模型。元-元模型与元模型之间的关系类似于元模型与模型之间的关系。 |
| Metamodel 元模型 | 对表述模型的语言进行定义的模型。 |
| Metaobject 元对象 | 元建模语言中所有元实体的统称。例如，元类型、元类、元属性以及元关联关系。 |

| | |
|--|--|
| Method 方法 | (1) 经常用来完成某项任务的系统化方式；完成任务或实现目标所遵循的详尽的、组织有序的计划或步骤。 (2) 操作的实施。它指定与操作相关联的算法或过程。 |
| method call 方法调用 | 消息 (message) 的同义词。 |
| Metric 度量 | 对软件构件或过程的某一属性所给出的量化值，它由一个或多个直接测量的数据项提取。例如缺陷密度以及准备速度等。 |
| MIB | 管理信息库 |
| Milestone 里程碑 | 迭代正式结束的时间点；与发布时间点相对应。 |
| MIME | 多用 Internet 邮件扩展 |
| minor defect 次要缺陷 | 一个工作产品中所存在的那些不会对使用者产生严重的结果，且若在产品开发周期的后期发现将不会比在产品开发前期发现更加难以改正的故障。 |
| model 模型 | 以完整的语义表示系统的抽象概念。 在 Unified Process 中，表示从特定角度对系统作出的完整说明（这里的“完整”是指从该角度理解系统时无需补充其他信息）；一组模型元素。两个模型不能重叠。请参见系统。 |
| model aspect 模型侧重面 | 建模的某一方面，用于强调元模型的某些特性。 例如：模型的结构侧重面强调的是元模型的结构特性。 |
| model elaboration 模型精化 | 从已发布的模型生成储存库类型的过程。 在此过程中要生成接口和实施方案，以便根据已精化的模型对储存库进行实例化和填充。 |
| model element 模型元素 | 模型元素用于表示从正在建模的系统中提取的抽象概念。对比：视图元素 (view element)。 |
| model View Controller (MVC) 模型视图控制器 | 用于分解应用程序构件的应用程序构架：模型代表业务逻辑或数据；视图代表用户界面；控制器用于管理用户输入，或者在某些情况下管理应用程序流。 |
| modeling conventions 建模约定 | 表述概念的方式，以及项目团队管理层已决定对建模语言施加的限制。 这些限制即指一些声明，如：“不要在子系统之间使用继承”；“不要在用例模型中使用扩展或包含关联关系”；“不要在 C++ 中使用友元构造”。建模约定在软件构架文档中提供。 |
| Moderator 评审组长 | 领导审查或其他正式评审活动的人，也称作评审领导。他（她）负责同作者一起计划该活动，制定进度、布置会议，从其领导的审查活动中收集和报告测量数据，并且可能参与验证作者的返工结果。作者不应充当该角色。 |
| Module 模块 | 软件的存储和操作单元。模块包括源代码模块、二进制代码模块和可执行代码模块。请参见构件。 |
| MOM | 面向消息的中间件 |
| multiple classification 多重分类 | 泛化关系的一种语义变化形式，其中一个对象可以直接属于多个类。请参见动态分类。 |

| | |
|--|--|
| multiple inheritance 多重继承 | 泛化关系的一种语义变化形式，其中一个类型可以具有多个超类型。对比：单继承 (single inheritance)。 |
| Multiplicity 多重性 | 用于指定某个集可能采用的可允许基数范围。可以为关联关系角色、组合部件、重复以及其他目的指定多重性。 多重性基本上是非负整数的子集（可能是无穷子集）。对比：基数 (cardinality)。 |
| multipurpose Internet Mail Extension (MIME) 多用 Internet 邮件扩展 | 支持文本、图像、音频和视频的邮件所遵循的 Internet 标准。 |
| multi-valued 多值 | 定义了多重性的模型元素，其 Multiplicity Type:: upper 属性被设为大于 1 的数值。 "多值"一词在任何时候都与属性、参数等具有的值个数无关。 对比：单值 (single-valued)。 |
| mutator methods 存取器方法 | 对象提供来定义其实例变量接口的方法。 用于返回实例变量值的存取器方法称为获取方法； 用于为实例变量赋值的存取器方法称为设置方法。 |
| MVC | 模型视图控制器 |
| MVS | 多重虚拟存储器 |
| n-ary association 多元关联关系 | 三个或更多个类之间的关联关系。 关联关系的每个实例分别是相应类中值的一个 n 元组。 对比：二元关联关系 (binary association)。 |
| n-fold inspection N 重审查 | 利用几个小分组来独立地审查同一工作产品。 |
| Namespace 名字空间 | 模型中用于定义和使用名称的部分。在名字空间中，每个名称都有唯一的含义。 |
| NC | 网络计算机或网络计算 |
| NCF | 网络计算框架 |
| NNTP | 网络新闻传输协议 |
| Node 节点 | 节点是一种代表运行时计算资源的分类器。一般来说，节点至少要具备内存，还需有处理能力。运行时对象和构件可驻留在节点上。 |
| NSAPI | |
| NT | |
| Object 对象 | 一种实体，它有明确定义的边界以及封装了状态和行为的标识。 状态由属性和关系来表示，行为由操作、方法和状态机来表示。对象是类的一个实例。请参见类、实例。 |
| object class 对象类 | 用于定义对象属性和方法的模板。一个对象类可包含其他对象类。单独表示的对象类称为对象。 |
| object diagram 对象图 | 对象图包含对象以及对象在某一时刻的关系。对象图可以被看作是类图或协作图的一个特例。请参见类图、协作图。 |

