

# 爱比价网

## 体系结构设计文档



131250030 余泳桦

131250035 罗浩然

131250044 王琨宇

131250081 戚航

131250084 刁子恒

131250086 张笛

131250097 王宁

131250099 刘磊

# 目录

修订历史.....	3
1. 引言.....	4
1.1 编制目的.....	4
1.2 词汇表.....	4
1.3 参考文献.....	4
2. 关注点.....	4
2.1 涉众及关注点描述.....	4
2.2 需求.....	6
2.3 项目环境.....	9
2.4 商业目标.....	9
3. 体系结构需求定义.....	10
3.1 体系结构需求及约束.....	10
3.2 用例视图.....	13
3.3 场景描述.....	15
4. 建立初始体系结构.....	21
4.1 建立依据.....	21
4.2 逻辑视图.....	22
4.2.1 部件和连接件.....	22
4.2.2 配置描述.....	25
4.3 开发视图.....	27
4.3.1 模块组织.....	27
4.3.2 子系统组织.....	28
4.3.3 分层结构.....	29
4.4 进程视图.....	29
4.5 部署视图.....	30
4.5.1 UML 部署图描述.....	30
5. 设计决策.....	31
5.1 设计决策 1.....	31
5.2 设计决策 2.....	32
5.3 设计决策 3.....	32
5.4 设计决策 4.....	33
5.5 设计决策 5.....	34
5.6 设计决策 6.....	34
5.7 设计决策 7.....	35
5.8 设计决策 8.....	36
5.9 设计决策 9.....	37
5.10 设计决策 10.....	37
5.11 设计决策 11.....	38
5.12 设计决策 12.....	38
5.12 设计决策 13.....	39

5.14 设计决策 14.....	39
5.15 设计决策 15.....	40
6. 最终高层结构.....	41
6.1 逻辑视图.....	41
6.2 开发视图.....	43
6.2.1 模块组织.....	43
6.3 进程视图.....	54
6.3.1 进程描述.....	54
6.3.2 进程通信描述.....	55
6.4 部署视图.....	58
6.4.1 基础设施类型.....	58
6.4.2 UML 部署图描述.....	59
7. 体系结构风格.....	59
7.1 MVC 风格.....	59
7.2 面向对象风格.....	60
7.3 管道过滤器风格.....	60
7.4 隐式调用风格.....	61

# 修订历史

修订人员	修订时间	修订原因	版本号
戚航	2015-12-25	初始化文档	1.0
戚航	2015-12-25	完成关注点描述	1.1
戚航	2015-12-27	完成逻辑视图	1.2
全体成员	2016-2-28	完成团队分工	2.0

# 1. 引言

## 1.1 编制目的

本文档提供了爱比价网的高层抽象并且描述了其体系结构及高层次的设计方案。描述了系统高层结构的抽象规格和实现，运行时表现，设计决策等内容，以指导后续的详细设计和开发工作，并持续的起到交流和沟通的作用。

## 1.2 词汇表

缩略语或术语	全意或解释
DD	Design Decision 即设计决策
FR	功能需求
ENV	项目环境
BG	商业目标
QUA	质量属性
AR	体系结构需求
DC	设计约束
AS	体系结构风格
CS	候选方案（Candidate solution）
M	模块
Sub	子系统
IF	接口

## 1.3 参考文献

1. 《爱比价网前景和范围文档》

2. 《爱比价网用例文档》

3. 《爱比价网需求规格说明书》

4. Philippe Kruchten Rational. Software Corp. *Architectural Blueprints—The “4+1” View Model of Software Architecture*. IEEE Software. November1995, pp42-50.

# 2. 关注点

## 2.1 涉众及关注点描述

涉众标识	女性网民
------	------

目标	可以更好更全面地对商品进行比价，帮助他们合理选择商品
任务	<ol style="list-style-type: none"> <li>1. 搜索商品信息</li> <li>2. 产品对比</li> <li>3. 评论商品</li> <li>4. 购买商品</li> </ol>
概念	比价，购买，评论
关注点	<ol style="list-style-type: none"> <li>1. 商品信息充足，覆盖面广，种类齐全</li> <li>2. 商品信息可信度高，来源安全</li> <li>3. 系统使用方便，操作简单，信息直观</li> <li>4. 支付渠道安全可靠，保护账户和个人信息安全</li> </ol>

涉众标识	系统管理员
目标	无
任务	<ol style="list-style-type: none"> <li>1. 管理商品信息</li> <li>2. 植入广告</li> <li>3. 筛选购物网站</li> <li>4. 处理系统异常，监控系统运行状态</li> <li>5. 评论审查和客户管理</li> </ol>
概念	监控，筛选，故障恢复
关注点	<ol style="list-style-type: none"> <li>1. 系统易于维护和调试</li> <li>2. 系统具有优秀的错误诊断和报告机制</li> <li>3. 系统提供充足的人工干预方式，操作简单，易于控制</li> </ol>

涉众标识	数据库管理员
目标	无
任务	<ol style="list-style-type: none"> <li>1. 管理商品信息，人工数据操纵</li> <li>2. 对数据库数据的一致性、正确性进行维护</li> <li>3. 数据库状态监控审查</li> </ol>
概念	数据库，信息，正确一致
关注点	<ol style="list-style-type: none"> <li>1. 数据库管理系统是否熟悉</li> <li>2. 数据库存储是否足够</li> <li>3. 数据库设计方案是否足够安全和健壮</li> <li>4. 对数据量和并发访问的需求</li> </ol>

涉众标识	加盟商城
目标	能够增加商品的销量，扩展销售渠道，丰富客户群体，提高知名度
任务	<ol style="list-style-type: none"> <li>1. 提供该商家的商品信息接口调用</li> <li>2. 提供购买链接</li> </ol>
概念	商品信息，链接
关注点	<ol style="list-style-type: none"> <li>1. 爱比价网的访问流量和客户层次信息</li> <li>2. 不能对自身服务器造成过大的额外数据访问压力</li> <li>3. 能否带来效益和利润</li> </ol>

涉众标识	客户公司
目标	通过系统获得众多商家加盟最终实现收益
任务	1. 提供信息整合平台，对信息进行初步处理如分类排序等 2. 投放广告并管理 3. 提供服务器、域名和系统供用户访问以及商城加盟
概念	信息整合，信息过滤，提供服务
关注点	1. 加盟商家的数量 4. 网站的流量和稳定用户数目 5. 能否获得利润

涉众标识	系统开发商
目标	以最少的成本实现客户的需求
任务	1. 开发系统，提供多版本迭代发布 2. 系统的维护升级
概念	开发，升级
关注点	1. 在规定的工期和预算内完成系统，满足客户需求 2. 能否尽可能复用系统，减少开发成本

## 2.2 功能需求

ID	关注点	类型	具体内容	灵活性
FR1	提供商品信息	功能需求	显示商品的评价、价格、近期价格走势和销量走势信息	商品信息种类具有一定灵活性，信息时间跨度和呈现方式有灵活性
FR2	提供贴吧样式的商品评论平台	功能需求	能够接收用户评论，并且支持用户对评论进行回复，即建立可多次评论回复的讨论机制	基本没有灵活性，基本仿照现行贴吧模式
FR3	提供可靠商家比价	功能需求	显示导航网站前 10 的购物网站及对比信息	对比的数量和方式以及商家的选择有一定灵活性
FR4	提供各大网上商城购物链接	功能需求 & 软件接口	直接链接到对应网站并进入购买流程	没有灵活性
FR5	提供网上支付	功能需求 & 软件接口	调用第三方支付接口	没有灵活性
FR6	搜索结果可调整	功能需求	1. 搜索结果排序展示，允许使用竞价规则对顺序进行调整； 2. 可以屏蔽特定商品，例	调整策略灵活

			如刚刚暴出负面新闻的行为	
FR7	用户管理	功能需求	自动检查用户发布不合法内容的次数，定义不合法标准，定义判定规则，自动判断水军，并通知系统管理员	水军判定规则灵活
FR8	商家关注信息推送	功能需求	允许商家定制词汇表以及一定的规则，并依据此规则给商家发送客户反馈信息。	通知规则具有灵活性，是否通知也是可变的
FR9	自主信息获取	功能需求	要求能够通过网络爬虫，从其他网站获取商品和价格信息	获取源头和爬取方式灵活多样，并且需要多变以应对可能的封锁
QUA1	数据存储高可靠性	质量属性（可靠性）	存储数据出现故障时要能够继续工作并快速恢复	数据库恢复时间小于 12 小时
QUA2	服务器高可靠性	质量属性（可靠性）	服务器出现故障时要能够及时发现，并且不能影响网站访问	不能影响正常访问，故障发现时间小于 1 小时，发现之后错误排查处理时间小于 1 天
QUA3	搜索易用性高	质量属性（可用性）	兼容同义词（定义同义词表）例如搜索 HP 时，可以显示惠普产品	同义词表有灵活性，需要具有一定的自学习能力
QUA4	用户操作的一致性和流畅性	质量属性（可用性）	后台信息的变化对用户是不可见的，不能影响在线使用	没有灵活性，数据一致性的保证是进行每日同步
QUA5	高峰吞吐量	质量属性（性能）	在高峰期，允许五百万用户同时在线使用	保证五百万用户的访问量不会使得系统故障率上升，提供一千万的极限访问量作为冗余，允许达到五百万用户之后响应时间出现 10% 的下降



QUA6	大数据量存储	质量属性 (性能)	最大要能够存储一千万商品的相关信息存储, 并保证存储效率	没有灵活性
QUA7	存取效率	质量属性 (效率)	保证数据的存取效率和对用户操作的响应时间	反应时间的理想应该是 2s 以内, 能够接受的上限是 5s (2M 下载速度)
QUA8	入侵检测	质量属性 (安全性)	2. 必须实现对短时间内频繁访问 IP 进行检测 (即插件扫频攻击) 3. 发现特定 IP (如搜索引擎 Agent) 的访问	检测规则和方法灵活
QUA9	隔离攻击	质量属性 (安全性)	对判定为非法入侵的访问进行有效屏蔽和拦截	基本没有灵活性
QUA10	更多的入侵检测手段支持	质量属性 (扩展性)	将来还会增加其他对访问的入侵检测规则, 需要实现灵活变更	判定规则灵活性大
QUA11	更多的人工干涉条件	质量属性 (扩展性)	将来可以设计新的条件判断规则, 并在条件满足的情况下通知相关人员进行处理	条件灵活性很大
QUA12	兼容各个网站	质量属性 (兼容性)	1. 兼容从各个网站获取的数据的不一致性 (定义匹配格式), 至少支持两个网站 2. 更开放的技术方案, 提供多语言支持, 比如 Amazon.com 等。	兼容网站和内容灵活性很大
QUA13	支持多种浏览器	质量属性 (兼容性)	支持主流浏览器, 比如 IE 系列, Chrome, Firefox 等等。	灵活性不大
QUA14	评论符合相关国家法律法规	约束	1. 定义词汇表, 自动判定敏感词汇 2. 发现后及时告知系统管理员 3. 系统管理员可修改删除	灵活性不大, 必须符合, 但是政策可能发生变化
QUA15	检测评论中的敏感词汇	质量属性 (扩展性)	1. 又管理员进行审查, 审查人员可以废除该条评论, 或是修改评论 2. 自动检测用户之前的评论, 分析是否是水军 3. 如果是商品商家特殊要	可以设计新的条件, 随时进行更新, 通知相应人员

			求的词汇，通知商品商家	
--	--	--	-------------	--

## 2.3 项目环境

ID	关注点	类型	具体内容	灵活性
ENV1	时间	开发环境	计划第一个版本两个月内完成，第二个版本六个月完成	最多可超期限 3 个星期
ENV2	人员	开发环境	开发人员人数有限，其中只有 3 个具有 Web 开发经验，2 个具有移动客户端开发经验，并且技能熟练度不够。并且没有成员具有开发大规模网络爬虫的经验。	基本没有灵活性，不能增加人员，在第一个版本发布之后有一定的时间安排学习
ENV3	团队	开发环境	开发团队包含内部包含 3 个小组，并且各个小组之间技能具有正交性，需要进行并行开发	基本没有灵活性
ENV4	发布计划	商业环境	第一版基础的 Web 应用，提供比价功能，两个月发布；第二版发布移动客户端，六个月发布；第三版提供在线支付功能呢。	基本没有灵活性
ENV5	采用 LAMP 技术构建网站解决方案	技术环境	使用 Linux，Apache，MySQL，PHP 进行网站开发	由于技能局限性，没有灵活性

## 2.4 商业目标

ID	关注点	类型	具体内容	灵活性
BG1	网站用户量	商业目标	在第一版应用 6 个月之后，增加至 500 万用户量，12 个月之后，至 1000 万用户量	基本没有灵活性，可以更多
BG2	合作商家数量	商业目标	在第一版应用之后达到 10 家左右	理想 15 家，最少 5 家
BG3	利润	商业目标	在第一版应用利润提高 30%	理想 50%，最少 20%

## 2.5 质量属性

类别	内容
功能性	系统满足用户的需求
可移植性	系统属于定制型，不考虑可移植性
可靠性	系统具有较高的可靠性，系统错误次数每年不得超过四次，（异常除外）
易用性	网站导航设计良好，用户可以快速找到自己所需的内容
可维护性	系统易于维护，允许全天运行，并允许系统在运行状态下更新系统组件

# 3. 体系结构需求定义

## 3.1 体系结构需求及约束

需求 ID	体系结构需求	优先级*	设计约束		相关约束
			ID	约束内容	
AR1	提供有效正确的商品信息	5	DC1	显示的商品信息来源可靠	阻碍 DC25
			DC2	显示的商品信息是正确的，保证至少在 3 天内是最新数据	
			DC3	显示商品信息的方法多样，包括数字，趋势图。	
AR2	提供商品评论平台	5	DC4	要能够方便的进行评论管理	
AR3	提供可靠商家的价格对比	5	DC5	至少提供 10 个购物网站的对比信息	
AR4	提供各大网上商城购物链接	5	DC6	链接必须有效且采取措施防止恶意链接替换攻击	促进 DC1
AR5	提供网上支付	1	DC7	严格符合网上银行和第三方支付平台的调用接口	
			DC8	与对应在线购物网站达成代售协议并得到相应下订单接口	
			DC9	进行加密处理	
AR6	搜索结果可调整	4	DC10	搜索结果排序显示	阻碍 DC30
			DC11	排序标准可以灵活变更	
			DC12	可以屏蔽特定商品	

AR7	用户管理方便	4	DC13	定义水军的判断规则，保持规则容易变更	
			DC14	发现水军，在 10 分钟内通知系统管理员	
			DC15	提供并维护禁言的账号列表	
AR8	商家关注信息推送	3	DC16	允许商家定制词汇表，并允许商家自定义发送频率和发送标准	阻碍 AR14
AR9	实现自主地信息获取	4	DC17	使用爬虫技术获取网站商品和评论信息	促进 DC2
			DC18	允许自定义爬取网站列表	
			DC19	使用多种爬虫技术和策略进行替换以应对可能的封锁	
AR10	实现数据存储高可靠性	4	DC20	存储数据出现故障时网站能够正常工作	
			DC21	数据恢复时间小于 12 小时	
AR11	实现服务器高可靠性	4	DC22	服务器出现故障时不能影响网站的正常访问	促进 DC25
			DC23	故障发现时间小于 1 小时，恢复时间小于 1 天	
AR12	搜索易用性高	4	DC24	定义同义词列表，显示对应搜索结果	阻碍 DC30
AR13	保证用户操作的一致性和流畅性	2	DC25	后台信息变化不影响用户的使用	
AR14	达到高峰吞吐量要求	3	DC26	保证五百万用户的并发访问量不会使系统故障率上升	阻碍 AR11 阻碍 AR10
			DC27	提供一千万的极限并发访问量作为系统奔溃的可接受范围	
			DC28	允许并发用户超过五百万之后响应时间出现 10% 的下降	
AR15	提供大数据量存储	3	DC29	保证最大能够存储一千万商品的相关信息	阻碍 DC30 阻碍 DC21
AR16	保证存取效率	3	DC30	反应时间理想是 2s 以内，能够接受的上限是 5s（2M 下载速度）	
AR17	实现入侵检测	4	DC31	实现对短时间内频繁访问 IP 进行检测	促进 AR11

			DC32	发现特定 IP 的访问	
AR18	实现隔离攻击	4	DC33	允许定义入侵事件列表	促进 AR11
			DC34	针对事件提供灵活的屏蔽或者拦截措施	
AR19	设计方案提供更多的入侵检测手段支持	3	DC35	增加一个新的入侵检测手段的支持代价小于 0.5 人月	
AR20	允许更多的人工干涉条件进行扩展	2	DC36	增加条件判断和通知人员的代价小于 0.5 人月	
AR21	兼容各个网站	4	DC37	兼容从各个网站获取信息的不一致，至少支持两个网站	促进 D39
			DC38	更开放的语言支持，比如 Amazon.com 等	
			DC39	能够在 0.5 人月内实现对新的网站的格式的兼容	
AR22	支持多种浏览器	4	DC40	兼容 IE8 以及其之后各个版本的 IE 浏览器	
			DC41	兼容 Chrome9.0 浏览器以及之后的各个版本	
			DC42	兼容 FireFox10.0 浏览器以及之后的各个版本	
AR23	保证评论符合相关国家法律法规	DC12	DC43	定义敏感词表，主动判定敏感词	
			DC44	发现敏感词需要在 1min 内通知系统管理员	促进 DC4
			DC45	允许系统管理员修改删除评论	
AR24	设计方案允许快速开发	4	DC46	第一版开发时间小于 2 个月	
			DC47	实施并行开发，子模块划分正交性强	促进 DC49
AR25	有效规避开发人员技能劣势	2	DC48	采用部件技术，对数据爬取部分进行封装	促进 DC47
AR26	支持多版本发布，快速迭代，持续渐进	2	DC49	从项目发起之日起两个月完成第一个版本，六个月完成第二个版本，一年完成第三个版本	
AR27	采用 LAMP 技术构建网站解决方案	5	DC50	使用 Linux 操作系统的服务器，Apache 系统，MySQL 数据库管理系统，PHP 脚本语言进行开发	

