

爱比价网

软件体系结构设计文档

石若非

141250108

目录

- 1 关注点.....3
 - 1.1 需求.....3
 - 1.2 涉众及关注点描述.....5
 - 1.3 项目环境.....6
 - 1.4 业务目标.....7
- 2 体系结构需求定义.....7
 - 2.1 体系结构需求及约束.....7
 - 2.2 用例视图.....10
 - 2.3 场景描述.....12
- 3 建立初始体系结构.....23
 - 3.1 逻辑视图.....23
 - 3.1.1 部件与连接件.....23
 - 3.1.2 配置描述.....25
 - 3.2 开发视图.....26
 - 3.2.1 模块组织.....26
 - 子系统组织.....27
 - 3.3 进程视图.....28
 - 3.4 部署视图.....29
- 4 设计决策.....29
 - 4.1 关于风格的决策.....29
 - 4.1.1 数据抽象和面向对象风格.....29

4.1.2	管道-过滤器风格.....	30
4.1.3	基于事件的隐式调用风格.....	31
4.1.4	分层+MVC 风格.....	32
4.1.5	仓库风格.....	33
4.2	相关设计决策.....	34
4.2.1	设计决策 1——攻击隔离与 IP 屏蔽.....	34
4.2.2	设计决策 2——搜索结果调整.....	36
4.2.3	设计决策 3——自动爬取各网站购物信息.....	37
4.2.4	设计决策 4——高可靠性服务器系统.....	38
4.2.5	设计决策 5——高性能服务器系统.....	39
4.2.6	设计决策 6——评论管理.....	41
4.2.7	决策 7——高可靠性数据库系统.....	42
4.2.8	决策 8——高性能数据库系统.....	43
4.2.9	决策 9——浏览器兼容与动态网页.....	44
4.2.10	决策 10——信息加密.....	45
5	最终高层结构.....	46
5.1	逻辑视图.....	46
5.2	开发视图.....	47
5.2.1	模块描述.....	47
5.2.2	接口定义.....	47
5.3	进程视图.....	60
5.3.1	进程描述.....	60

5.3.2	进程通信描述.....	61
5.4	部署视图.....	65

1 关注点

1.1 需求

ID	关注点	类型	内容描述	灵活性
FR1	提供高易用性商品搜索功能	功能需求	可以允许用户通过输入关键词进行商品检索，兼容同义词，屏蔽特定商品	灵活性表现在可扩展的同义词表和动态的屏蔽选择上
FR2	搜索结果排序	功能需求	搜索结果可以按价格、销量、评分排序，并允许竞价调整顺序	排序结果可根据动态的竞价规则调整顺序
FR3	提供商品信息	功能需求	提供商品有效的详细信息，包括评价、价格、近期销量趋势图	商品信息的种类和展示方式具有灵活性
FR4	提供评论平台	功能需求	形成论坛式的可多次评论回复机制	灵活性较低
FR5	评论管理	功能需求	通过设置评论的敏感词汇，进行相应的水军判定、通知商家、审查等操作	灵活性表现在敏感词汇表的扩展和相应通知操作上
FR6	提供商家间比	功能需求	展示导航网站排行前 10 的购	对比方式与展示方

	价功能		物网站的价格比较	式具有灵活性
FR7	信息爬取	功能需求	定期爬取各大购物网站的商 品信息并更新	灵活性体现在兼容 不同的数据格式上
FR8	提供购物网站 的购物链接和 支付功能	功能需求	系统可以为商品提供对应购 物网站的购买页面链接，同 时允许直接支付	灵活性较低
QUA1	数据库的高可 靠性	质量属性	数据库可以及时更新且更新 时不影响用户正常访问系 统，故障时可快速回复	故障发生后时不能 影响正常工作并 12 小时内恢复
QUA2	数据库性能	质量属性	数据库容纳至少 1000 万条数 据，约 2000 条被频繁访问	灵活性较低
QUA3	服务器的高可 靠性	质量属性	服务器出现故障时及时发现 且不影响访问	发现时间小于 1 小 时
QUA4	服务器性能	质量属性	高峰期允许 500 万用户在线， 不能发生并发性故障	响应时间允许波动 但故障率要低于 1%且可恢复
QUA5	系统实时性	质量属性	用户访问时的响应时间应小 于 1s 且数据库信息实时更新	响应时间允许随负 载发生波动
QUA6	多浏览器兼容	质量属性	兼容各平台的多种浏览器	兼容性可拓展可升 级
QUA7	可兼容多网站 数据的爬虫	质量属性	兼容不同购物网站的数据格 式，不能出现转换错误	兼容性可拓展、可 应对外部网站变更