| | |
|---|---|
| object flow state 对象流状态 | 活动图中的一种状态，它表示将对象从一种状态中的动作输出传递到另一种状态中的动作输入。 |
| object lifeline 对象生命线 | 序列图中表示对象存在时间段的一条线。请参见序列图。 |
| object model 对象模型 | 系统实施的抽象表示。 |
| object Request Broker (ORB) 对象请求代理 | 一个 CORBA 术语，表示对象以透明的方式提出请求并从其他本地或远程对象接收响应的方法。 |
| object-oriented programming(OOP) 面向对象程序设计 | 以数据抽象和继承概念为基础的编程方法。与过程编程方法不同，面向对象程序设计侧重于构成问题的那些数据对象以及它们的操作方式，而不是完成任务的方式。 |
| ODBC | 开放数据库连接标准 |
| OLTP | 联机事务处理 |
| OMG | 对象管理组 |
| online transaction processing (OLTP) 联机事务处理 | 一种支持交互式应用程序的计算方式。在这种计算方式中，终端用户所提交的请求一经接收即被处理。处理结果会在相当短的时间内返回给请求方。 联机事务处理系统对资源共享进行监控，以确保在同一时间有效地处理多个事务。 |
| OO | 面向对象 |
| OOP | 面向对象程序设计 |
| Operation 操作 | 可从对象中请求来实现某种行为的服务。每个操作都有一个签名，用以限制可能的实参。 |
| operating system process 操作系统进程 | 类和子系统的实例在其中驻留并运行的唯一地址空间和执行环境。 执行环境可分为一个或多个控制线程。另请参见进程和线程。 |
| ORB | 对象请求代理 |
| organization unit 组织单元 | 业务角色、业务实体、关系、业务用例实现、图和其他组织单元的集合。 组织单元用于将业务对象模型分成较小部分，从而建立业务对象模型的结构。 |
| Originator 发起者 | 发起者是提交变更请求 (CR) 的任何人。 标准的变更请求机制要求发起者按照变更请求的形式提供有关当前问题的信息，以及建议的解决方案。 |
| output 输出 | 作为处理步骤结果的任何工件。请参见可交付工件。 |
| Package 包 | 将元素分组的一种通用机制。包之间可进行嵌套。 |
| pair programming 结对编程 | 一种软件开发活动，在该活动中，通常由两人合作编写代码，共享同一台工作终端，不断地评审和改进其共同的工作。 |

| | |
|---|---|
| parameter 参数 | 用于指定可更改、传递或返回的变量。参数可包含名称、类型和方向。参数在操作、消息和事件中使用。同义词：形参 (formal parameter)。对比：实参 (argument)。 |
| parameter connection 参数连接 | 通过提供特征值或者动作、方法或脚本的返回值来满足动作参数或方法参数的一种连接。参数始终是连接源。另请参见"连接"。 |
| parameterized element 参数化元素 | 带有一个或多个未绑定参数的类的描述符。同义词：模板 (template)。 |
| parent 父 | 在泛化关系中指对子元素的泛化关系。请参见子类、子类型。对比：子 (child)。 |
| parent class 父类 | 其他 Bean 或类从其中继承数据和（或）方法的类。 |
| participates 参与 | 将模型元素连接到关系或具体化的关系。例如：类参与关联关系；主角参与用例。 |
| partition 分区 | (1) 活动图：活动图中用于组织动作职责的部分。请参见泳道。 (2) 构架：同一抽象级别上分类器或包的子集。分区表示对构架的纵向划分，而层表示对构架的横向划分。 对比：层 (layer)。 |
| Passaround 轮查 | 一种非正式的同级评审，工作产品的作者分别请求其他人检查可能的缺陷。 |
| pattern 模式 | 在软件中：指经过验证，至少适用于一种实用环境（更多时候是好几种环境）的解决方案模板（用于结构和行为），"模式"因此而得名。 在 UML 中,模式由参数化的协作来表示，但 UML 不直接对模式的其他方面（如使用结果列表、使用示例等，它们可由文本来表示）进行建模。 要将模式实例化，可将值绑定到该模式的参数。存在各种范围和抽象程度的模式，例如，构架模式、分析模式、设计模式和代码模式或实施模式。 |
| PDR | 初步设计评审 |
| peer deskcheck 同级桌查 | 一种非正式的同级评审，工作产品的作者请求另外一个人检查其可能存在的缺陷。 |
| peer review 同级评审 | 除工作产品的作者之外的一个或多个人检查该产品，以期发现缺陷及其改进时机的一种活动。 |
| peer review coordinator 同级评审协调者 | 一个组织中负责管理和分析该组织的所有同级评审数据、对评审者或评审组长进行辅导、制订培训课程计划并尽可能实施培训、同同级评审的过程拥有者一同工作以利提高评审工作的有效性的那个人。 |
| PERL | 实用析取报告语言。 |
| persistent object 永久对象 | 在创建对象的进程或线程消亡后仍然存在的对象。 |
| PGP | 优秀密钥 |