AR28	时间效率	4	DC51	反应时间最好在 1s 以内	
AR29	并发性	4	DC52	高峰期间允许 500 万用户在线同时使用	
AR30	可扩展性	4	DC53	增加一个新需求代价小于一人月，增加一个新评论检测规则代价小于 0.25 人月，增加一个入侵检测规则代价小于 0.25 人月	
AR31	团队	3	DC54	采用部件技术，各个部件封装独立功能	促进 DC53
AR32	发布计划	3	DC55	第一阶段期限 1 年 第二阶段期限 6 个月 第三阶段期限 2 个月	
AR33	商业目标	4	DC56	允许定制竞价排序策略	促进 DC2

\*备注：优先级从 1 到 5，数字越大表示优先级越高。

### 3.2 用例视图

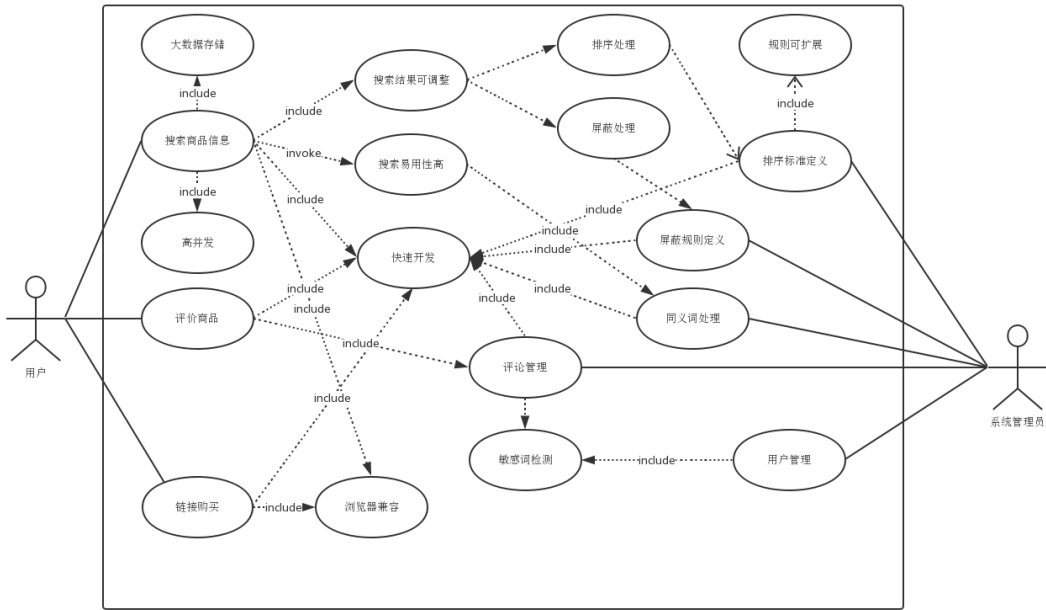


图 1 用例视图（以用户为主）

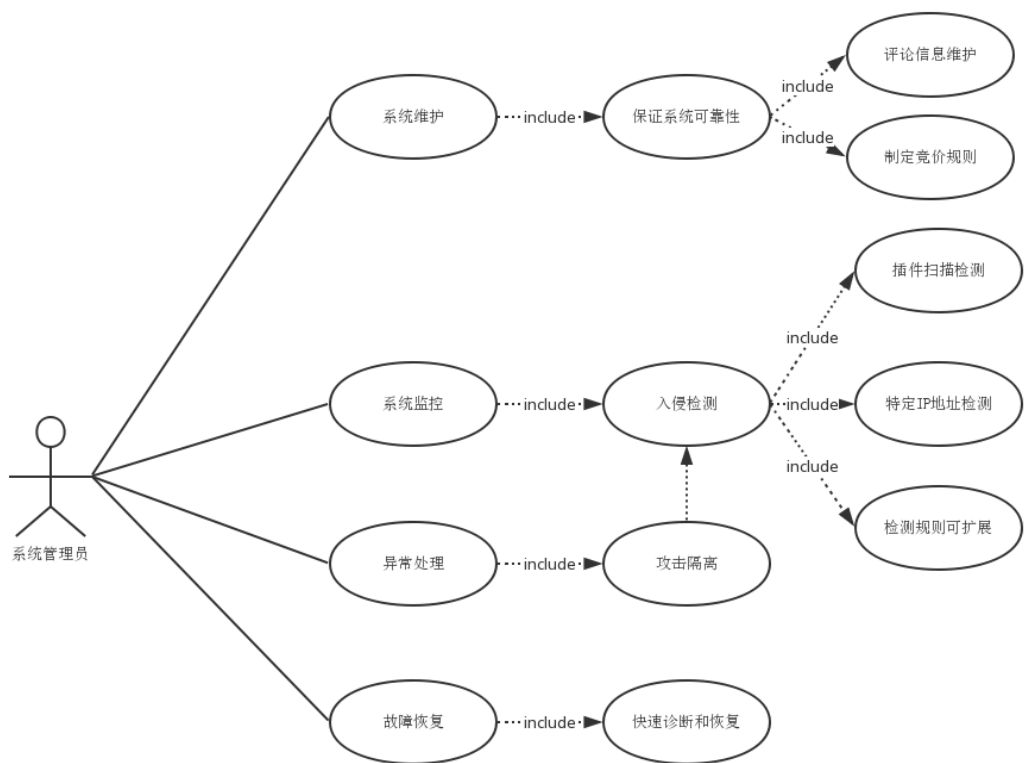


图 2 用例视图 (系统管理员为主)

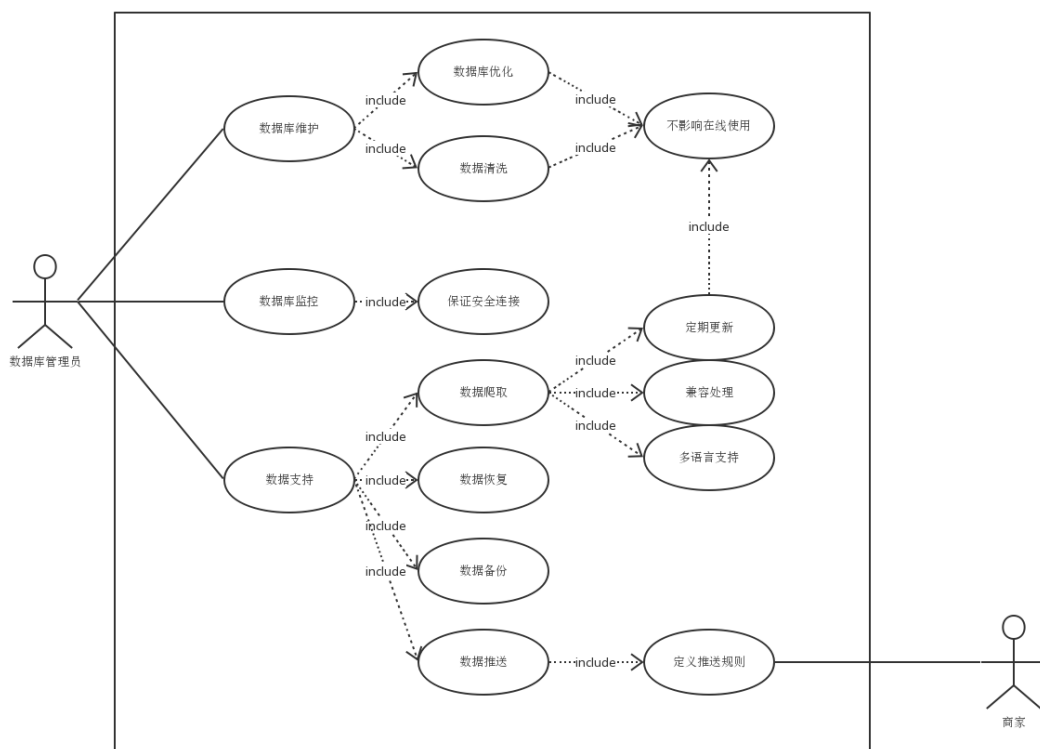


图 3 用例视图 (数据库管理员)

### 3.3 场景描述

场景 ID		S1	
商业目标		调整搜索结果	
相关需求设计约束		DC10 搜索结果排序显示 DC11 排序标准课灵活变更	AR6 搜索结果可调整
场景内容	刺激	竞价规则下产生了新的竞价结果	
	刺激源	商家	
	环境	正常使用时	
	制品	搜索结果排序模块	
	响应	对搜索结果按照竞价规则进行排序	
	响应的度量	任意商品所在位置与其商家在竞价结果中所购买的位置误差不能超过 $\pm 1$ 位	

场景 ID		S2	
商业目标		屏蔽特定搜索结果	
相关需求设计约束		DC12 可以屏蔽特定商品	AR6 搜索结果可调整
场景内容	刺激	发生需要屏蔽事件，商品符合屏蔽条件	
	刺激源	外部环境（比较复杂，包括舆论，商家的产品等）	
	环境	正常使用时	
	制品	搜索结果过滤模块	
	响应	搜索结果过滤模块加入新的过滤规则	
	响应的度量	任何符合屏蔽条件的商品绝对不能够出现在显示结果中	

场景 ID		S3	
商业目标		用户管理	
相关需求设计约束		DC13 定义水军的判断规则，保持规则容易变更 DC14 发现水军，在 10 分钟内通知管理员 DC15 提供并维护禁言的账号列表	AR7 用户管理方便
场景内容	刺激	检测到符合水军判断规则的用户	
	刺激源	用户	
	环境	正常使用时	
	制品	禁言账号列表	
	响应	通知管理员	
	响应的度量	一旦检测到符合水军定义的用户，则需要在 10 分钟内通知系统管理	



	量	员，管理员及时人工做出处理
--	---	---------------

场景 ID		S4	
商业目标		推送商家关注信息	
相关需求设计约束		DC16 允许商家定义词汇表，并允许商家自定义发送频率和发送标准	AR8 商家关注信息推送
场景内容	刺激	检测到符合商家定义标准的信息（比如特定词汇出现的频率等）	
	刺激源	用户	
	环境	正常使用时	
	制品	信息推送模块	
	响应	按照商家对发送时间的要求把信息推送给商家	
	响应的度量	发送信息给商家的时间距离检测到该信息的时间与商家规定的误差在 30 分钟以内	

场景 ID		S5	
商业目标		自主获取商品信息	
相关需求设计约束		DC17 使用爬虫技术获取网站商品和评论信息 DC18 允许自定义爬取网站列表 DC19 使用多种爬虫技术和策略进行替换以应对可能的封锁	AR9 实现自主地信息获取
场景内容	刺激	信息爬取发生异常，数据量下降，爬虫工作状态异常	
	刺激源	数据爬虫模块	
	环境	爬虫故障	
	制品	信息爬虫模块	
	响应	通知系统管理员，进行故障排除或者爬取策略替换	
	响应的度量	故障排除的时间小于 1 周	

场景 ID		S6	
商业目标		保证数据访问高可靠性	
相关需求设计约束		DC20 存储数据出现故障时网站能够正常工作	AR10 实现数据存储高可靠性
场景内容	刺激	产生数据查询或更新请求	
	刺激源	用户	
	环境	数据存储出现故障时	
	制品	备份数据	
	响应	正常地响应用户请求	
	响应的度量	响应用户请求的效率满足一般标准	

场景 ID		S7	
-------	--	----	--

商业目标		快速数据恢复	
相关需求设计约束		DC21 数据库恢复时间小于 20 小时	DC21 数据库恢复时间小于 20 小时
场景内容	刺激	检测到数据库存储故障	
	刺激源	数据库	
	环境	数据存储出现故障时	
	制品	数据库修复模块	
	响应	通知数据库管理员修复问题	
	响应的度量	故障发生与通知管理员的时间间隔小于 12 小时 数据库管理员在 20 小时内修复故障	

场景 ID		S8	
商业目标		服务器访问可靠	
相关需求设计约束		DC22 服务器出现故障时不能影响网站的正常访问	AR11 实现服务器高可靠性
场景内容	刺激	收到 Web 请求	
	刺激源	用户	
	环境	服务器出现故障时	
	制品	负载分配模块和备份服务器	
	响应	把请求分配给备份服务器使得能够正常地响应用户请求	
	响应的度量	响应用户请求的效率满足一般标准	

场景 ID		S9	
商业目标		服务器快速恢复	
相关需求设计约束		DC23 故障发现时间小于 1 小时	AR11 实现服务器高可靠性
场景内容	刺激	服务器发生异常	
	刺激源	异常检测模块发出信号	
	环境	服务系出现故障时	
	制品	故障检测排查模块	
	响应	通知系统管理员服务器故障	
	响应的度量	故障发现时间小于 1 小时 故障恢复时间小于 1 天	

场景 ID		S10	
商业目标		后台信息变化不影响用户使用	
相关需求设计约束		DC25 后台信息变化不影响用户的使用	AR13 保证用户操作的一致性和流畅性
场景内容	刺激	用户检索信息时	
	刺激源	用户	
	环境	系统运转正常且后台数据信息发送正常变化时	
	制品	数据更新模块和数据镜像	

	响应	用户检索数据镜像 数据更新模块修改数据 更新数据镜像
	响应的度量	响应用户请求的效率满足一般标准

场景 ID		S11	
商业目标		支持五百万并发用户量	
相关需求设计约束		DC25 保证五百万用户的并发访问量不会使得系统故障率上升 DC26 提供一千万的极限并发访问量作为崩溃标准 DC27 允许并发用户超过五百万之后响应时间出现 10% 的下降	AR14 达到高峰吞吐量要求
场景内容	刺激	并发访问用户量达到五百万	
	刺激源	用户	
	环境	正常使用时（特定的有大量用户使用的时间）	
	制品	负载均衡模块	
	响应	负载均衡模块对服务器访问进行分流和调整，保证正常使用	
	响应的度量	五百万用户以下，系统必须保证故障率和一般水准持平，大于五百万用户每天故障次数不能超过原来的 10%	

场景 ID		S12	
商业目标		提供大数据量存储	
相关需求设计约束		DC29 保证最大能够存储一千万商品的相关信息	AR15 提供大数据量存储
场景内容	刺激	新数据产生，需要存储	
	刺激源	数据爬虫或者用户请求	
	环境	系统正常使用时	
	制品	存储管理模块和磁盘整理模块	
	响应	存储管理模块为数据选择合适的物理节点和存储空间进行存储 磁盘整理模块定期整理磁盘碎片	
	响应的度量	能够保证至少存储一千万商品的相关信息	

场景 ID		S13	
商业目标		对用户请求快速响应	
相关需求设计约束		DC30 反应时间理想是 2s 以内，能够接受的上限是 5s（2M 下载速度）	AR16 保证存取效率
场景内容	刺激	用户发出服务请求	
	刺激源	用户	
	环境	正常使用时	

容	制品	信息检索模块
	响应	信息检索模块执行检索，返回信息
	响应的度量	反应时间理想是 2s 以内，能够接受的上限是 5s

场景 ID		S14	
商业目标		入侵检测	
相关需求设计约束		DC31 实现对短时间内频繁访问 IP 进行检测 DC32 发现对特定 IP 的访问	AR17 实现入侵检测
场景内容	刺激	出现恶意攻击或入侵	
	刺激源	恶意用户	
	环境	正常使用时	
	制品	入侵检测模块	
	响应	入侵检测模块发现入侵，通知系统管理员	
	响应的度量	满足构成入侵的事实标准和检测到入侵的间隔不超过 10s 检测到入侵和通知系统管理员的时间间隔不超过 10s	

场景 ID		S15	
商业目标		隔离攻击	
相关需求设计约束		DC33 允许定义入侵事件列表 DC34 针对事件提供灵活的屏蔽或者拦截措施	AR18 实现隔离攻击
场景内容	刺激	检测到入侵	
	刺激源	入侵检测模块	
	环境	系统受到攻击时	
	制品	攻击隔离模块	
	响应	攻击隔离模块通过匹配攻击事件列表进行屏蔽和拦截操作	
	响应的度量	攻击发起方被成功拦截	

场景 ID		S16	
商业目标		更多入侵检测手段支持	
相关需求设计约束		DC36 增加一个新的入侵手段检测的支持代价小于 0.5 人月	AR19 设计访问提供对更多入侵检测手段支持
场景内容	刺激	采用新的入侵检测手段	
	刺激源	客户公司（爱比价网）	
	环境	正常使用时	
	制品	入侵检测模块	
	响应	开发者扩展入侵检测模块	
	响应的度量	从提出扩展要求到开发结束并测试通过的时间小于 0.5 人月	