QUA8	入侵检测和攻击隔离	质量属性	可屏蔽 XSS 攻击和特定 IP，并可能增加新的入侵检测规则	XSS 攻击判定、屏蔽 IP 列表和其他屏蔽方式有灵活性
QUA9	可测试性	质量属性	要求可以通过变更配置迅速进入测试模式	灵活性体现在测试模式的展示方式上
QUA10	可维护性	质量属性	系统易于维护，且能自动组织有规律的错误码	灵活性体现在错误码的种类和数量上
QUA11	高易用性系统	质量属性	人机交互良好，系统使用人性化，评论管理自动化	灵活性较高，可能以各种形式变更
QUA12	符合相关法规	约束	符合国家相关网络购物法规，自动屏蔽相应敏感词	敏感词列表和屏蔽选项具有灵活性

1.2 涉众及关注点描述

涉众	主要目标	态度	关注点
女性网民	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	由于对现有比价网有不同程度的不满，所以期待新系统的出现	使用方便 信息全面可靠 评价交流方便
加盟商城	更够增加商品的销量，扩大顾客量，打造知名度	支持，但希望植入相关广告	商品销售量 商品销售利润
客户公司	通过系统获得更多商	支持	加盟商家数量

	家加盟获取利润		普通用户数量 投资与利润
系统管理员	无	支持，但对系统的负载能力持担忧态度	系统的可维护性 系统的稳定性
数据库管理员	无	支持，但希望数据库足够强大且便于管理	数据库的健壮性 数据库维护是否方便快捷
系统开发商	以最少的成本开发出客户所需的系统	支持	是否满足客户需求

1.3 项目环境

ID	关注点	类型	描述	灵活性
ENV1	进度	项目属性	计划 2 个月内完成第一版，6 个月内第二版，第二版实现 95%功能特性	在不包括责任人评审情况下，最多延期 3 周
ENV2	质量	项目属性	必须通过 95%以上用户验收测试，必须通过所有安全性测试	灵活性较低
ENV3	人员	项目属性	开发人员有限，缺少相应技术，需要额外增加或学习	灵活性较低
ENV4	信息保护	操作环境	数据库中的商品信息和商家信息需要保护和安全权限	加密方式和程度具有灵活性

1.4 业务目标

ID	关注点	类型	描述	灵活性
BUS1	网站用户量	业务目标	第一版应用 6 个月内增加 500 万用户量，12 个月内增加 1000 万用户量	灵活性较低
BUS2	用户满意度	业务目标	第一版应用 3 个月内满意度达 3 星，12 月内提高 1.5	灵活性较低
BUS3	加盟商家与利润	业务目标	第一版应用 12 月内加盟 10-15 家，利润提高 30%到 50%	灵活性较低

2 体系结构需求定义

2.1 体系结构需求及约束

优先级由 1 到 5 逐渐增加

ID	体系结构需求	优先级	约束 ID	约束设计	相关约束
R1	提供高易用性搜索功能	5	C1	兼容同义词，维护词汇表	
			C2	及时屏蔽负面商品	
R2	搜索结果可调整	5	C3	结果可按销量价格等排序	
			C4	允许竞价调整顺序	
			C5	排序方案可拓展	
R3	提供详细商品信息	5	C6	保证信息及时可靠	阻碍 C39
			C7	可拓展的信息展示方式	