| | |
|--|--|
| phase 阶段 | 两个主要项目里程碑之间的时间段，在此期间要实现一组既定的目标、完成工件并决定是否进入下一阶段。 |
| PKI | 公开密钥基础设施 |
| pOP3 | 邮局协议 3 |
| port 端口 | 作为 TCP/IP 术语，端口是应用程序可连接到的独立可寻址点。例如：默认情况下，HTTP 使用端口 80，安全 HTTP (HTTPS) 使用端口 443。 |
| post-condition 后置条件 | 在操作结束时必须为"true"的约束。 |
| pRA | 项目评审委员会 |
| pRD | 产品需求文档 |
| pre-condition 前置条件 | 在调用操作时必须为"true"的约束。 |
| preliminary design review (PDR) 初步设计评审 | 在瀑布式生命周期中，构架设计结束时所进行的主要评审。 |
| primitive type 基础类型 | 预定义的、不具有任何子结构的基础数据类型，如整数或字符串。 |
| private 私有 | 与类成员相关联的存取权限。它只允许该类本身访问此成员。 |
| Process 进程、过程 | (1) 操作系统中的重量级并行和执行单元。 对比：线程 (thread)，它包含重量级和轻量级进程。如有必要，可以使用原型在实施方面予以区分。 (2) 软件开发过程，即开发系统所遵循的步骤和指南。 (3) 执行一种算法或是以其他方式动态处理一些事务。 |
| process assets library 过程资产库 | 对过程进行描述的一个集合，包括步骤、模板、表格、检查表，以及其他帮助一个组织成功实施其已经过定义并符合质量要求的过程的那些工作。 |
| process owner 过程拥有者 | 监督一个组织的同级评审活动的管理者或高级过程工程师。通常是评审的一名倡议者，并联系过程评审的各方以进行改进工作。 |
| process view 进程视图 | 描述系统的并行情况（即任务/进程及其交互）的构架视图。 |
| processor 处理器 | 一种能够运行一个或多个进程的节点。通常，它需要具备计算能力、内存、输入输出设备等。另请参见节点、进程和设备。 |
| product 产品 | 作为开发结果的软件，以及一些相关的工件（文档、发布版介质、培训材料）。 |
| product champion 产品推介人 | 掌握产品前景的高层人物，他在产品开发团队和客户之间充当推介人的角色。 |
| product-line architecture 产品线构架 | 定义元素类型、它们的交互方式，以及产品功能映射到这些元素的方式。 它还可能进一步对构架元素的一些实例进行定义。该术语一般可用来指组织或公司内部的一套产品。 |

| | |
|--|--|
| product requirements document (PRD) 产品需求文档 | 在较高层次上对产品（系统）、产品用途及产品特性集作出的说明。 |
| Project 项目 | 项目由相应的人员来完成，受到有限资源的约束，需要得到计划、执行和控制。项目是为创造独特的产品或服务而付出的暂时性努力。 暂时指的是每个项目都有明确的起止时间。 独特指的是该产品或服务有别于所有其他类似的产品或服务。 在实现组织的经营策略时，项目往往是非常关键的部分。 |
| project manager 项目经理 | 承担项目总体角色的角色。项目经理需要确保按照项目时间表、预算和质量要求来安排、分配并完成任务。 |
| project Review Authority (PRA) 项目评审委员会 | 项目经理向其汇报工作的组织实体。PRA 负责确保软件项目符合政策规定、惯例和标准。 |
| projection 投影 | 从一个集到其子集的映射。 |
| Promotion 晋升 | 在 JavaBean 中，指让被包含的 Bean 的特性可用于建立连接。例如，一个 Bean 由面板上的三个按钮组成。 如果将该 Bean 放在一个框架中，则应将这些按钮的特性晋升，让它们可从该框架中进行访问。 |
| property 特征 | 表示元素特点的指定值。特征具有语义上的作用。有些特征已在 UML 中预定义，而其他特征可由用户定义。请参见标注值。 |
| property-to-property connection 特征-特征连接 | 从一个对象的特征到另一个对象的特征的连接。另请参见"连接"。 |
| protected 保护 | 与类成员相关联的存取权限。它允许该类本身、子类和同一包中的所有类访问此成员。 |
| protocol 协议 | 在封装体之间通信所使用的一组可兼容消息的规约。协议用于定义一组输入和输出消息类型（例如操作、信号）；还可以定义一组序列图，以规定消息传递应遵循的顺序。协议也可定义一个状态机，以指定协议的参与者必须提供的抽象行为。 |
| prototype 原型 | 不必经由变更管理和配置控制的发布版。 |
| proxy 代理 | 不同网络之间的一种应用程序网关，可用于特定的网络应用程序（如 FTP 的 Telnet）。 例如，防火墙的代理 Telnet 服务器会执行用户验证，然后让信息流通过代理，就好像代理根本不存在。 这种功能是在防火墙系统中（而不是在客户工作站上）执行的，从而会增加防火墙的负载。请与 SOCKS 作比较。 |
| pseudo-state 伪状态 | 状态机中的一个顶点，它具有状态的形式，但不执行状态的行为。伪状态包含初始顶点和历史顶点。 |
| published model 已发布的模型 | 已定型的模型，可用于对储存库进行实例化，并有助于定义其他模型。已定型模型的元素将无法变更。 |