场景 ID		S17	
商业目标		更多人工干涉条件	
相关需求设计约束		DC36 增加条件判断和通知人员的代价小于 0.5 人月	AR20 允许更多的人工干涉条件进行扩展
场景内容	刺激	产生新的需要通知特定人员的事件	
	刺激源	客户公司内部或者是其他网站	
	环境	正常使用时	
	制品	消息通知模块	
	响应	开发者定义新的事件和通知列表，对系统进行扩展	
	响应的度量	从产生扩展请求到交付的时间小于 0.5 人月	

场景 ID		S18	
商业目标		兼容多个网站	
相关需求设计约束		DC37 兼容从各个网站获取信息的不一致，至少支持两个网站 DC38 更开放的语言支持 DC39 能够在 0.5 人月内实现对新的网站的格式的兼容	AR21 兼容各个网站
场景内容	刺激	从新的网站获取数据	
	刺激源	客户公司	
	环境	正常使用时	
	制品	格式转换模块	
	响应	开发者开发新的格式转换模块	
	响应的度量	能够在 0.5 人月内实现对新的网站的数据格式的兼容	

场景 ID		S19	
商业目标		支持多种浏览器	
相关需求设计约束		DC40 IE8 以及其之后各个版本的 IE 浏览器 DC41 兼容 Chrome9.0 浏览器以及之后的各个版本 DC42 兼容 FireFox10.0 浏览器以及之后的各个版本	AR22 支持多种浏览器
场景内容	刺激	来自支持范围内浏览器的网页请求	
	刺激源	用户	
	环境	正常使用时	
	制品	前台兼容显示模块	
	响应	返回请求网页	
	响应的度量	不正常显示的频率低于 99%	

场景 ID		S20	
商业目标		评论符合规范	
相关需求设计约束		DC43 定义敏感词表, 主动判定敏感词 DC44 发现敏感词需要在 1min 内通知系统管理员 DC45 允许系统管理员修改删除评论	AR23 保证评论符合相关国家法律法规
场景内容	刺激	评论中含有敏感词汇	
	刺激源	用户评论	
	环境	正常使用时	
	制品	文本检测模块	
	响应	1. 系统检测到敏感词汇, 通知系统管理员 2. 系统管理员修改或删除评论	
	响应的度量	能够检测出 95% 的敏感内容 系统管理员需要在收到通知后 1 小时内做出处理	

## 4. 建立初始体系结构

### 4.1 建立依据

由用例图可知,系统分为主要功能为查看商品信息、搜索商品、发表评论、商品信息爬取与存储、维护评论信息、水军检测,数据库与服务器的维护。其中搜索商品包含对搜索结果进行竞价排序、屏蔽特定商品;爬取商品信息包括对信息爬取、信息解析存储、爬取信息更新。系统的维护包括攻击检测:发现短时间内频繁访问 IP 并屏蔽;屏蔽特定 IP 的访问。因此将系统主要分为如下几个模块:搜索、查看、评论商品信息模块;爬取、存储、维护数据;入侵检测、隔离攻击;系统维护。

借鉴已有的 Web 应用框架和以前的开发经验,建立了初步的体系结构设计方案,首先参考了体系结构风格,由于系统属于 Web 应用,比较适合 MVC 风格的体系结构,所以采用其指导设计,在层次级别的抽象上,把系统分为三个子模块,即 Model, View, Controller。以前期的体系结构功能需求和非功能需求分析结果,结合用例视图实现功能分配和质量属性的分配。在此期间,结合了模块化和信息隐藏等基本设计原则,使得设计方案更容易满足质量需求。

使用流行的“4+1View”对体系结构进行描述,采用 UML 扩展的方法进行展示,用例视图参见本文档 3.2 一节,4 个 View 分别是逻辑视图,开发视图,进程视图和部署视图。其中,逻辑视图关注软件体系结构的需求,开发视图描述了软件体系结构的抽象规格,进程视图关注软件体系结构的运行时表现,部署视图关注可靠性、容错性、吞吐量容量等与基础设施相关的非功能性需求。

## 4.2 逻辑视图

### 4.2.1 部件和连接件

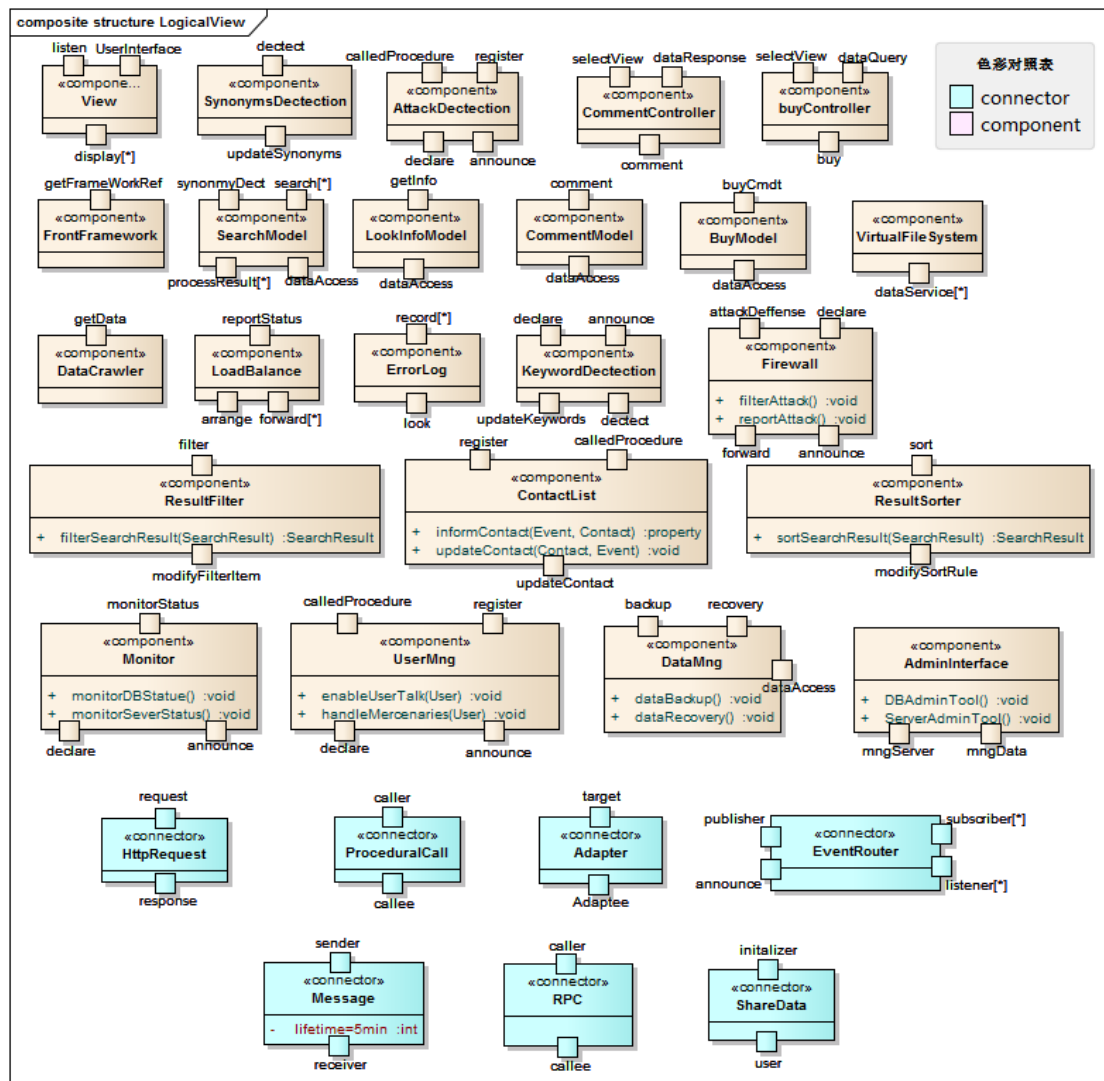


图 4 比价系统的部件类型和连接件类型描述

比价系统包括 23 个部件类型和 7 个连接件类型：

- (1) 部件 View，主要功能是作为与用户直接交互的接口，接受用户请求，显示信息给用户并负责进行展示效果渲染。端口 display 负责展示信息，可以有多个实例；端口 listen 负责监听逻辑层，以获得事件广播；端口 UserInterface 描述了负责与用户进行直接交互。
- (2) 部件 commentController，主要功能是分发评论请求，执行界面选择。端口 comment 描述了对评论的需求，端口 selectView 负责执行界面选择，端口 dataResponse 提供数据响应服务。
- (3) 部件 buyController，主要功能是封装购买相关的控制逻辑，分发购买请求，执行界面选择。端口 buy 负责获得购买链接和信息，端口 selectView 负责执行界面选择，端口 dataResponse 提供数据响应服务。
- (4) 部件 FrontFramework，主要功能是封装提供浏览器兼容性的前端框架，实现浏览器兼容性。端口 getFrameworkRef 通过现行前端框架的复用提供浏览器兼容服务。



- (5) 部件 SearchModel, 主要功能是执行搜索相关的业务逻辑。端口 search 提供搜索商品服务, 端口 synonymyDect 提供执行同义词检测, 端口 processResult 负责对搜索结果进行包括排序、筛选在内的处理, 端口 dataAccess 描述了对于数据访问的需求。
- (6) 部件 LookInfoModel, 主要功能是执行查看信息相关的业务逻辑。端口 getInfo 提得到商品信息服务, 端口 dataAccess 描述了对于数据访问的需求。
- (7) 部件 CommentModel, 主要功能是执行评论相关的业务逻辑。端口 comment 提供了发布评论的服务, 端口 dataAccess 描述了对于数据访问的需求。
- (8) 部件 buyModel, 主要功能是执行购买商品相关的业务逻辑。端口 buyCmdt 负责提供购买相关服务, 端口 dataAccess 描述了对于数据访问的需求。
- (9) 部件 VirtualFileSystem, 主要功能是负责提供虚拟文件系统, 屏蔽数据的物理存储细节, 提供数据的统一存储接口, 以对异构的数据进行访问。端口 dataService 是提供了数据访问服务, 可以有多个实例。
- (10) 部件 DataCrawler, 主要功能是负责从其他信息源头网站爬取信息, 实现爬取细节的封装, 只根据需要提供预期结果。端口 getData 负责提供商品数据服务。
- (11) 部件 LoadBalance, 主要功能是负责进行负载均衡, 即把对服务器的请求和访问进行分流和控制处理, 分发到不同存储节点, 降低单点负载, 提高并发性和服务器可靠性。端口 arrange 负责提供分派访问请求的服务, 端口 forward 负责将请求转发, 端口 reportStatus 负责报告各个服务器当前运行状况。
- (12) 部件 ErrorLog, 主要功能是负责对各类系统错误进行集中记录。端口 record 负责记录错误日志, 可以有多个实例。端口 look 提供日志查询服务。
- (13) 部件 KeywordDetection, 主要功能是负责进行特定词汇 (包括敏感词) 做为关键字的检测。端口 dectect 提供检测服务, 可以有多个实例; 端口 updateKeywords 负责进行关键字管理, 端口 declare 描述了其要广播的事件类型, announce 负责广播事件出现 (如某关键词出现)
- (14) 部件 SynonymsDetection, 主要功能是负责同义词检测和处理。端口 dectect 提供检测服务, 可以有多个实例; 端口 removeSynonyms 和 addSynonyms 负责添加和删除同义词组。
- (15) 部件 AttackDetection, 主要功能是负责攻击检测和告知。端口 declare 负责描述其要广播的事件类型, 端口 announce 负责广播事件出现, 端口 register 负责注册事件 (如特定 IP 特定访问次数作为一个事件), 端口 calledProcedure 描述了相关事件发生后被调用的服务。
- (16) 部件 ResultFilter, 主要功能是负责对查询结果进行过滤, 过滤规则可以是去掉特定的结果。端口 filter 提供过滤查询结果的服务, 可以有多个实例。端口 modifyFilterItem 负责对过滤条件进行修改。
- (17) 部件 ContactList, 主要功能是负责维护系统的联系人和对应的事件列表, 使得事件可以及时告诉相关人员。端口 updateContact 负责对联系人和事件对的添加和移除, 端口 register 声明了需要监听的事件并进行注册, 端口 calledProcedure 描述了相关事件发生之后被调用的服务 (即通知对应的联系人)。
- (18) 部件 ResultSorter, 主要功能是负责对查询结果进行排序。端口 sort 提供排序服务, 可以有多个实例; 端口 modifySortRule 负责对排序规则进行修改
- (19) 部件 DataMng, 主要功能是负责对服务器数据的备份和恢复以及相关的维护工作, 提高数据存储的可靠性。端口 backup 负责数据备份, 端口 recovery 负责数据恢复。
- (20) 部件 Monitor, 主要功能是监控系统的运行状态, 发现异常并广播。端口 monitorStatus 负责查询、接收和处理各个服务器所告知的正常运行信息, 端口 declare 描述了其要广播的事件类型, 端口 announce 广播服务器状态异常相关事件。
- (21) 部件 UserMng, 主要功能是负责对用户的管理, 端口 declare 描述了其要广播的事件类型 (比如达到“水军”标准的用户), 端口 announce 提供了事件广播服务, 端口 register 负责向注册



事件（比如向敏感词检测模块注册以累计敏感词数目），端口 calledProcedure 描述了相关事件发生后调用的服务。

- (22) 部件 AdminInterface，主要功能是为管理员提供操作接口。端口 mngServer 提供系统服务员的管理接口，端口 mngData 提供了数据管理员的管理接口。
- (23) 部件 Firewalls，主要功能是进行入侵检测和请求过滤，端口 attackDeffense 负责进行入侵检测，端口 declare 和 announce 负责将入侵事件广播，端口 forward 负责将过滤后的请求传入。
- (24) 连接件 RPC 描述了远程调用过程调用机制，角色 caller 调用了另一个角色 callee 的程序。特征 remote=true 说明程序调用机制是能够实现远程过程调用的。
- (25) 连接件 Message，描述了一种消息收发机制，角色 sender 和 receiver 之间相互发送消息，属性 lifetime=5min 表示消息的生命周期是 5 分钟。
- (26) 连接件 ShareData，描述了共享数据机制，角色 initializer 将设置的数据共享给一个或多个其他角色 user 使用。
- (27) 连接件 HttpRequest，描述了描述了 http 超文本通信协议，用于客户端浏览器和服务器通信。以两个角色 request-response 请求响应的方式相互通信。
- (28) 连接件 ProceduralCall，描述了常见的程序调用机制，角色 caller 调用另一个角色 callee 的程序。
- (29) 连接件 Adaptor，描述了描述了配适器机制，提供对不同标准和规格匹配为统一标准的转换处理。角色 adaptee 会被转化为和 adapter 具有统一的接口表现的部件。
- (30) 连接件 EventRouter，描述了事件路由机制。角色 publisher 描述了事件路由为时间声明者提供的交互协议；角色 subscribe 描述恶劣事件路由为时间注册者提供的交互协议。角色 announce 描述了事件路由为时间广播者提供的交互协议；角色 listener 描述了事件路由对事件接收者的行为期望。

4.2.2 配置描述

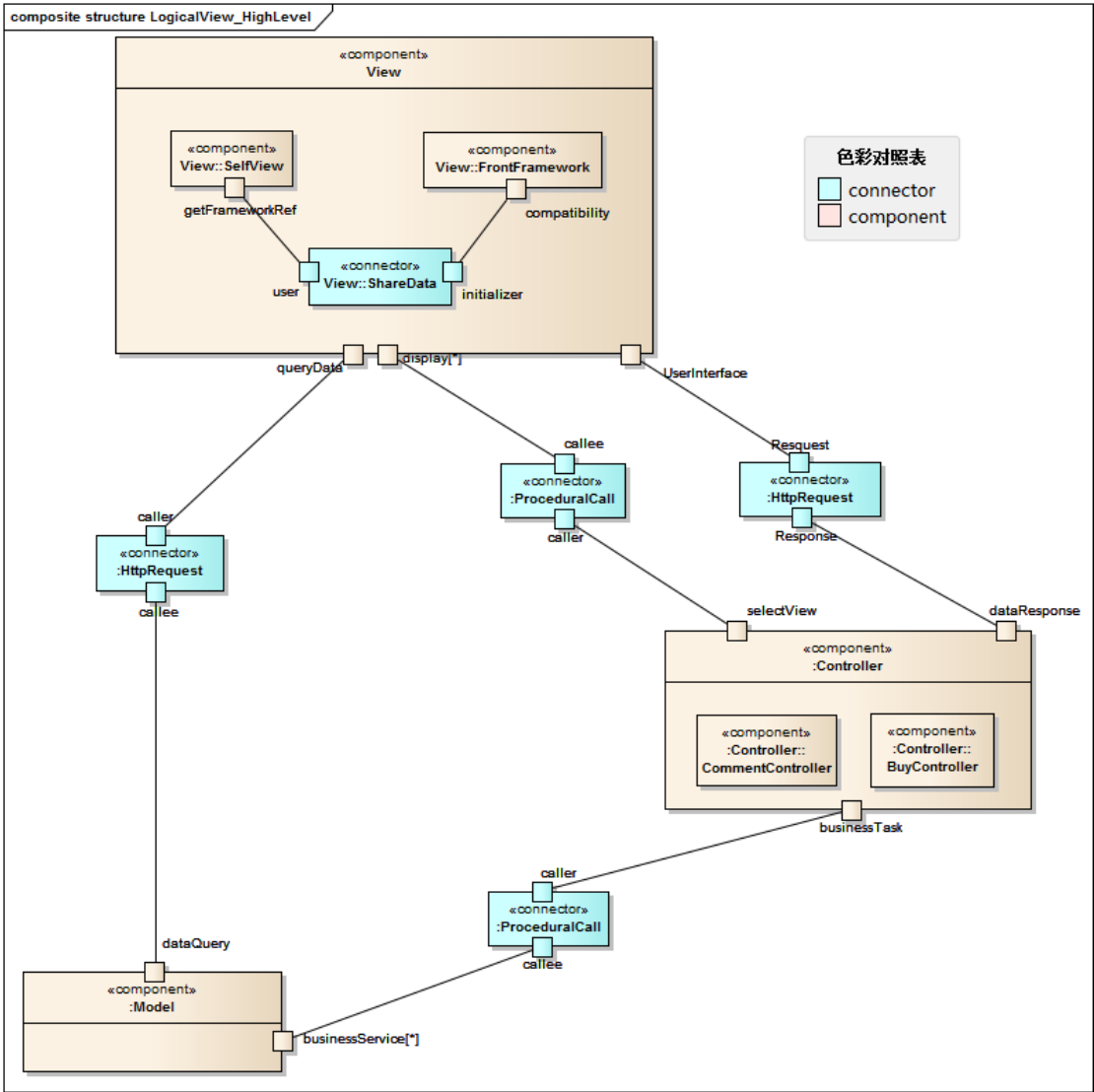


图 5 比价系统配置高层描述

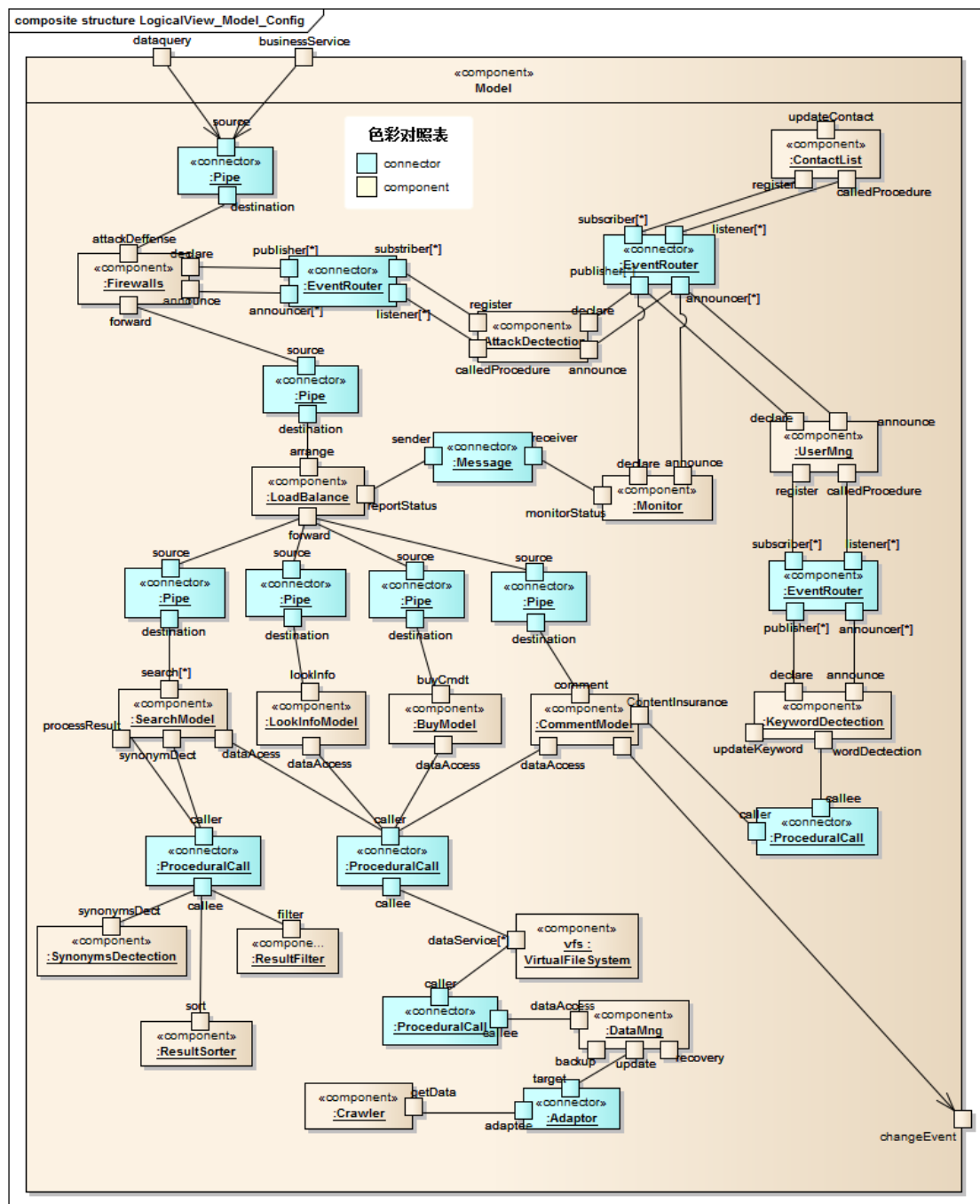


图 6 比价系统 Model 复合型部件的配置描述

4.3 开发视图

4.3.1 模块组织

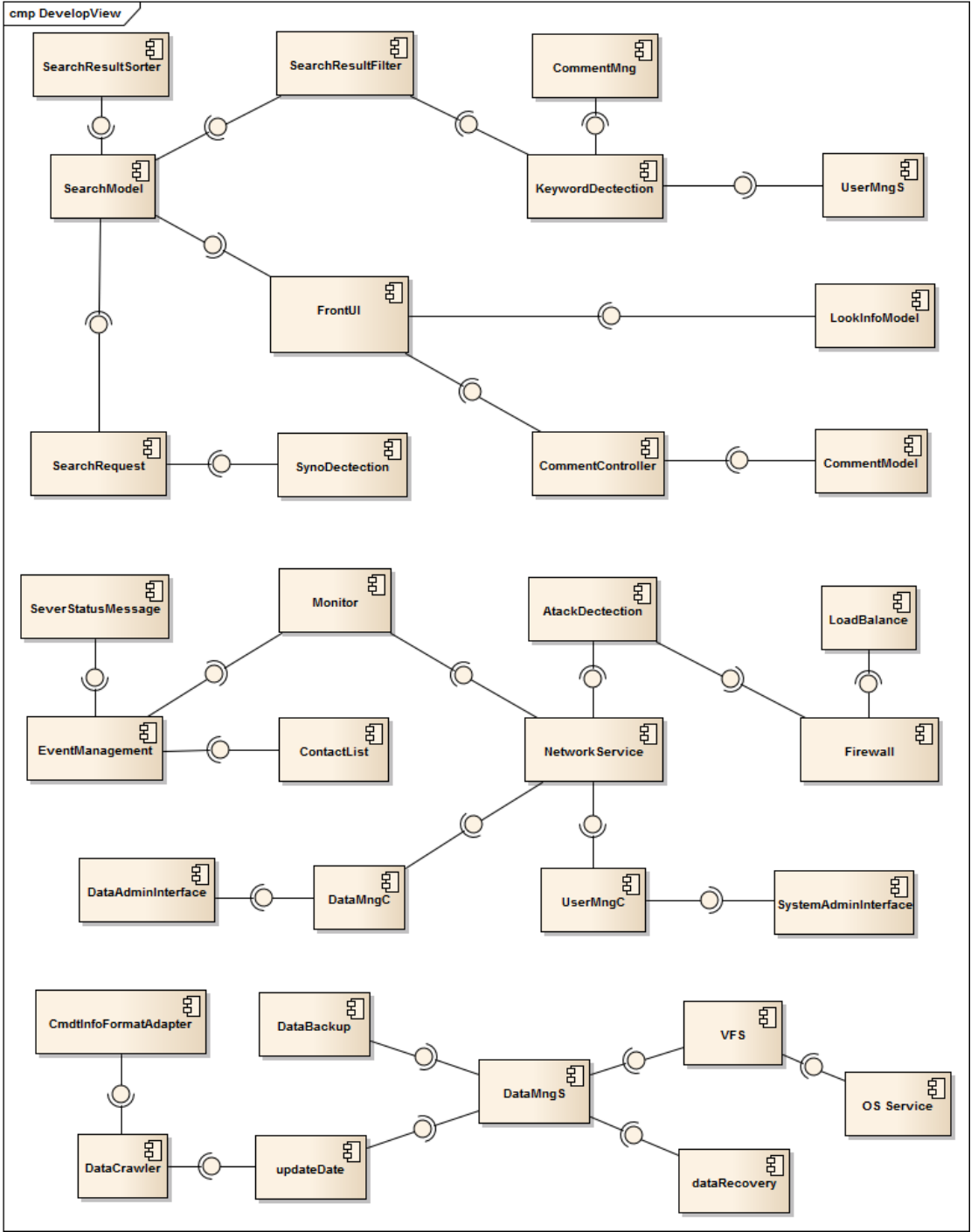


图 7 比价系统的模块组织

4.3.2 子系统组织

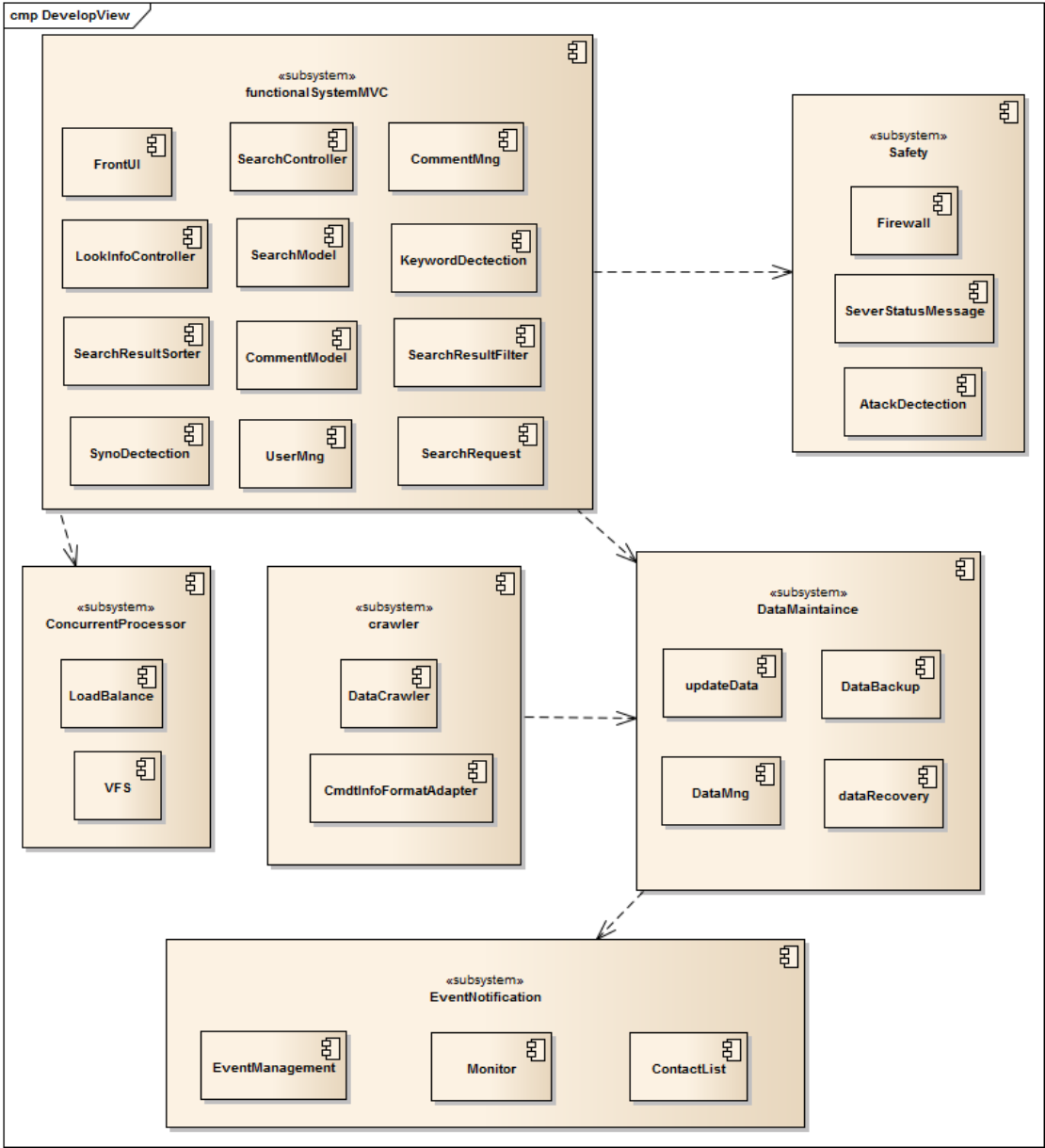


图 8 比价系统的子系统组织

4.3.3 分层结构

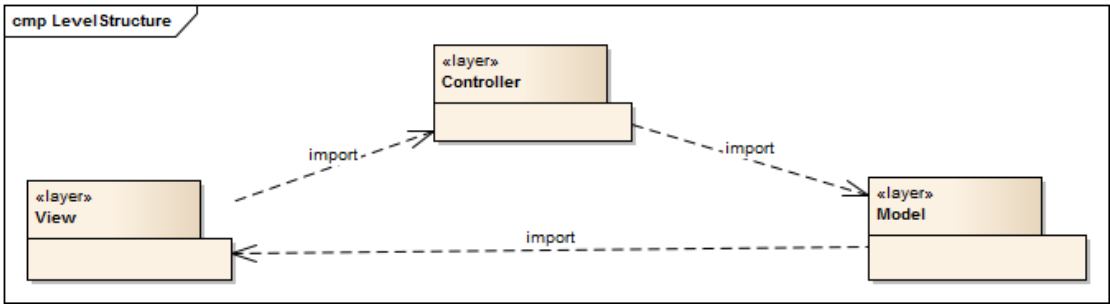


图 9 比价系统服务器端功能操作子系统的分层结构

4.4 进程视图

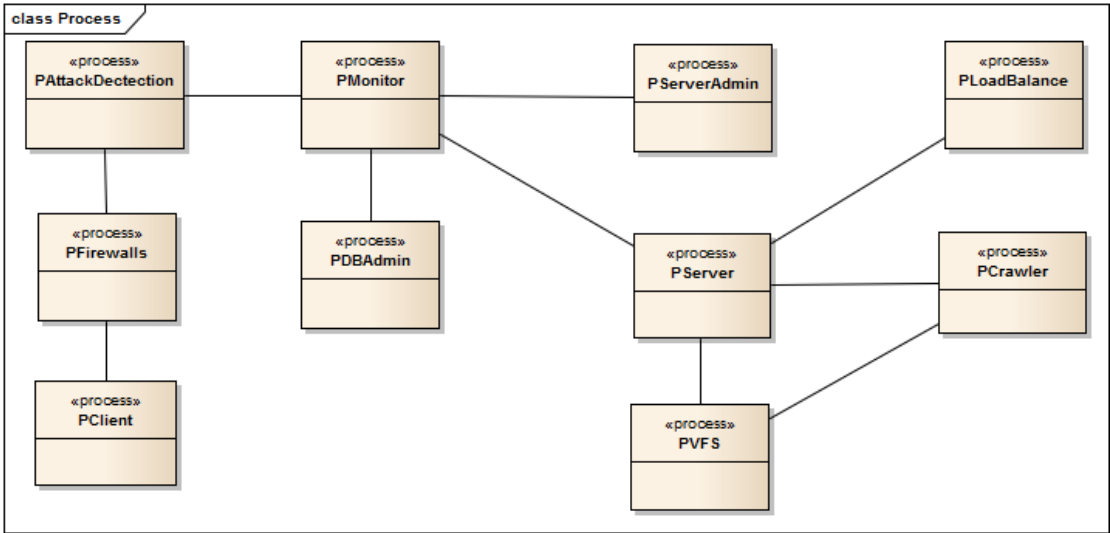


图 10 比价系统的进程描述图

整个比价系统被设计为 10 类进程，它们与逻辑视图元素的映射关系如下表：

进程	部件或连接件
PAttackDetection	AttackDetection, EventRouter
PMonitor	Monitor, Message, EventRouter
PServerAdmin	AdminInterface
PLoadBalance	LoadBalane
PFirewalls	Firewalls, EventRouter
PDBAdmin	AdminInterface
PServer	View, CommentController, BuyController, FrontFramework, SearchModel, LookInfoModel, CommentModel, BuyModel, ContactList, ResultSorter, ResultFilter, HttpRequet,Monitor
PClient	DataMng, UserMng