| | |
|---------------------------------------|---|
| QA | 质量保证 |
| qualifier 限定词 | 一个关联关系属性或属性元组，其属性值用来划分与关联关系对象相关的一组对象。 |
| quality assurance (QA) 质量保证 | 质量保证是项目经理的职责，该角色负责确保所有项目成员都按项目标准正确执行了项目。 |
| race condition 竞争状态 | 两个或更多的独立任务同时访问或修改同一状态信息时出现的状态。 这种状态可能导致系统行为的不一致，因此，这是并行系统设计中的根本问题。 |
| rank 等级 | 用例或场景的一个属性，它描述的是对构架的影响，或说明对发布版的重要性。 |
| rationale 理由 | 陈述或解释作出某种选择的原因。 |
| RDBMS | 关系数据库管理系统。 |
| receive a message 接收消息 | 对从发送方实例传递的激励进行处理。参见发送方、接收方。 |
| receiver object 接收方对象 | 对从发送方对象传递的激励进行处理的对象。对比：发送方 (sender)。 |
| reception 接收 | 一个声明，表明分类器准备好对接收到的信号作出反应。 |
| reference 引用 | (1) 表示模型元素。 (2) 分类器中一个指定的槽，便于快速导航到其他分类器。 同义词：指针 (pointer)。 |
| refinement 改进 | 表明已在某种详细程度上对某事进行更为全面说明的一种关系。例如：设计类是对分析类的改进。 |
| relationship 关系 | 模型元素间的语义连接。举例来说，关联关系和泛化关系就是两种关系。 |
| release 发布版 | 最终产品的一个子集，要在主要里程碑处进行评估。发布版是产品的稳定和可执行版本，随同发布的还有要使用该发布版的所有工件。 如：发布说明或安装说明。发布版可以是内部的，也可以是外部的。内部发布版（作为里程碑的一部分）仅由开发组织使用，也可以用于向用户或客户进行演示。外部发布版（或交付产品）则要交付给最终用户。 发布版不必是一个完整的产品，而可以只是全过程的一个阶段性成果，它的有用性只是从工程角度来衡量的。发布有强制的功能，它使开发团队定期结束任务，从而避免"完成了 90%，还剩下 90%"这种情形的出现。 另请参见原型、基线。 |
| release manager 发布经理 | 发布经理负责确保所有软件资产都已按要求进行控制并可配置为内部或外部发布版。 |

| | |
|---|---|
| remote Method Invocation (RMI) 远程方法调用 | ==JDK 中的 API，用于编写分布式的 Java 程序，让远程 Java 对象的方法可由其他 Java 虚拟机访问。 |
| remote Procedure Call (RPC) 远程过程调用 | 一个通信模型，其中，请求要通过函数调用向分布在其他位置的过程发出。过程的位置对调用方应用程序来说是透明的。 |
| report 报告 | 自动生成的对一个或多个工件的说明。报告本身并不是工件。报告在多数情况下是开发过程中的一个临时产品，它起到了传达演进系统的特定方面的作用。报告是对非文档工件的快照性的说明。 |
| Repository 储存库 | 对象模型、接口和实施方案的存储位置。 |
| requirement 需求 | 需求用于说明系统必须符合的条件或具备的功能。它可以直接来自于用户需要，或在合同、标准、规约或其他正式规定的文档中阐明。 |
| requirement attribute 需求属性 | 与特定需求相关联的信息，它在该需求和其他项目元素（例如：优先级、时间表、状态、设计元素、资源、成本、危害）之间提供链接。 |
| requirements 需求 | 软件工程过程中的核心工作流程，其目的是要确定系统应该做什么。其中最重要的活动就是要制定前景、用例模型和软件需求规约。 |
| Requirements management 需求管理 | 一种系统化的方法，用来获取、组织和记录系统的需求，还要使客户和项目团队在系统变更需求上达成并保持一致。 |
| requirements tracing 需求跟踪 | 从某一个需求到其他需求和其他关联的项目元素的链接。 |
| requirement type 需求类型 | 需求的分类-例如，涉众需要、特性、用例、补充规约、测试需求、文档需求、硬件需求、软件需求等-分类的依据是一般的特征和属性。 |
| resource file 资源文件 | ==Java 程序所引用的文件。图形和音频文件都属于资源文件。 |
| responsibility 职责 | 分类器要履行的合同或承担的义务。 |
| result 结果 | 输出的同义词。另请参见可交付工件。 |
| resurrect 反串行化 | 反串行化 (deserialize) 的同义词。 |
| Review 评审 | 一组人员检查软件工程或其他工程项目的产品的一种活动，目的是要找出一组工件中潜在的缺陷并评估质量。 |
| reuse 复用 | 进一步使用或重复使用某个工件。 |
| Rework 返工 | 作者或其他人修改审查或测试所发现的工作产品的缺陷。 |
| RFC | 征求意见文件。称为 RFC，Internet 标准即在这些文档中定义。 |