PVFS	VirtualFileSystem
PCrawler	DataCrawler, Adapter

## 4.5 部署视图

### 4.5.1 UML 部署图描述

需求&约束	AR6. 搜索结果可调整 DC10. 搜索结果排序显示 DC11. 排序标准可以灵活变更	
候选方案	CS1.1	应用面向对象编程思想，将搜索结果进行封装，利用编程语言提供的类的特性，对 sort 方法进行重载，用多态机制实现变更。
	CS1.2	采用策略模式，对排序策略进行封装，建立排序器，新的排序标准可以用新的 sorter 来实现。
选择选项	CS1.2 结合 CS1.1	
选择理由	<ol style="list-style-type: none"> <li>为了解决 DC10 和 DC11，主要选择使用 CS1.2，即把排序功能独立出来，基于策略模式进行实现。这样可以避免为了排序编写过多的结果类以及过深的继承树，同时也是从一个更高的抽象层次上思考问题，把排序看成对结果进行处理，这样比如屏蔽等处理也可以用策略模式进行处理，处理机制统一，符合完整一致性的思想。</li> <li>对查询结果的处理可以使用组合形式来处理，可以用几个单独的子策略构成复杂的结果处理方案，而不用为此重新编写类并将其置于继承树的合适位置。</li> <li>对搜索结果进行封装，这是基于信息隐藏的原则，隐藏结果的数据存储方式，并且便于网络通信中的加密和分级传输等处理。</li> </ol>	
影响	逻辑视图，开发视图	
详细设计约束	<ol style="list-style-type: none"> <li>封装排序规则，使用策略模式</li> <li>封装搜索结果，隐藏数据类型</li> </ol>	

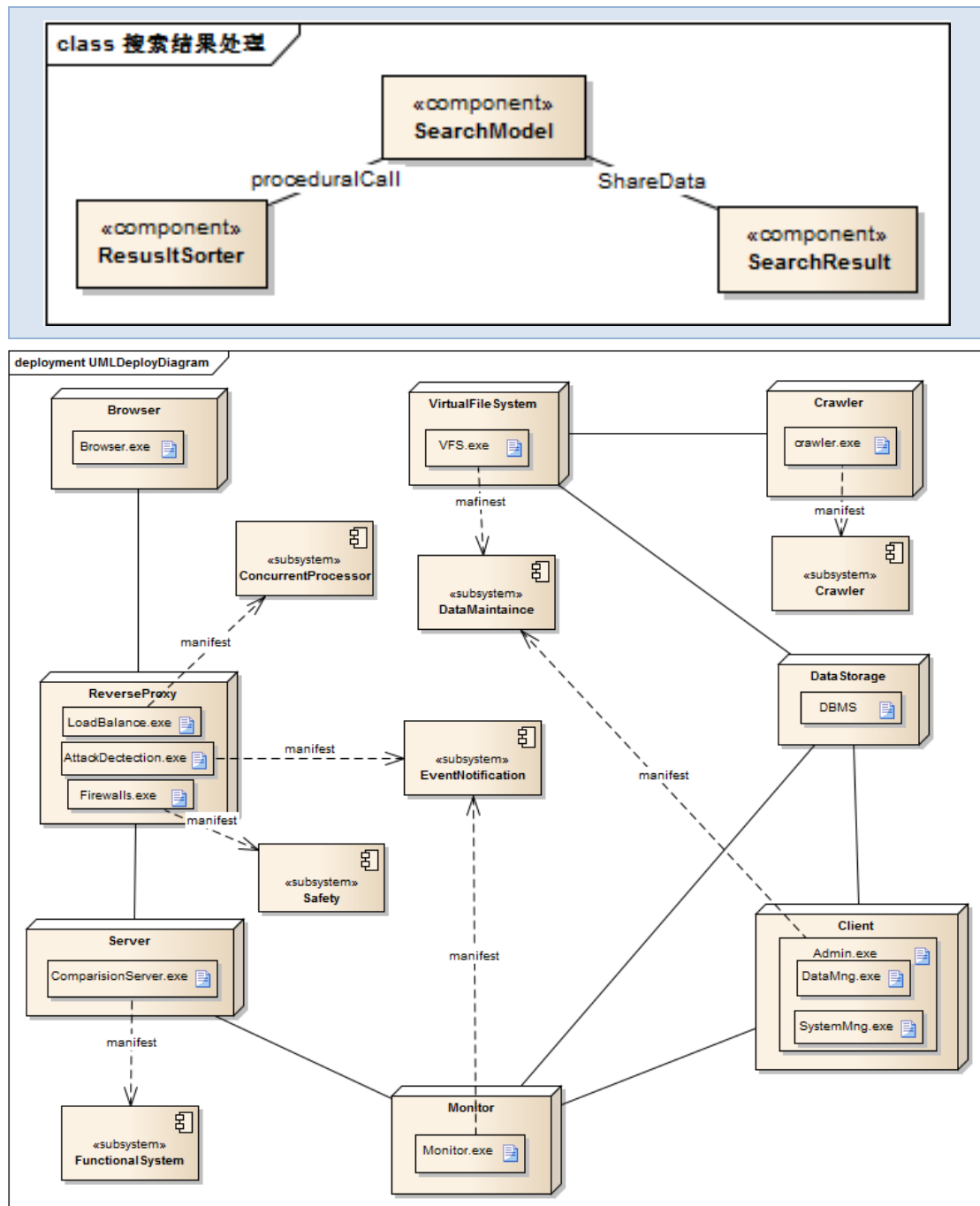


图 11 比价系统的 UML 部署图描述

## 5. 设计决策

### 5.1 设计决策 1



## 5.2 设计决策 2

需求&约束	AR6. 搜索结果可调整 DC12. 可以屏蔽特定商品	
候选方案	CS2.1	将是否被屏蔽做为商品的一个属性进行存储，得到 SearchResult 时，之间判断此属性
	CS2.2	建立过滤器，过滤掉符合条件的信息。过滤条件的灵活性结合过滤器和配置文件实现。
选择选项	CS2	
选择理由	<div>1. CS2.1 可以减少代码，只需要给管理员改变属性的接口即可。但是会对数据存储带来更大的压力。难以根据一定的规则批量选择屏蔽商品。</div> <div>2. CS2.2 较为复杂，但是可以提供更大的灵活性，并且不会对数据存储产生更大的压力。可以提供多种过滤条件，比如对特定类型商品、对含有特定关键字、对特定商家的产品、对特定产地的产品等等。只需要定义相关的配置文件格式，编写相应的过滤器即可实现灵活的扩展和变化，可以适应更复杂场景的需要。</div> <div>3. CS2 可以与设计决策 1 选择的策略模式在更高的抽象层次上实现统一，都用策略模式处理，符合完整一致性的要求。</div>	
影响	逻辑视图、开发视图	
详细设计约束	<div>1. 封装屏蔽规则为过滤器，使用策略模式实现过滤规则</div> <div>2. 使用文件对过滤器进行辅助，避免程序中的硬数据，实现灵活性。</div>	

class 搜索结果屏蔽

«component»  
SearchModel

— ProcedureCall —

«component»  
ResultFilter

----->

«File»  
[[formatted]  
Filter Condition]

## 5.3 设计决策 3

需求&约束	AR7. 用户管理方便 DC13. 定义水军的判读规则，保持规则容易变更 DC14. 发现水军，在 10 分钟内通知系统管理员	
候选方案	DC3.1	将敏感词次数设计为用户的属性，每次发表评论时 SearchModel 更新敏感词次数进行并判断，如果是水军，则 SearchModel 通知管理员。
	DC3.2	封装水军检查作为单独的模块。由 SearchModel 使用水军检测模块判断用户是否为水军，并通知系统管理员。
	DC3.3	封装水军检查作为单独的模块，并作为事件封装。此事件对维护用户信息的模块进行注册，判断用户信息是否满足某种标准，如果事件发生，则依据隐式调用规则，触发事件后调用已经注册的通知系统管理员的服务。

选择选项	DC3.3
选择理由	1. 封装水军检测作为单独的模块，可以实现 DC13 2. 使用事件机制（隐式调用）可以实现 DC14
影响	逻辑视图、开发视图
详细设计约束	1. 封装水军检测规则作为事件 2. 事件使用 Observer 模式实现

**class 水军检测**

```

classDiagram
    class WaterArmyDetection {
        «component» UserMng
        «component» Admin
    }
    UserMng --> Admin : EventRouter

```

## 5.4 设计决策 4

需求&约束	AR8. 商家关注信息推送 DC16. 允许商家定制词汇表，并允许商家自定义发送频率和发送标准	
候选方案	CS4.1	使用 observer 模式实现关注信息推送，对同一种类型的事件定义一种文件格式，统一的描述触发条件和被通知者之间的对应关系。
	CS4.2	将关键字检测单独封装
选择选项	CS4.1 和 CS4.2	
选择理由	1. 与设计决策 3 可以通过事件机制形成统一的解决方案，实现复用性和一致性。 2. 对一种类型的事件定义一种文件格式，可以有效实现商家关注信息的可调整性和可定制性，并且除非出现一种全新的（无法用已有事件类型表示的）时间类型，可以不用对系统进行变更。 3. 基于功能的内聚性原则和模块化的思想将关键字检测封装	
影响	逻辑视图、开发视图	
详细设计约束	1. 使用 Observer 模式通知商家 2. 新加入不同商家和不同的关键词通过变更文件实现	

**class 商家关注**

```

classDiagram
    class MerchantAttention {
        «component» CommentModel
        «component» KeywordDetection
        «component» EventRouter
        «component» ContactList
    }
    CommentModel --> KeywordDetection : ProcedureCall
    KeywordDetection --> EventRouter
    ContactList --> EventRouter
    ContactList -.-> File : {Event-Contact[List]}

```

## 5.5 设计决策 5

需求&约束	AR9. 实现自主的信息获取 DC17. 使用爬虫技术获取网站商品和评论信息 DC18. 允许自定义爬取网站列表 DC19. 使用多种爬虫技术和策略进行替换以应对可能的封锁	
候选方案	CS5.1	对爬取数据进行封装，作为单独的模块。网站列表通过定义配置文件，作为爬虫模块的一个输入实现。
选择选项	CS5.1	
选择理由	可以满足 DC17-19，配置文件可以灵活定义爬取的网站列表。多种爬虫技术的替换由于对爬取进行封装可以局部化，在模块内部实现，直接使用策略模式进行替换。	
影响	逻辑视图、开发视图、进程视图、部署视图	
详细设计约束	1. 封装数据爬取做为独立的子模块, 提供具有高内聚性的功能子系统和对外接口。 2. 模块内部爬虫技术使用策略模式 3. 爬取网站源头使用配置文件实现	

class 数据爬虫

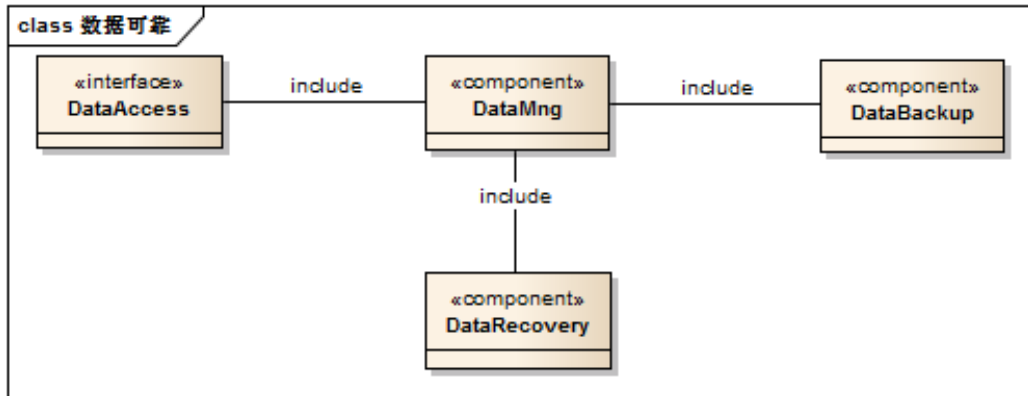
«component»  
Crawler

«File»  
{[input]  
web URLs}

## 5.6 设计决策 6

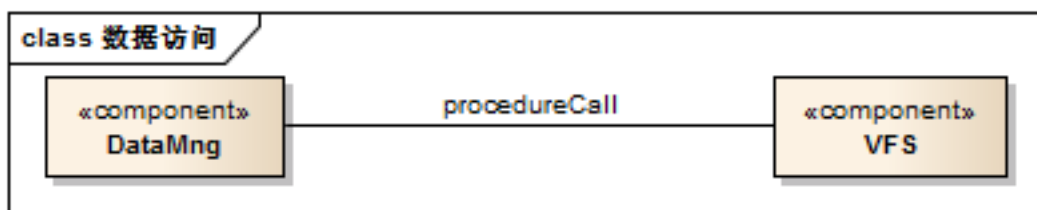
需求&约束	AR10 实现数据存储的高可靠性 DC20. 存储数据出现故障时网站能够正常工作 DC21. 数据恢复时间小于 20 小时 AR13. 保证用户操作的一致性和流畅性 DC25. 后台信息变化不影响用户的使用 AR15. 提供大数据量存储 DC29. 保证最大能够存储一千万商品的相关信息	
候选方案	CS6.1	使用热备份，分布式数据存储，一个物理节点故障，系统自动选择新节点启用。自动数据恢复。
	CS6.2	使用冷备份，计算结果分为两个部分，只有一个处于运行状态，另一个作为备份，故障时，人工切换到备份数据。人工恢复数据。
选择选项	CS6.1	
选择理由	1. 使用热备份可以使用户基本完全感受不到数据故障，更符合 DC20 的要求 2. 自动化程度高，减少管理员的维护成本	

	3. 可以使用现成的分布式数据架构和系统
	4. 可以方便对数据量进行扩展,以实现对大数据存储的支持,即辅助 AR15 的实现
影响	逻辑视图、开发视图、进程视图、部署视图
新约束	CA1 数据一致性更新与数据访问分派
详细设计约束	1. 使用分布式数据存储,屏蔽物理节点差异和异构事实 2. 对数据访问进行封装,使用统一的接口来访问



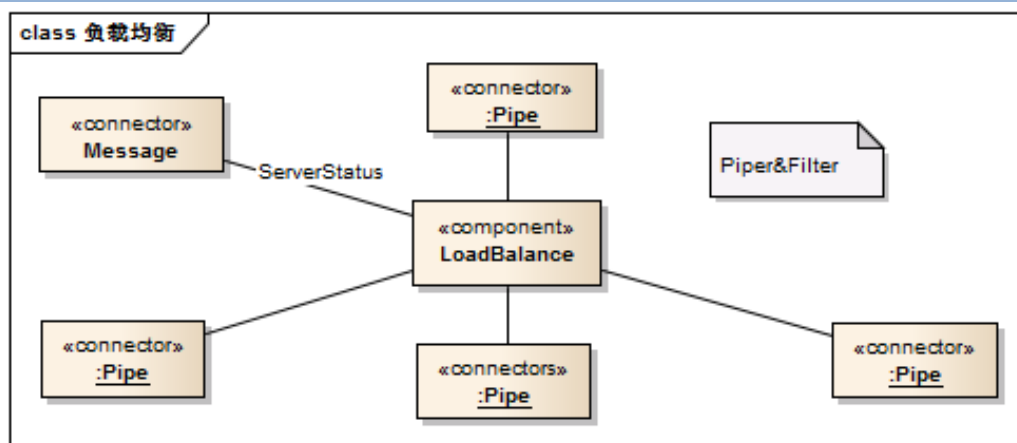
## 5.7 设计决策 7

需求&约束	AR10 实现数据存储的高可靠性 CA1. 数据一致性更新与数据访问分派
候选方案	CS7.1 1. 虚拟化,开发独立的 VirtualFileSystem 管理文件,提供对数据文件的读写,屏蔽物理异构。 2. 定时备份数据用于故障恢复,故障恢复自动进行。
选择选项	CS7.1
选择理由	1. 基于信息隐藏的原则,封装物理文件存储,提供统一的虚拟粘合层,对异构的数据进行存储 2. VFS 是一套比较成熟的有着容错性和健壮性的系统(被淘宝应用),对于物理节点的故障有着比较好的处理能力
影响	逻辑视图、开发视图、进程视图、部署视图
详细设计约束	1. 系统集成 VFS 2. 数据管理模块对故障备份和恢复进行封装



## 5.8 设计决策 8

需求&约束	AR11. 实现服务器高可靠性 DC22. 服务器出现故障时网站能够正常工作 DC23. 故障发现时间小于 1 小时，恢复时间小于 1 天 AR14. 达到高峰吞吐量要求 DC26. 保证五百万用户的并发访问量不会使得系统的故障率上升 DC27. 提供一千万的极限并发访问量 DC28. 允许并发用户超过五百万之后响应时间下降 AR16. 保证存取效率 DC30. 反应时间理想是 2s 以内，能够接受的上限是 5s	
候选方案	CS8.1	硬件负载均衡，直接在外部网络和服务器之间安装负载均衡器，设备专用性能优越。
	CS8.2	软件负载均衡，在一台或多台服务器的操作系统上安装负载均衡软件实现。
	CS8.3	使用 ping/echo 验证服务器状态，进行故障检测
	CS8.4	使用心跳机制，即各个服务器定期发送心跳，进行故障检测
	CS8.5	由于负载均衡系统需要将请求分派，所以会随时得到各个服务器的状态，则依据此使用 Message 通信机制进行故障检测。
选择选项	CS8.2, CS8.5	
选择理由	1. CS8.2 成本低，配置简单 2. CS8.2 使用灵活，根据分析，对此系统的使用以查看和检索为主，即对数据操作以查询为主，更新较少，可以据此对流量方案进行定制，保证查询效率并提高带宽的利用率。 3. CS8.2 可以满足后续更多的负载均衡需求 4. CS8.5 可以提供所有服务器的全貌特征，并且由于附带具体状态信息，比如当前负载，内存占用量等内容，可以方便管理员不仅仅进行故障恢复，还可以进行故障预防，提供了更高的灵活性。	
影响	逻辑视图、开发视图、进程视图、部署视图	
详细设计约束	1. 封装负载均衡，屏蔽实现细节 2. 负载均衡内部对数据查询和更新分别处理 3. 使用 Proxy 模式实现负载均衡（Reverse Proxy）	