| | |
|---------------------------------------|--|
| RMI | 远程方法调用 |
| RMI compiler RMI 编译器 | 生成辅助 RMI 通信的桩模块和框架文件的编译器。该编译器可从"工具"菜单项自动激活。 |
| RMI registry RMI 注册表 | 一个服务器程序，允许远程客户机获得对服务器 Bean 的引用。 |
| risk 风险 | 很有可能阻碍主里程碑成功的当前或未来发生的事项。 |
| role 角色 | 参与特定环境的实体的特定行为。角色可以是静态的（例如：关联关系的一端），也可以是动态的（例如：协作角色）。 |
| RPC | 远程过程调用 |
| s/MIME | 安全 MIME |
| sandbox 沙箱 | 由 Web 浏览器提供的有限环境，可供 Java Applet 在其中运行。沙箱为这些 Applet 提供服务，并阻止它们执行规定之外的操作，如：执行文件输入/输出，与不是装载该 Applet 的服务器通信。如果将 Applet 类推到子级，将调用这些 Applet 运行的沙箱环境。 |
| scenario 场景 | 用于描述行为、按特定顺序排列的动作。场景可用来描述用例实例的交互或执行。请参见交互。 |
| scope management 范围管理 | 根据可用的资源和时间，对可在特定的发布周期中实现的一组需求进行确定和按优先级排序的过程。此过程随着变更的不断出现贯穿整个项目的生命周期。另请参见变更管理。 |
| semantic variation point 语义分歧点 | 元模型在语义上的分歧点。这是为自由解释元模型的语义而特意准备的。 |
| send a message 发送消息 | 从发送方实例向接收方实例传递激励的操作。请参见发送方、接收方。 |
| sender object 发送方对象 | 将激励传递给接收方对象的对象。对比：接收方 (receiver)。 |
| SEPA | 软件工程过程管理委员会 |
| sequence diagram 序列图 | 用于显示按时序安排的对象交互的图。其中突出显示了参与交互的对象以及交换顺序的一系列消息。 与协作图不同，序列图包括了时间顺序，但不包括对象关系。序列图可以是通用格式（说明所有可能的场景）的，也可以是实例格式（说明某一实际场景）的。 序列图和协作图表述的是相似的信息，但表述的方式却不同。请参见协作图。 |
| serialize 串行化 | 串行化 (de-marshall) 的同义词。 |
| server 服务器 | 为网络中的多个用户或工作站提供服务的计算机，例如：文件服务器、打印服务器或邮件服务器。 |
| servlet | |
| SET | 安全电子商务 |
| Severity | 问题或缺陷可能造成的影响程度。 |

| | |
|--|---|
| 严重性 | |
| SHTTP | 安全超文本传输协议 |
| signal 信号 | 对在实例间传送的异步激励的指定。信号可带有参数。 |
| signature 签名 | 一个行为特性的名称和参数。签名中可选择包含返回的参数。 |
| single inheritance 单重继承 | 是泛化关系的语义变化形式，其中，一种类型只能有一个超类型。对比：多重继承 (multiple inheritance)。 |
| single valued 单值 | 定义了多重性的模型元素，当 Multiplicity Type:: upper 属性设置为 1 时它只能有一个值。 "单值"一词在任何时候都与属性、参数等具有的值的个数无关，这是由于一个单值属性（例如：多重性的下限为 0）可能没有值。对比：多值 (multi-valued)。 |
| single-byte character set 单字节字符集 | 每个字符均由单字节代码表示的字符集。 |
| SMTP | 简单邮件传输协议 |
| SNMP | 简单网络管理协议 |
| socket Secure 套接字保护 | 允许相容的客户代码（使套接字安全的客户代码）与远程主机建立会话的网关。 |
| SOCKS | 套接字保护 |
| software Engineering Process Authority (SEPA) 软件工程过程管理委员会 | 负责过程定义、评估和改进的组织实体。 |
| software engineering process group (SEPG) 软件工程过程组 | 一个软件开发组织中领导和协调过程改进活动的小组。 |
| software requirement 软件需求 | 有关系统外部可见行为的规约。例如：系统输入、系统输出、系统功能、系统属性或系统环境的属性。 |
| software requirements specifications (SRS) 软件需求规约 | 完整定义要构建系统的外部行为的一组需求。-有时称为功能规约。 |
| software specification review (SSR) 软件规约评审 | 在瀑布式生命周期中，软件需求规约完成时进行的主要评审。 |
| specification 规约 | 对某事物是什么和做什么的一种公开说明。对比：实施 (implementation)。 |
| SQL | 结构化查询语言 |
| SRR | 系统需求评审 |
| SRS | 软件需求规约 |

| | |
|--|--|
| SSL | 安全套接层 |
| SSR | 软件规约评审 |
| stakeholder 受益者, 涉众 | 会受到系统结果重大影响的个人。 |
| stakeholder need 涉众需要 | 必须解决的业务或运作问题（或能够带来的商机），以此证明购买或使用物有所值。 |
| stakeholder request 涉众请求 | 来自涉众的任意类型的请求-例如：变更请求、扩展请求、需求变更请求、缺陷。 |
| start page 起始页 | 用户浏览 Web 站点时看到的第一页，也称为首页。 |
| state 状态 | 对象生命周期中的状况或情形，在这样的环境中，对象满足某个条件、执行某个活动或等待某个事件。 |
| statechart diagram 状态图 | 显示状态机的图。请参见状态机。 |
| state machine 状态机 | 一个行为，它指定某对象或交互过程在其整个生命周期中对事件作出响应而先后经历的各种状态，同时表明响应和动作。 |
| static artifact 静态工件 | 流程中使用的、但不能改动的工件。 |
| static classification 静态分类 | 泛化关系的语义变化形式，按照这种分类，对象不能改变类型也不能改变角色。对比：动态分类。 |
| static information 静态信息 | 每次访问时都不会改变的 Web 文件。 |
| stereotype 构造型 | 扩展了元模型语义的一种新的建模元素。构造型必须基于元模型中特定的现有类型或类。 构造型可扩展已有类型和类的语义，但不能改动它们的结构。 有些构造型已在 UML 中预定义，其他可由用户定义。 |
| stimulus 激励 | 信息从一个实例到另一个实例的传递，如：发出信号或激活操作。接收到的信号通常就被视作事件。请参见消息。 |
| structural feature 结构特性 | 模型元素的静态特性，例如属性。 |
| structural model aspect 模型的结构侧重面 | 强调系统中对象结构的模型侧重面，包含这些对象的类型、类、关系、属性和操作。 |
| stub 桩模块 | 具备测试功能的构件。桩模块可以是纯粹的“哑模块”，只返回一些预定义的值，也可以“模拟”一个更复杂的行为。 |
| subactivity state 子活动状态 | 活动图中的一种状态，表示执行有一定时长的非基本步骤序列。 |
| subclass 子类 | 在泛化关系中，对另一个类（超类）的特化。请参见泛化关系。对比：超类 (superclass)。 |
| submachine state 子机状态 | 状态机中的一种状态，相当于组合状态，但它的内容由另一个状态机来说明。 |
| substate 子状态 | 组合状态中的状态。请参见并行子状态、互斥子状态。 |

| | |
|---|---|
| subsystem 子系统 | 子系统是对模型元素的分组，其中一些元素构成由分组中另一些元素所提供的行为的规约。请参见包。另请参见系统。 |
| subtype 子类型 | 在泛化关系中，对另一种类型（超类型）的特化。请参见泛化关系。对比：超类型 (supertype)。 |
| superclass 超类 | 在泛化关系中，对另一个类（子类）的泛化。请参见泛化关系。对比：子类 (subclass)。 |
| supertype 超类型 | 在泛化关系中，对另一种类型（子类型）的泛化。请参见泛化关系。对比：子类型 (subtype)。 |
| supplier 提供端 | 一种分类器，它提供可供其他分类器激活的服务。对比：客户端 (client)。 |
| swimlane 泳道 | 活动图上的一个分区，用于划分动作的职责。泳道通常对应于业务模型中的组织单元。请参见分区。 |
| synch state 同步状态 | 状态机中的一个顶点，用来对状态机中的并行区域进行同步。 |
| synchronous action 同步操作 | 发送对象暂停来等待对结果的请求。对比：异步动作 (asynchronous action)。 |
| system 系统 | (1) 为达成特定目的而组成的互连单元的集合。系统可由一个或多个模型从多种不同的角度进行说明。 同义词：物理系统。 (2) 一个顶级子系统。 |
| system requirements review (SRR) 系统需求评审 | 在瀑布式生命周期中，系统规约完成时进行的主要评审的名称。 |
| tagged value 标注值 | 对作为名称值对的特征的明确定义。在标注值中，名称被当作标记。有些标记已在 UML 中预定义，而其他特征可由用户来定义。标注值是 UML 中的三种可扩展性机制之一。请参见约束、构造型。 |
| Task 任务 | 请参见操作系统进程、进程和线程。 |
| TCP | 传输控制协议 |
| TCP/IP | 传输控制协议/Internet 协议 |
| team leader 团队负责人 | 团队负责人负责在项目管理层和开发人员之间进行协调沟通。他将负责对任务进行分配和监督，直至任务完成。他还负责确保开发人员遵循项目标准，遵守项目日程安排。 |
| technical authority 技术权威 | 项目的技术权威有权裁定是否需要实施变更请求以及如何实施，并且还具备相应的技术专长。技术权威将依据变更请求来确定变更任务，并估计完成任务所需的工作量。 |
| telnet | 美国国防部虚拟终端协议。 |
| Template 模板 | 工件的预定义结构。同义词：参数化元素 (parameterized element)。 |
| Test 测试 | 软件工程过程的一个活动，它将软件在预定的条件下运行以判断软件是否符合预期结果，如单元测试、集成测试、系统测试和验收测试等。 |