## 5.9 设计决策 9

需求&约束	AR12. 搜索易用性高 DC24. 定义同义词表，显示对应搜索结果	
候选方案	CS9.1	对同义词检测进行封装，维护专门的同义词库，对同义词库进行封装。
选择选项	CS9.1	
选择理由	<div>1. 同义词检测单独作为一个模块，从逻辑上来说功能是功能内聚的。</div> <div>2. 有助于隔离主干程序的复杂度</div> <div>3. 对同义词库进行封装主要是出于信息隐藏的考虑，同义词库的存储方式可以视为一个需要隐藏的方式，对于文本检索和匹配的检索方式需要隐藏。</div> <div>4. 同义词库是系统无关的，可以采用任何其他的开源词库，并且可能具有复杂的学习算法。而与系统相关的具体同义词选择选项和补充措施则在同义词检测部件完成。</div>	
影响	逻辑视图、开发视图	
详细设计约束	<div>1. 封装同义词检测</div> <div>2. 封装同义词库</div>	

class 同义词检测

«component»  
SearchModel

ProcedureCall

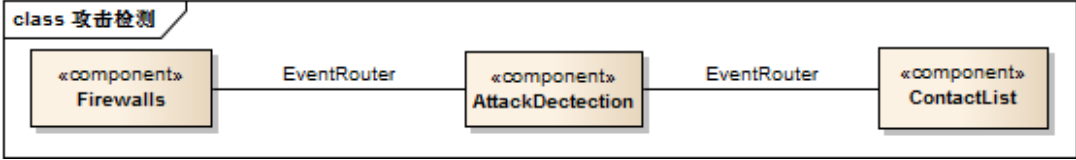
«component»  
SynonymDetection

ProcedureCall

«component»  
SynonymWordBag

## 5.10 设计决策 10

需求&约束	AR17. 实现入侵检测 DC31. 实现对短时间内频繁访问 IP 进行检测 DC32. 发现特定 IP 的访问 AR18. 实现隔离攻击 DC33. 允许定义入侵事件列表 DC34. 针对事件提供灵活的屏蔽或者拦截措施 AR19. 设计方案提供对更多的入侵检测条件进行扩展 DC36. 增加条件判断和通知人员的代价小于 0.5 人月	
候选方案	CS10.1	使用防火墙，进行拦截和请求过滤
	CS10.2	封装攻击检测，使用事件机制
	CS10.3	使用事件机制通知系统管理员
选择选项	CS10.1, CS10.2, CS10.3	
选择理由	1. CS10.1-10.3 可以实现 AR17-AR19	

	2. 事件机制与前面的几个决策在高层抽象上是一致的,体现了完整一致性
影响	逻辑视图、开发视图、进程视图、部署视图
详细设计约束	1. 攻击检测标准可替换
 <pre> classDiagram     class AttackDetection {         &lt;&lt;component&gt;&gt;     }     class Firewalls {         &lt;&lt;component&gt;&gt;     }     class ContactList {         &lt;&lt;component&gt;&gt;     }     Firewalls --&gt; AttackDetection : EventRouter     AttackDetection --&gt; ContactList : EventRouter         </pre>	

## 5.11 设计决策 11

需求&约束	AR21. 兼容各个网站 DC37. 兼容从各个网站获取信息的不一致，至少支持两个网站 DC38. 更开放的语言支持，比如 amazon.com 等 DC39. 能够在 0.5 人月内实现对新的网站的格式的兼容	
候选方案	CS11.1	使用配适器模式解决格式兼容问题
	CS11.2	自定义词汇表实现多语言支持
	CS11.3	使用翻译平台 API 实现多语言支持
选择选项	CS11.1, CS11.3	
选择理由	1. 配适器模式是将一种接口转化成另一种接口,使得不能兼容的接口可以一起工作的模式。很适合用来满足 AR21 2. 配适器模式是 GOF 的设计模式中所提出来的，具有可复用性和可修改性好的特点 3. 选择 CS11.3 主要是为了节省成本，降低系统负责度，促进快速开发，并且当前翻译平台的结果要比自定义的结果强。	
影响	逻辑视图、开发视图	
详细设计约束	1. 使用配适器模式实现格式兼容	

class 数据格式

«component»  
Crawler

Adaptor

«component»  
DataMng

RPC

«component»  
TranslationAPI

## 5.12 设计决策 12

需求&约束	AR22. 支持多种浏览器 DC40. 兼容 IE8 以后的各个版本 DC41. 兼容 chrome9.0 浏览器以及以后的各个版本 DC42. 兼容 Firefox10.0 浏览器以及以后的各个版本
-------	---

候选方案	CS12.1	自己实现兼容性功能，主要包括判断和纠错处理，进行封装处理
	CS12.2	使用前端框架解决
选择选项	CS12.2	
选择理由	1. 节省成本 2. 只要求 IE8 以后的版本，有很多前端框架可以选择，并且错误率很低。	
影响	逻辑视图、开发视图	
详细设计约束	1. 使用 jQuery、bootstrap 等前端框架实现浏览器兼容	

class 浏览器兼容

«component»  
View

shareData

«component»  
FrontFramework

## 5.12 设计决策 13

需求&约束	AR23. 保证评论符合相关国家法律规范 DC43. 定义敏感词表，主动判定敏感词 DC44. 发现敏感词需要在 1min 内通知系统管理员 DC45. 允许系统管理员修改删除评论	
候选方案	CS13.1	封装关键词检测模块，敏感词表使用文件实现，并且运用 Observer 模式确保系统管理员被通知到。
选择选项	CS13.1	
选择理由	1. CS13.1 可以实现 DC43-DC45 2. CS13.1 与决策 3、4 在机制上都使用事件机制，体现了系统的完整一致性 3. 可以服用	
影响	逻辑视图、开发视图	
详细设计约束	使用 Observer 模式，通知系统管理员	

class 评论检测更正

«component»  
CommentModel

ProcedureCall

«component»  
KeywordDetection

EventRouter

«component»  
Contact  
- Admin :

## 5.14 设计决策 14

需求&约束	AR23. 设计方案允许快速开发 AR25. 有效规避开发人员的技能劣势
-------	---



	AR26. 支持多版本发布，快速迭代，持续演进	
候选方案	CS14.1	使用分层风格
	CS14.2	使用 MVC 风格
选择选项	CS14.2	
选择理由	1. CS14.1 和 14.2 都体现了模块化思想，可以实现 AR23 2. CS14.2 更适合 web 开发，与系统情况相符合	
影响	逻辑视图、开发视图	
详细设计约束	1. 符合 MVC 风格	

class MVC

```
classDiagram
    package MVC {
        <<component...>> Model
        <<component>> View
        <<component>> Controller
    }
    Model ..> View : PC
    Model ..> Controller : PC
    View ..> Controller : PC
    Controller ..> View : PC
```

## 5.15 设计决策 15

需求&约束	AR11. 实现服务器高可靠性 DC21. 故障发现时间小于 1 小时，恢复时间小于一天	
候选方案	CS15.1	使用高级语言提供的异常机制，自定义的异常和捕捉机制
	CS15.2	使用错误日志，详细记录事故现场，建立容易的错误和记录机制。
选择选项	CS15.2	
选择理由	1. 使用异常机制,但是不对异常做统一处理,而是自定义的错误日志部件,这样灵活性更高 2. 可以实现对错误的集中记录和对事故现场的保存，方便后续分析和调试。	
影响	逻辑视图、开发视图	
详细设计约束	1. 实现自定义的错误日志，建立统一的错误处理和事故现场保存机制。	

class 错误日志

«component»  
AiBiJia System

ProcedureCall

«component»  
ErrorLog

# 6. 最终高层结构

## 6.1 逻辑视图

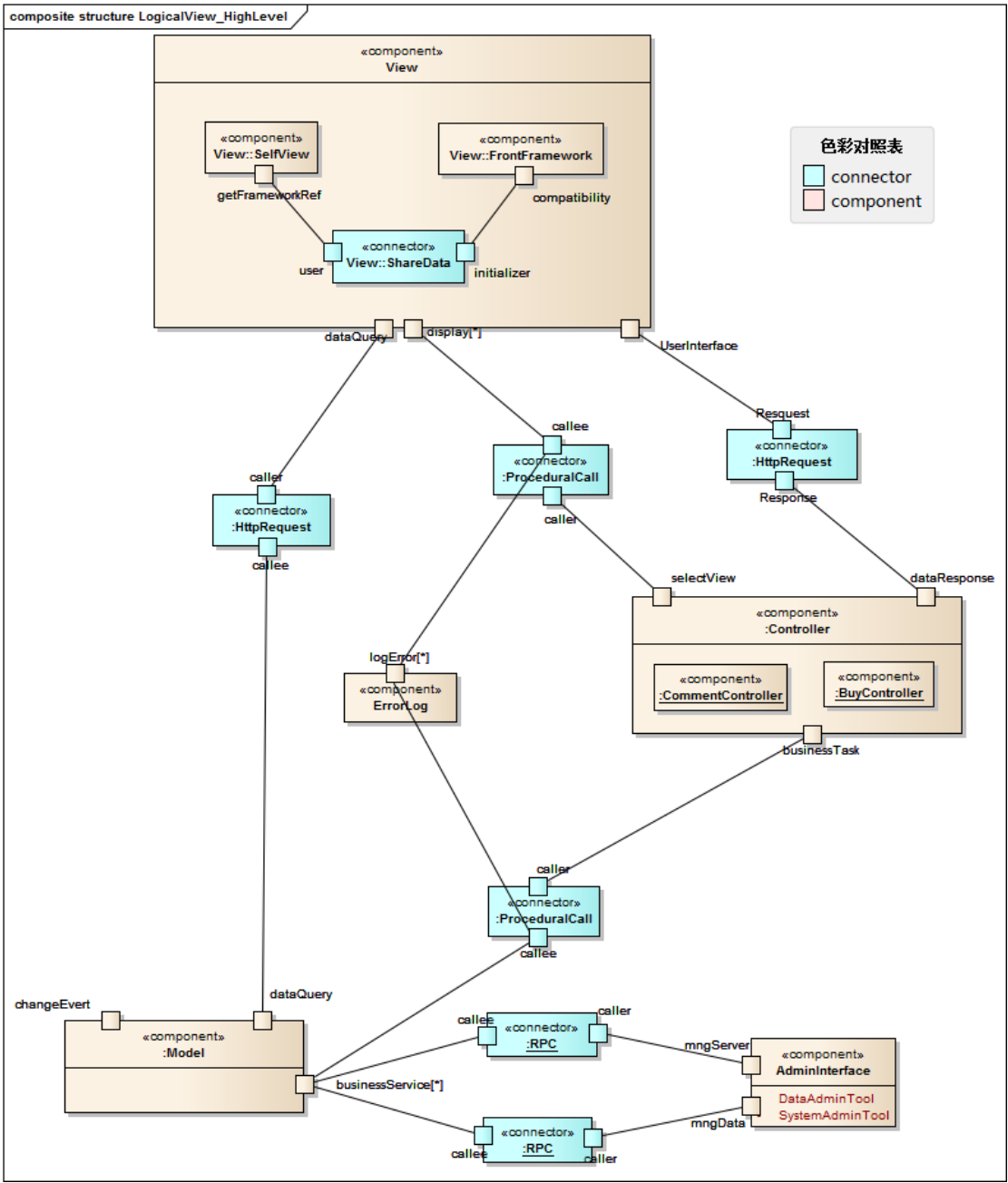


图 12 比价系统配置的高层描述



6.2 开发视图

6.2.1 模块组织

6.2.1.1 模块描述

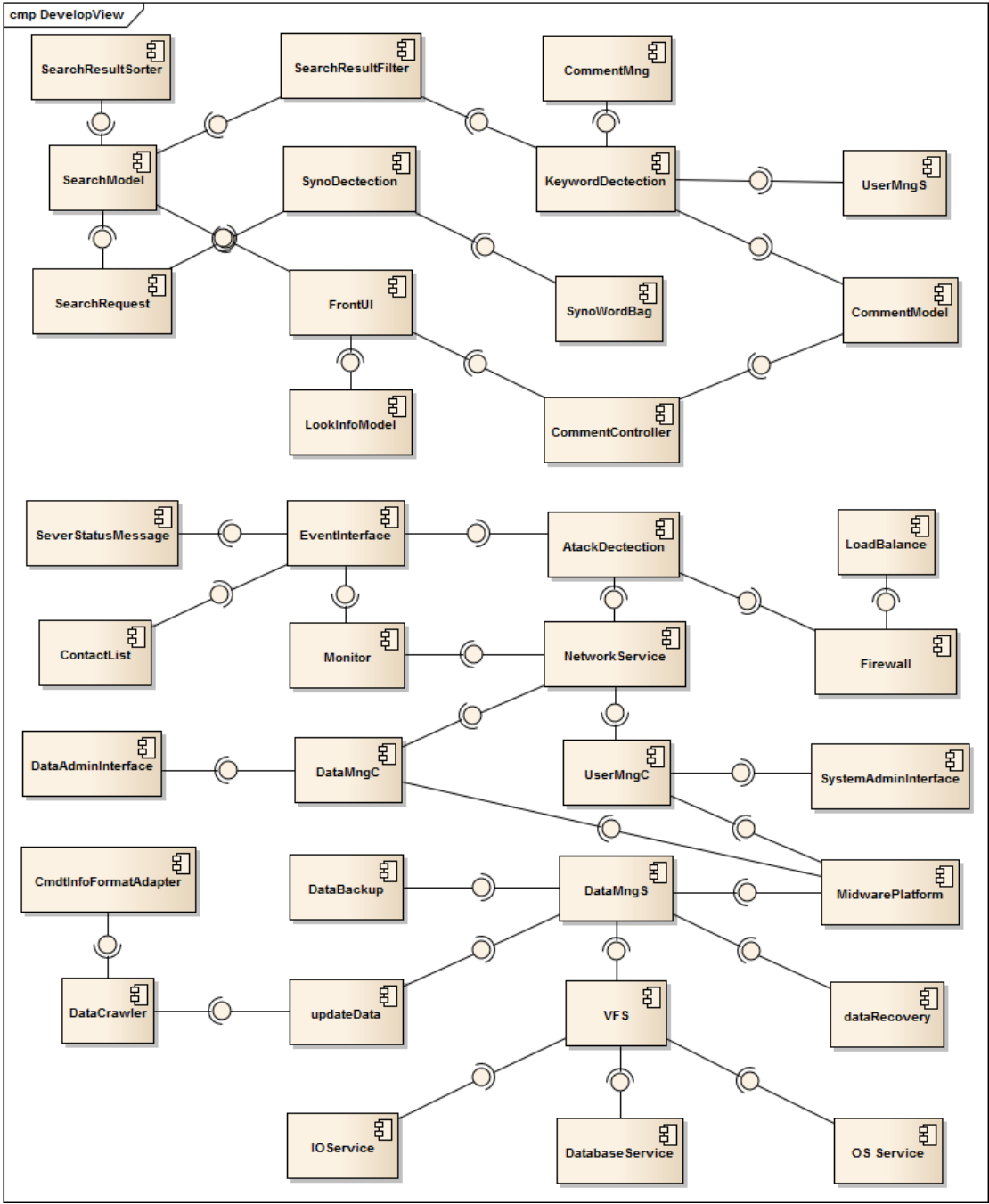


图 14 比价系统的模块组织

逻辑视图与开发视图之间的映射关系如下表所示：

部件/连接件	实现模块/构件
View	FrontUI
commentController	CommentController
FrontFrmework	FrontUI
SearchModel	SearchModel
LookInfoModel	LookInfoModel
CommentModel	CommentModel
VirtualFileSystem	VFS
DataCrawler	DataCrawler
TranslationAPI	TransLation
LoadBalance	LoadBalance
ErrorLog	ErrorLog
KeywordDetection	KeywordDetection
SynonymsDetection	SynonymsDetection
SynonymsWordBag	SynoWordBag
AttackDetection	AttackDetection
ResultFilter	ResultFilter
ContactList	ContactList
ResultSorter	ResultSorter
DataMng	DataMngC, DataMngS, DataBackup, DataRecovery
Monitor	Monitor
UserMng	UserMngC, UserMngS
AdminInterface	DataAdminInterface, SystemAdminInterface
Firewalls	Firewalls
RPC	MiddlewarePlatform
PC	无
Message	SeverStatusMessage
Adaptor	FormatAdaptor
EventRouter	EventInterface
HttpRequest	NetworkService
ShareData	IOService, DatabaseService