| | |
|---------------------------------------|--|
| test case 测试用例 | 为特定目标而开发的一组测试输入、执行条件和预期结果，其目标可以是测试某个程序路径或核实是否满足某个特定的需求。 |
| test coverage 测试覆盖 | 一项给定测试或一组测试对某个给定系统或构件的所有指定测试用例进行处理所达到的程度。 |
| test driver 测试驱动程序 | 用于调用测试项的软件模块或应用程序，它可通常用于提供测试输入（数据）、控制和监测执行情况，并报告测试结果。测试驱动程序使测试过程可以自动执行。 |
| test item 测试项 | 作为测试对象的工作版本。请参见工作版本。 |
| test procedure 测试过程 | 测试过程是指设置、执行给定测试用例并对测试结果进行评估的一系列详细步骤。 |
| thin client 瘦客户机 | 瘦客户机通常是指在资源紧张的计算机上运行的系统，或者是指运行小型操作系统的系统。瘦客户机不需要本地系统管理，它们执行通过网络传送的 Java 应用程序。 |
| Thread 线程 | 一种独立的计算，执行位置为其所在的操作系统进程定义的执行环境和地址空间。有时也称作"轻量级进程"。 |
| time event 时间事件 | 表示自进入当前状态后所经过时间的事件。请参见事件。 |
| time expression 时间表达式 | 求解结果为绝对或相对时间值的表达式。 |
| timing mark 时间标记 | 用于表示事件发生或消息出现的时间。在设置约束时可使用时间标记。 |
| tool mentor 工具向导 | 用于说明如何使用特定的软件工具来执行特定的流程活动或步骤。 |
| Traceability 可追踪性 | 将项目元素追踪到其他相关项目元素（尤其是那些与需求相关的元素）的能力。 |
| Trace 追踪 | 一种依赖关系，表示两个元素在历史或流程方面的关系，这两个元素代表同一概念，但没有特定的相互派生规则。 |
| Transaction 事务 | 单个请求启动的一个或多个应用程序所组成的处理单元。执行事务时可能需要启动一项或多项任务。 |
| transaction processing 事务处理 | 一种支持交互式应用程序的计算方式。在这种计算方式中，用户所提交的请求一经接收即被处理。处理结果会在相当短的时间内返回给请求方。事务处理系统可以对资源共享进行监控，以便同时处理多个事务。 |
| transient object 临时对象 | 仅在执行其创建进程或线程的过程中存在的对象。 |
| Type 类型 | 用于说明一组具有共同特点、关系、属性和语义的实体。 |
| type expression 类型表达式 | 求值结果为一个或多个类型的引用的表达式。 |

| | |
|--|---|
| typo list 微错清单 | 一个记录了由审查活动所发现的、在个人准备过程中产生的关于印刷、美化和其他次要错误的列表。微错清单上的条款不会在审查会议上予以讨论。 |
| UI | 用户界面 |
| UML | 统一建模语言 |
| Unicode 统一编码 | 一种字符编码系统，可用于转换、处理和显示用当今世界的多种语言书写的文本。统一编码字符通常使用 16 位无符号整数来进行编码。 |
| unified Modeling Language (UML) 统一建模语言 | |
| uniform resource locator (URL) 统一资源定位符 | 万维网中资源的标准标识符，Web 浏览器使用它来启动连接。URL 包括要使用的通信协议、服务器名称和标识服务器上待检索对象的路径信息。 |
| URL | 统一资源定位符 |
| Usage 用途 | 一个元素（客户）为实现其正确功能或实施需要另一个元素（提供端）参与的依赖关系。 |
| use case 用例 | 用例定义了一组用例实例，其中每个实例都是系统所执行的一系列操作，这些操作生成对于特定主角可见的值。用例类包含所有与产生“可见结果值”有关的主事件流和备用事件流。从技术角度来说，用例是实例为场景的类。 |
| use-case diagram 用例图 | 一种说明系统内主角和用例之间的关系的图形。 |
| use-case instance 用例实例 | 操作序列的执行情况在用例中指定。用例的实例。 |
| use-case model 用例模型 | 根据用例说明系统的功能性需求的模型。 |
| use-case package 用例包 | 用例包是用例、主角、关系、图和其他包的集合，用于通过将用例模型分成若干个较小的部分来建立用例模型。 |
| use-case realization 用例实现 | 用例实现描述如何在设计模型内部使用协作对象来实现一个特定用例。 |
| use-case view 用例视图 | 一种描述在系统中如何执行关键用例的构架视图，主要侧重于对构架具有重要影响的构件（对象、任务、节点）。在 Unified Process 中，它是用例模型的视图。 |
| user interface (UI) 用户界面 | (1) 使用户可以和计算机进行交互的硬件和/或软件。(2) "用户界面"这一术语一般是指可见外观及其底层与用户交互的软件。 |
| Utility 实用工具 | 以类声明的形式对全局变量和过程分组的构造型。实用工具属性和操作分别变成全局变量和全局过程。实用工具不是基本的建模结构，而是一个编程工具。 |
| Validation 确认 | 评价工作产品以判定其是否满足客户需求的过程。 |
| Value 值 | 类型域的元素。 |

| | |
|--|---|
| Variable 变量 | (1) 数据特性在对象内的存储位置。数据特性（如数字或日期）是存储为包含对象的属性的对象。(2) 接收运行时标识的 Bean。变量本身不包含任何数据或程序逻辑；变量必须要进行连接，以便可在应用程序中的其他地方的 Bean 接收运行时标识。 |
| Verification 验证 | 在工作产品开始生成的开发初始阶段评价该工作产品，以判断它是否满足规格说明及条件限制的过程。 |
| Version 版本 | 某个工件的变体；工件的后期版本一般是在初期版本的基础上进行的扩展。 |
| Vertex 顶点 | 状态机中状态转移的来源或目标。顶点既可以是状态，也可以是伪状态。请参见状态、伪状态。 |
| View 视图 | 模型的简化说明（抽象），即采取特定角度或观点并忽略与相应角度或观点无关的实体。另请参见构架视图。 |
| view element 视图元素 | 视图元素是模型元素集合的文本或图形的投影。 |
| view projection 视图投影 | 模型元素在视图元素上的投影。视图投影为每个视图元素提供了位置和样式。 |
| virtual machine (VM) 虚拟机 | 执行其他计算机程序的软件程序。它使实际运行的这台机器（例如一台计算机）表现得就象是另一台实际机器在运行一样。 |
| Visibility 可见性 | 一种枚举，它的值（公有、保护或私有）表明它所引用的模型元素在其所在的名字空间之外的可见性。 |
| Vision 前景 | 用户或客户的待开发产品的视图，它是在关键涉众需要和系统特性的层次上指定的。 |
| visual programming tool 可视化编程工具 | 一种借助于图形来对程序进行说明的工具。应用程序编程人员通过操纵构件的图形表示来编写应用程序。 |
| VM | 虚拟机 |
| VPN | 虚拟专用网络 |
| Walkthrough 走查 | 一种非正式的同级评审，工作产品的作者向评审人员描述该产品并要求对其做出评论。 |
| web application Web 应用程序 | 将 Internet 作为系统用户和系统之间主要通信手段的系统。另请参见 Web 系统。 |
| web browser Web 浏览器 | 在客户机上运行的软件，用户使用它可以发出请求并显示 HTML 页。 |
| web server Web 服务器 | 万维网的服务器构件。它负责为来自于 Web 浏览器的信息请求提供服务。请求信息既可以从服务器的本地磁盘上检索得到的文件，也可以是服务器通过调用程序来执行特定应用程序功能而生成的文件。 |
| web site Web 站点 | 所有内容都在一个服务器上的 Web 系统。用户使用浏览器来浏览 Web 站点。 |
| web system Web 系统 | 包含多个信息页的超级媒体系统，这些信息页通过图形形式互相链接，而不采用分层或线性形式。Web 系统本身就是一个可以通过浏览器访问的 Web 服务器。 |

| | |
|---|--|
| widget 窗口组件 | 在此环境中，它是一个可以放置在窗口中的组件的统称，例如，按钮、滚动条、标签、列表框、菜单或复选框等。 |
| work breakdown structure 工作细分结构 | 计划框架，即将项目分解成多个可以从中分配和追踪成本、工件和活动的工作单元。 |
| work guideline 工作指南 | 提供如何执行一个或一组活动的实践指南的说明。它通常将考虑对于活动比较实用的方法。 |
| work product 工作产品 | 在开发产品软件时所产生的文档、程序或其他成品，可能是项目中间产品或最终发布的产品，以及一些支持项目成功开发的支持性文档。例如各种项目计划、需求规格说明书、设计文档、用户接口设计、源代码、测试文档、用户及系统文档、培训材料，以及过程文档等。参见工件。 |
| worker 角色 | 在软件工程组织的环境中，个人或协同工作的小组的行为和职责的定义。角色代表项目中个人承担的作用，并确定了如何完成工作。 |
| workflow 工作流程 | 在业务中执行的活动序列，它对于业务主角个体生成一个可见值结果。 |
| workflow detail 工作流程明细 | 对于需要密切协作以获得某种结果的活动所进行的分组。这些活动通常以并行或迭代方式执行，一个活动生成的输出将作为另一个活动的输入。工作流程明细用于对活动进行分组，以便提供较高层次的抽象并使工作流程更加简明易懂。 |
| workspace 工作区 | 工作区包含当前正在进行开发的所有代码，即当前版本。工作区还包含标准的 Java 类库和其他类库。 |
| workstation 工作站 | 操作员使用的输入/输出设备配置。一个终端或微型计算机，通常连接主机或网络，用户可以使用它来执行应用程序。 |
| world Wide Web 万维网 | 一种图形化的超文本多媒体 Internet 服务。 |
| WYSIWYG | 所见即所得。 |
| XML | 可扩展标记语言 |

10.2 待确定问题清单

暂无