### 6.2.1.2 接口定义

比价系统各个构件（模块）之间的接口描述如下：

ID	SSort		
模块名	SearchResultSorter		
接口ID	sort	功能描述	对搜索结果进行排序
		语法	SearchResult sort(SearchResult sr0);
		前置条件	无

		后置条件	无
		不变量	InputSearchResult.elemSum=outputSearchResult.elemSum
System.sort(Object);			

ID		SFilter	
模块名		SearchResultFilter	
接口ID	filter	功能描述	过滤掉部分搜索结果
		语法	SearchResult filter(SearchResult sr0);
		前置条件	无
		后置条件	无
		不变量	结果中元素顺序不能发生变化
SearchResult.Delete(Element);			

ID		Cmng	
模块名		CommentMng	
接口 ID	mod	功能描述	修改评论
		语法	Comment modifyComment(Comment cm);
		前置条件	评论被检测出含有敏感词汇
		后置条件	发表修改后评论
		不变量	无
接口 ID	del	功能描述	删除评论
		语法	Boolean deleteComment(Comment cm);
		前置条件	评论被检测出含有敏感词汇
		后置条件	删除评论
		不变量	无
需求接口			
Comment.ModifyContent;			

ID		SMod	
模块名		searchModel	
接口ID	search	功能描述	执行搜索
		语法	SearchResult search(SearchRequest srq);
		前置条件	无
		后置条件	显示搜索结果
		不变量	无
SSort.sort SFilter.filter			

ID		KDtct	
模块名		KeywordDetection	
接口ID	detect	功能描述	从评论中检测关键字
		语法	Boolean detect(Text, KeyList);
		前置条件	关键词列表以及初始化
		后置条件	无
		不变量	无
System.String			

ID		UmngS	
模块名		UserMngS	
接口 ID	noTalk	功能描述	禁言特定用户
		语法	Void enableTalk(User user);
		前置条件	用户被判定为水军
		后置条件	用户不能再发表评论
		不变量	无
接口 ID	del	功能描述	删除特定用户
		语法	Void deleteUser(User user);
		前置条件	满足删除用户条件，具有管理员权限执行
		后置条件	数据库中用户的历史数据需要保留
		不变量	无
需求接口			
NetworkService			

ID		FUI	
模块名		FrontUI	
接口ID	dsplay	功能描述	显示信息给使用浏览器的用户
		语法	Void display(Info);
		前置条件	无
		后置条件	无
		不变量	只显示，不改变信息内容
JQuery&Bootstrap			
CCtrl.comment			
SMod.search			
LkInfoMod.lookCmdtInfo			

ID	CCtrl		
----	-------	--	--

模块名		CommentController	
接口 ID	cmmt	功能描述	发表评论，向模型发出请求
		语法	Void comment(Comment);
		前置条件	评论信息合法
		后置条件	向模型发出评论请求
		不变量	无
CMod.comment, CMod.searchComment			

ID		CMod	
模块名		CommentModel	
接口 ID	cmmt	功能描述	发表评论的业务逻辑
		语法	Void Comment(Comment);
		前置条件	无
		后置条件	评论数据存储
		不变量	无
接口 ID	search Cmmt	功能描述	返回评论内容
		语法	Comment searchComment(Condition);
		前置条件	无
		后置条件	返回信息给视图以展示
		不变量	无
需求接口			
DBService KDtct.detect			

ID		SReq	
模块名		SearchRequest	
接口 ID	Getter	功能描述	得到搜索条件具体内容
		语法	Something getSomething(SearchRequest);
		前置条件	无
		后置条件	无
		不变量	无
接口 ID	Setter	功能描述	设置搜索条件具体内容
		语法	Void setSomething(SearchRequest);
		前置条件	无
		后置条件	无
		不变量	无
需求接口			
System.Collection			



ID		SResult	
模块名		SearchResult	
接口 ID	getter	功能描述	得到搜索结果具体内容
		语法	List getSearchResult(SearchResult);
		前置条件	无
		后置条件	无
		不变量	无
接口 ID	del	功能描述	删除搜索结果中的元素
		语法	Boolean deleteElem(SearchResult, Elem);
		前置条件	搜索结果中存在该 Element
		后置条件	删除对应项目
		不变量	无
需求接口			
System.Collection			

ID		SynoDtct	
模块名		SynoDetection	
接口ID	detect	功能描述	检测是否含有同义词
		语法	Synonym detect(SearchRequest);
		前置条件	同义词袋已建立
		后置条件	同时进行原词和同义词的搜索
		不变量	无
SWBg,detect			

ID		SWBg	
模块名		SynoWordBag	
接口 ID	detect	功能描述	检测同义词
		语法	Synonym detect(String keyword);
		前置条件	词袋已建立
		后置条件	无
		不变量	无
IOService			

ID		LinfoMod	
模块名		LookInfoModel	
接口	lkInfo	功能描述	产看商品信息

□ ID	语法	CmdtInfo lookCmdtInfo(Cmdt);
	前置条件	产品存在
	后置条件	返回信息
	不变量	无
DBService		

ID		Event	
模块名		EventInterface	
接口 ID	Ntf	功能描述	广播通知事件监听者
		语法	Void notify();
		前置条件	监听者已经注册
		后置条件	事件广播
		不变量	无
接口 ID	Regst	功能描述	监听者注册事件
		语法	Void register(Event, Contact);
		前置条件	Contact 没有注册过同一事件
		后置条件	无
		不变量	无
需求接口			
OS 进程通信 程序语言 ProcedureCall			

ID		ADtct	
模块名		AttackDetection	
接口 ID	Rpt	功能描述	报告出现攻击
		语法	Void reportAttack(AttackEvent);
		前置条件	满足攻击检测条件且系统不奔溃
		后置条件	无
		不变量	无
EventInterface.ntf			

ID		LdBlc	
模块名		LoadBalance	
接口 ID	Arg	功能描述	分发用户的 Http 请求
		语法	Void arrange(HttpRequest);
		前置条件	Http 请求合法安全
		后置条件	转给相应功能处理模块
		不变量	无

接口 ID	SendStatus	功能描述	汇总当前服务器的各个性能指标，形成映像，发送服务器状态消息
		语法	sendSeverStatusMessage();
		前置条件	无
		后置条件	StatusMessage 有生存周期
		不变量	无
需求接口			
Model.busesstask			

ID		CtctList	
模块名		ContactList	
接口ID	SMsg	功能描述	给相应的关注客户发送消息，消息形式多样，可以是系统通知（管理员）也可以是邮件（比如给商家）等。
		语法	Void sendMessage(Event);
		前置条件	关注者已经注册
		后置条件	广播方式发送信息给所有关注者
		不变量	
POP3			

ID		Mnt	
模块名		Monitor	
接口ID	Rpt	功能描述	报告服务器异常
		语法	reportServerExceptio(Source);
		前置条件	从服务器状态信息中检测到服务器异常
		后置条件	通知系统管理员
		不变量	无
LoadBalance.sendStatus			

ID		Fwall	
模块名		Firewalls	
接口ID	Fwd	功能描述	传入用户的请求
		语法	Void forwardRequest(HttpRequest);
		前置条件	Firewalls.atkJg=false
		后置条件	进行负载均衡
		不变量	无
接口ID	atkJg	功能描述	对用户请求进行攻击检测
		语法	Boolean attackJudge(HttpRequest);
		前置条件	存储一定时间之内的访问数据，以供判断

		后置条件	根据检测结果进行处理
		不变量	无
需求接口			
NetService.send			
NetService.recieve			

ID		DAdmin	
模块名		DataAdminInterface	
接口ID	opD	功能描述	操作数据
		语法	Void operateData();
		前置条件	管理员权限验证
		后置条件	检查数据一致性
		不变量	无
DBMS			
FileSystemCall			

ID		DMngC	
模块名		DataMngC	
接口ID	Upd	功能描述	数据更新对外接口
		语法	updateData(Data);
		前置条件	具有更新权限
		后置条件	无
		不变量	无
Middleware.rpt			
DataMngS.upd			

ID		UMngC	
模块名		UserMngC	
接口 ID	del	功能描述	删除用户
		语法	Void enableTalk(User user);
		前置条件	用户被判定为水军
		后置条件	用户不能再发表评论
		不变量	无
接口 ID	noTalk	功能描述	删除用户
		语法	Void deleteUser(User user);
		前置条件	满足删除用户条件，具有管理员权限执行
		后置条件	数据库中用户的历史数据需要保留
		不变量	无

需求接口
Middleware.rpc

ID		SAdmin	
模块名		SystemAdminInterface	
接口 ID	sysS	功能描述	查看系统状态
		语法	getSystemStatus();
		前置条件	得到 Monitor 的报告，得到服务器状态 Message 清单
		后置条件	人工监控
		不变量	无
Monitor.rpt ServerMessageStatus.list			

ID	FAdaptor		
模块名	CmdtFormatAdaptor		
接口 ID	Adpt	功能描述	配适不同网站数据格式
		语法	Void adapt();
		前置条件	无
		后置条件	无
		不变量	信息内容不变

ID		DRcvry	
模块名		DataRecovery	
接口ID	rcvry	功能描述	数据恢复
		语法	dataRecovery
		前置条件	服务器硬件故障或系统故障已经排除
		后置条件	写入数据
		不变量	无
IOSercive VFS.AData			

ID	DMngS		
模块名	DataMngS		
接口	Upd	功能描述	数据更新对外接口
		语法	updateData(Data);

ID		前置条件	具有更新权限
		后置条件	无
		不变量	无
VFS.AData			

ID		DBkp	
模块名		DBkp	
接口ID	bkp	功能描述	数据热备份
		语法	Void recovery();
		前置条件	检查数据服务器状态
		后置条件	写入数据
		不变量	无
IOService NetworkService			

ID		Middleware	
模块名		Middlewareplatform	
接口ID	rmC	功能描述	远程过程调用
		语法	Call
		前置条件	本地 Stub
		后置条件	调用远程方法
		不变量	无
NetworkService			

ID		Crawler	
模块名		DataCrawler	
接口ID	getD	功能描述	爬取数据
		语法	Data getData();
		前置条件	数据已经爬取
		后置条件	无
		不变量	无
HttpRequest			

ID	UpdateD
模块名	updateData

接口ID	updD	功能描述	大规模更新数据
		语法	Void updateData(Data)
		前置条件	数据格式匹配
		后置条件	不影响前台正常使用
		不变量	无
DataCrawler.getD			

ID		VFS	
模块名		VFS	
接口 ID	AData	功能描述	提供数据访问的一致性接口和对文件的组织方式的屏蔽
		语法	Result accessData(DataRequest);
		前置条件	数据存在
		后置条件	数据访问或者数据变更
		不变量	无
FileSystemCall IOService			

6.3 进程视图

6.3.1 进程描述

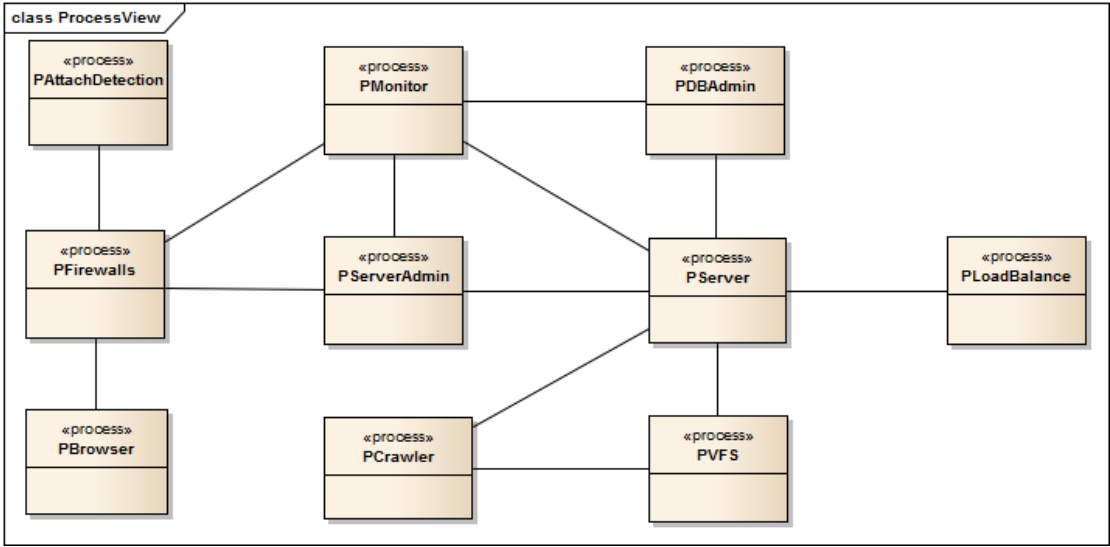


图 15 比价系统的进程描述图

逻辑视图和进程视图之间的映射关系如下表所示：

进程	部件或连接件
PMonitor	Monitor, Message, EventRouter
PServerAdmin	AdminInterface
PAttackDetection	EventRouter, AttackDetection
PLoadBalance	LoadBalane
PFirewalls	Firewalls, EventRouter
PDBAdmin	AdminInterface
PServer	View, CommentController, BuyController, FrontFramework, SearchModel, LookInfoModel, CommentModel, BuyModel, ContactList, ResultSorter, ResultFilter, HttpRequet,Monitor
PClient	DataMng, UserMng
PVFS	VirtualFileSystem
PCrawler	DataCrawler, Adapter

### 6.3.2 进程通信描述

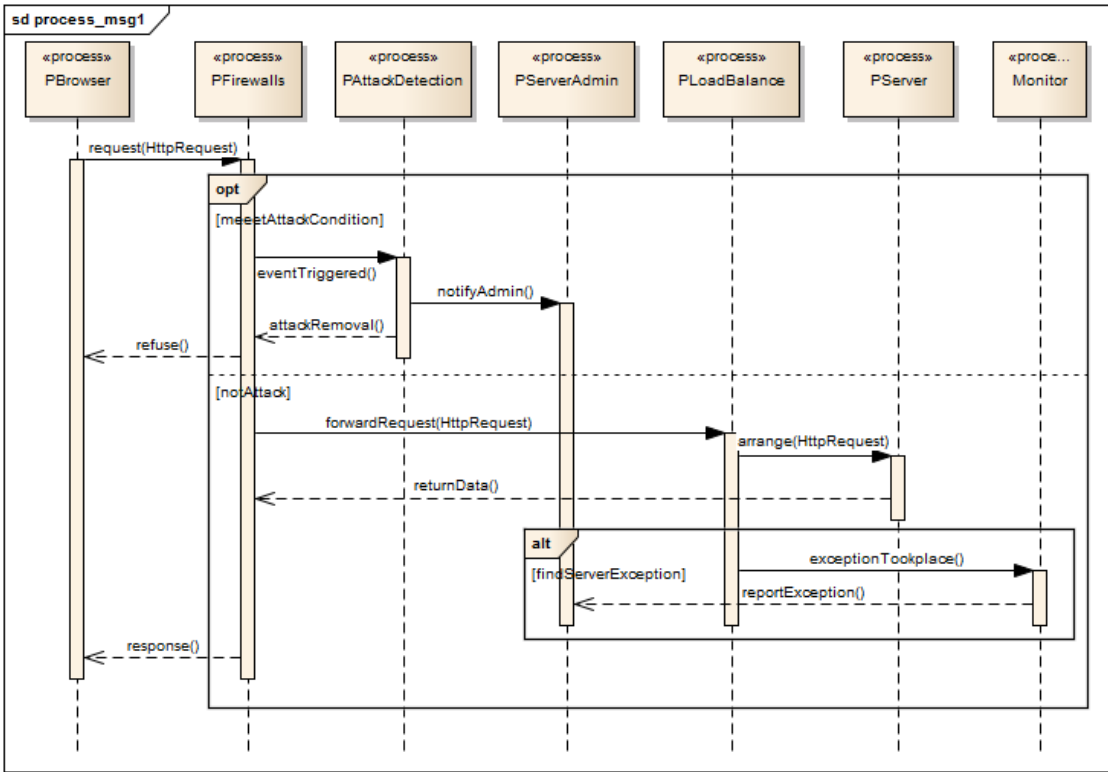


图 16 比价系统进程通信描述一

对应图 16 的接口描述：

ID		PBrowser	
发送接口			
接口 ID	reqst	功能描述	浏览器方发送 Http 请求
		通信方式	Http 通信协议
		接口定义	Request(HttpRequest);



接收接口			
PFirewalls.Fwd			
PFirewalls.EDtct			

ID		PFirewalls	
发送接口			
接口ID	Fwd	功能描述	传入外界合法请求
		通信方式	ProcedureCall
		接口定义	Forward(HttpRequest);
接口ID	EDtct	功能描述	满足攻击检测进程注册的条件，通知出现攻击
		通信方式	ProcedureCall+操作系统进程间通信
		接口定义	eventTriggered(HttpRequet, Process);
PAttackDetection.ntf			
PLoadBalance.arg			

ID		PAttackDetection	
发送接口			
接口ID	ntf	功能描述	通知系统管理员出现攻击
		通信方式	ProcedureCall
		接口定义	notifyAdmin(AttackEvent);
接收接口			
PServerAdmin			

ID		PLoadBalance	
发送接口			
接口 ID	arg	功能描述	分配对服务器的请求
		通信方式	网络通信的封装模块
		接口定义	Arrange(HttpRequest);
接收接口			
PServer			

ID		Monitor	
发送接口			
接口 ID	rpt	功能描述	向系统管理员报告服务器异常
		通信方式	ProcedureCall+操作系统进程间通信
		接口定义	reportException(Exception);
接收接口			
PServerAdmin			

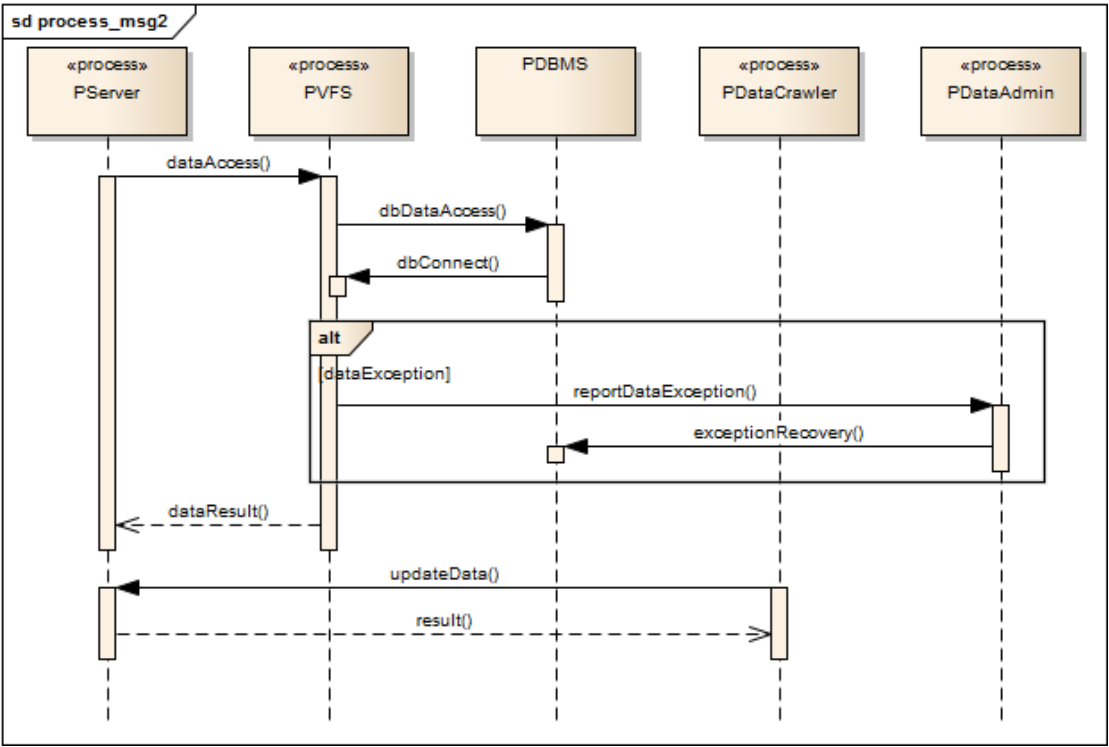


图 17 比价系统进程通信描述二

对应图 17 的接口描述如下：

ID		PServer	
发送接口			
接口 ID	data	功能描述	向虚拟文件系统访问数据
		通信方式	ProcedureCall
		接口定义	dataAccess(DataRequest);
接收接口			
PVFS.dbcnct			
PVFS.rpt			

ID		PVFS	
发送接口			
接口 ID	dbcnct	功能描述	访问物理数据
		通信方式	ProcedureCall
		接口定义	dbDataAccess(DataRequest);
接口 ID	rpt	功能描述	报告数据异常
		通信方式	ProcedureCall
		接口定义	reportDataException(Exception);
PDBMS			
PDataAdmin			

ID	PDataCrawler
----	--------------

发送接口			
接口ID	updt	功能描述	数据爬虫更新数据
		通信方式	ProcedureCall
		接口定义	updateData(Data);
接收接口			
PServer.Datamng			

## 6.4 部署视图

### 6.4.1 基础设施类型

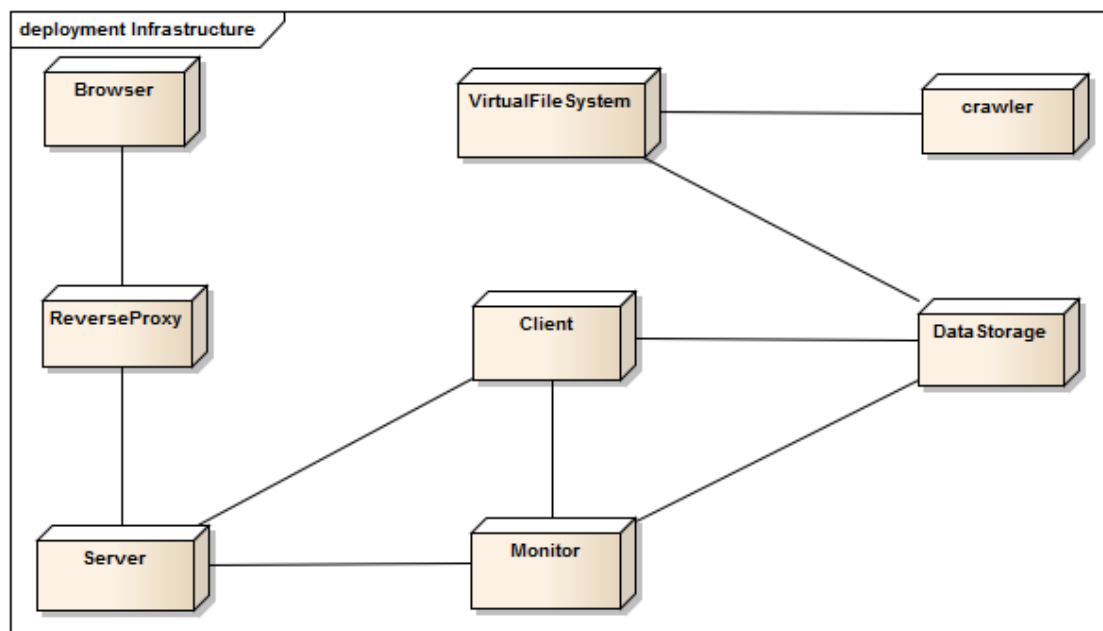
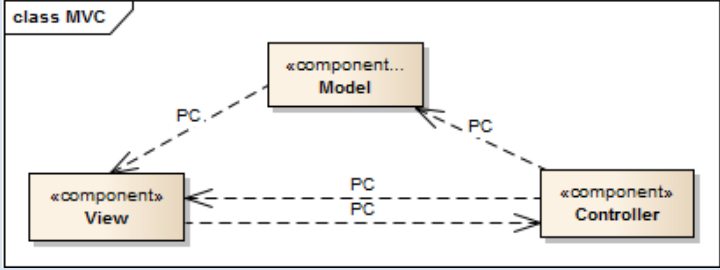


图 18 比价系统基础设施类型描述

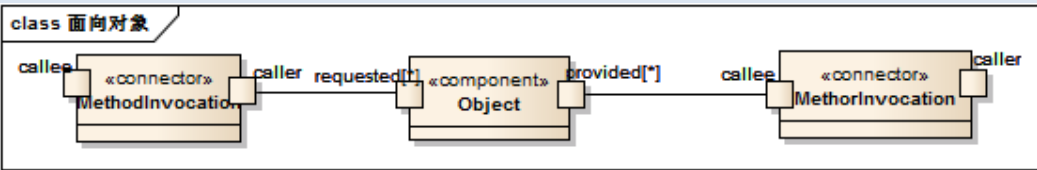
如上图描述了比价系统的基础设施，它需要八种类型的物理节点：

- (1) Browser: 广域网中的普通节点，用户使用多个浏览器访问系统，发出请求。
- (2) ReverseProxy: 安装有反向代理系统的物理节点，专门用于负载均衡，即将并发的 HTTP 请求分配给不同的服务器，有多个实例，按照层次结构（如树状）组织起来。
- (3) Server: 安装有比价系统脚本程序，可以进行业务逻辑处理的物理节点，负责业务逻辑处理，有多个实例。
- (4) Monitor: 专门为保证服务器和数据存储可靠性设置的监控节点，有 2 个实例，监控全网的运行状态。
- (5) Client: 安装有数据管理模块和服务器管理模块的物理节点，实例个数由管理员数目决定，是管理员工作的节点，用户管理员在后台对数据进行操作，对系统进行维护。
- (6) VirtualFileSystem: 安装有 VFS 的物理节点，专门用于分配数据访问，并且实现数据备份和恢复，提高数据可靠性和访问的效率。简单情况下单个实例，复杂情况下多个实例，但对外表现为一个整体。
- (7) DataStorage: 专门用于数据存储的物理节点。



	2. 视图和控制可修改性，适合于此系统用户量大，显示终端不一致的情况 3. 适宜于网络开发特征，现有的 LAMP 技术生态中提供对应的 MVC 框架可以使用 4. 参见设计决策 14
影响视图	逻辑视图、开发视图
	

## 7.2 面向对象风格

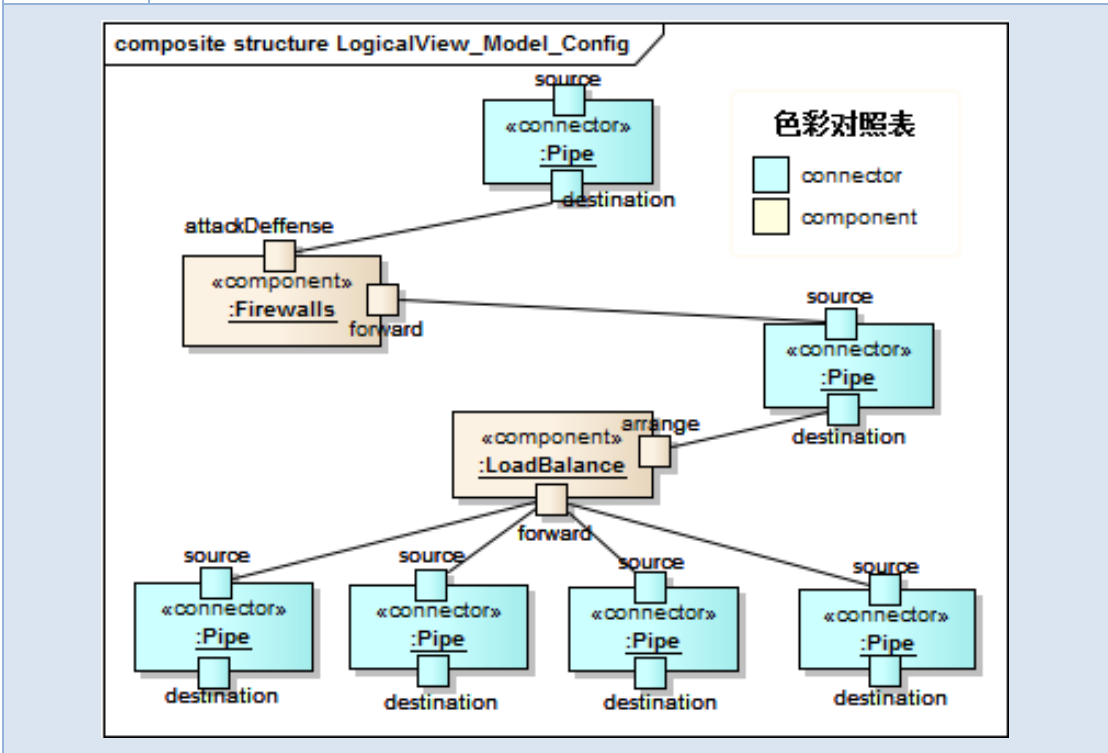
风格 ID	AS2
相关功能和约束	提供商品信息、提供商品评论平台、价格对比、搜索结果可调整、商家关注信息推送、自主信息获取、用户管理、入侵检测、隔离攻击
质量要求	商品信息正确、数据存储高可靠性、服务器高可靠性、搜索易用性高、大吞吐量、高并发、大数据量存储、可扩展性、快速开发
决策依据	1. 基于面向对象风格的封装可以实现内部的可修改性，对外提供统一的接口，符合信息隐藏和模块化的思想 2. 可以降低复杂度，促进并行开发 3. 易于理解和组织 4. 参见设计决策 1、2、3、4、5、9、10、13
影响视图	逻辑视图、开发视图
	

## 7.3 管道过滤器风格

风格 ID	AS3
相关功能和约束	入侵检测、攻击隔离、查看信息、评论商品、搜索信息、购买商品
质量要求	高并发、安全性、保证存取效率
决策依据	1. 管道过滤器具有可复用性；支持并发，可以提高性能 2. 管道过滤器符合信息隐藏的观点，内部修改性好，提供了很好的封装，

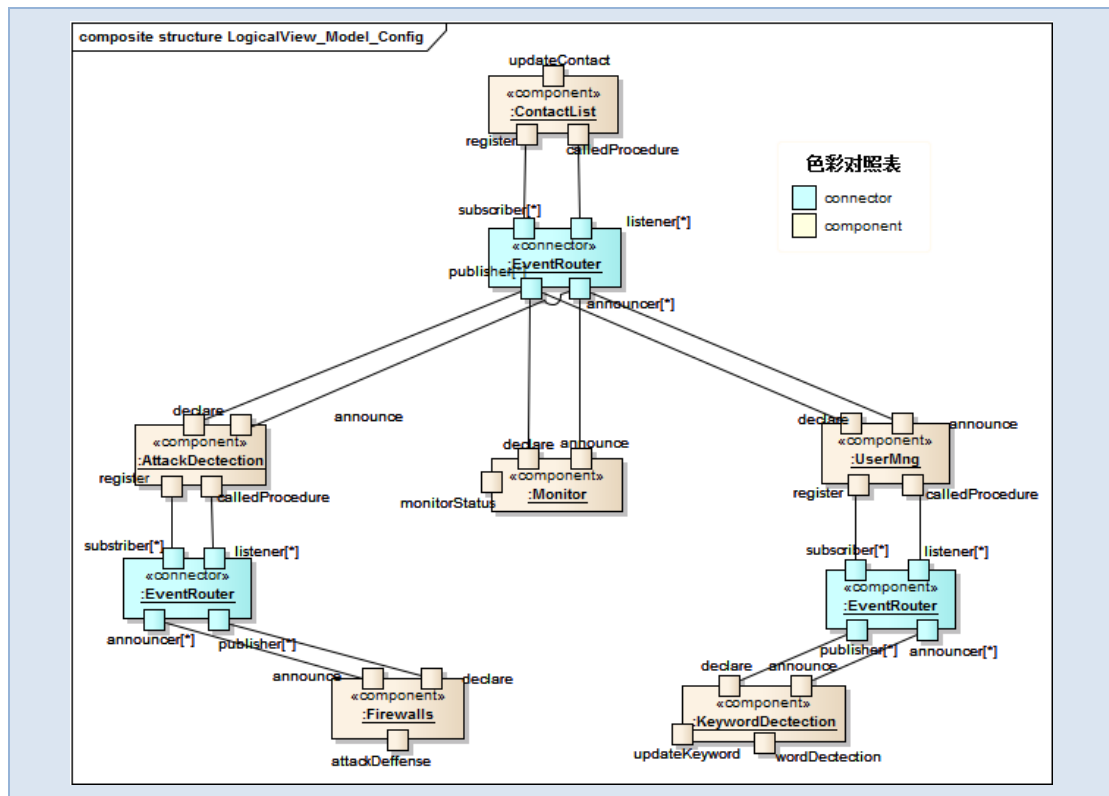
	并一次可以方便的进行扩展和替换。 3. 符合数据流的抽象，具有抽象一致性 4. 参见设计决策 8
--	--

影响视图	逻辑视图、开发视图、进程视图
------	----------------



## 7.4 隐式调用风格

风格 ID	AS4
相关功能和约束	入侵检测、攻击隔离、通知商家感兴趣信息、敏感词检测、用户管理（水军检测）
质量要求	入侵检测策略可扩展、商家感兴趣信息可扩展、敏感词可重定义、数据存储高可靠性、服务器高可靠性
决策依据	1. 隐式调用风格提供优秀的可复用性、可修改性特点 2. 可以满足在某一时间内通知用户的实时性要求，可以用多进程并发实现，性能好 3. 参见设计决策 3、4、10、13
影响视图	逻辑视图、开发视图



## 8.小组分工

根据功能和工作量，将开发视图中的包进行如下分组：

名称	包含的开发包	负责人
商品 1	SearchModel,SearchResult,SearchResultSorter	余泳桦
商品 2	SynoDetection,SynoWordBag,SearchRequest	刘磊
评论+外围	CommentModel,CommentController,KeywordDecection	王珺宇
管理员+视图	UserManagerC,SystemAdminInterface,CommentManager,FrontUI,LookInfoModel	罗浩然
数据库管理 1	DataAdminInterface,DataManagerC,DataManagerS,CommodityInfoAdapter	王宁
数据库管理 2	DataBackup,DataRecovery,DataCrawler	戚航
外围 1	EventInterface,ContactList,FireWall.LoadBalance	刁子恒
外围 2	Monitor,ServerStatusMessage,FireWall.AttackDetection	张笛