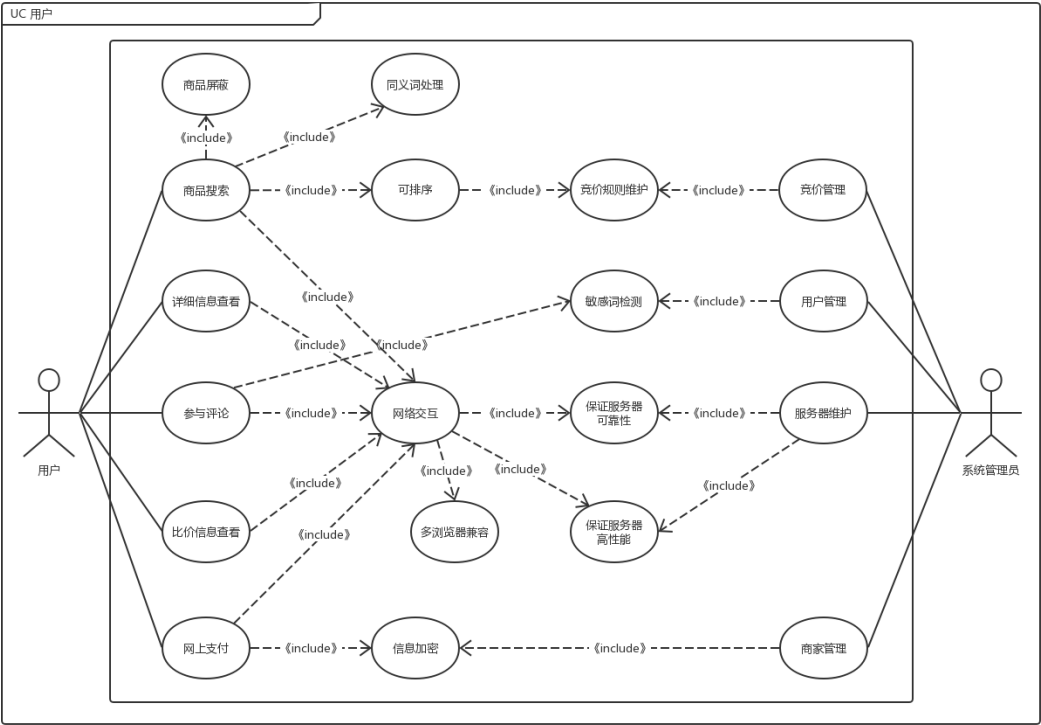
R4	提供灵活可靠的评论平台	4	C8	灵活快捷，同时方便管理的贴吧式平台	
R5	完备的评论管理系统	4	C9	自动检测敏感词汇	促进 C41
			C10	自动化的通知模块	促进 C41
			C11	可拓展的评论处理方式	
R6	提供可靠的比价信息	5	C12	进行比价的购物网站不小于 10 个	
R7	可兼容可拓展的信息爬取模块	5	C13	兼容不同网站的数据格式	
			C14	自动爬取，及时更新	阻碍 C21
			C15	可拓展的爬取和处理模块	
R8	便捷的购物链接和支付功能	3	C16	支付过程加密	促进 C47
			C17	与银行系统和其他网站制定严格接口	促进 C48
R9	实现高可靠性数据库系统	5	C18	故障后 12 小时内恢复	
			C19	保证数据库可维护性	
R10	实现高性能数据库系统	5	C20	容纳至少 1000 万条数据	
			C21	保证数据快速存取	
R11	实现高可靠性服务器系统	5	C22	故障 1 小时内发现	
			C23	故障时不影响正常访问	
R12	实现高性能服务器系统	5	C24	高峰时容纳 500 万用户	
			C25	不发生并发性错误	阻碍 C21
R13	保证系统使用	5	C26	响应时间小于 1s	阻碍 C24

	数据更新实时性		C27	数据即使更新	阻碍 C21
R14	兼容多浏览器	3	C28	兼容各平台不同浏览器	
			C29	兼容方式可更新	
R15	实现数据自动爬 取与处理模块	4	C30	自动爬取处理数据	促进 C14
			C31	数据定时更新至数据库	促进 C14
R16	实现入侵检测与 攻击隔离功能	4	C32	自动检测各类入侵	促进 C48
			C33	自动隔离相应 IP	促进 C48
			C34	可拓展的检测和隔离模块	
R17	保证可测试性	3	C35	通过更改配置进入测试	
			C36	测试模式输出调用路径， 且方式可拓展	促进 C22
R18	保证可维护性	3	C37	组织有规律的错误码	促进 C22
R19	保证系统的高易 用性	3	C38	保证良好的人机交互	
			C39	后台数据更新不影响用户 正常访问	
			C40	自动化用户管理	促进 C10
R20	保证符合相关法 律法规	3	C41	自动检测敏感词并通知	
			C42	可拓展的词汇表	
R21	设计方案可以快 速开发	2	C43	第一版 2 个月内发布	
			C44	模块间可以并行开发	
R22	设计方案符合开 发人员现有技能	2	C45	使用主流的 WEB 技术	促进 C43
			C46	封装数据爬取模块	

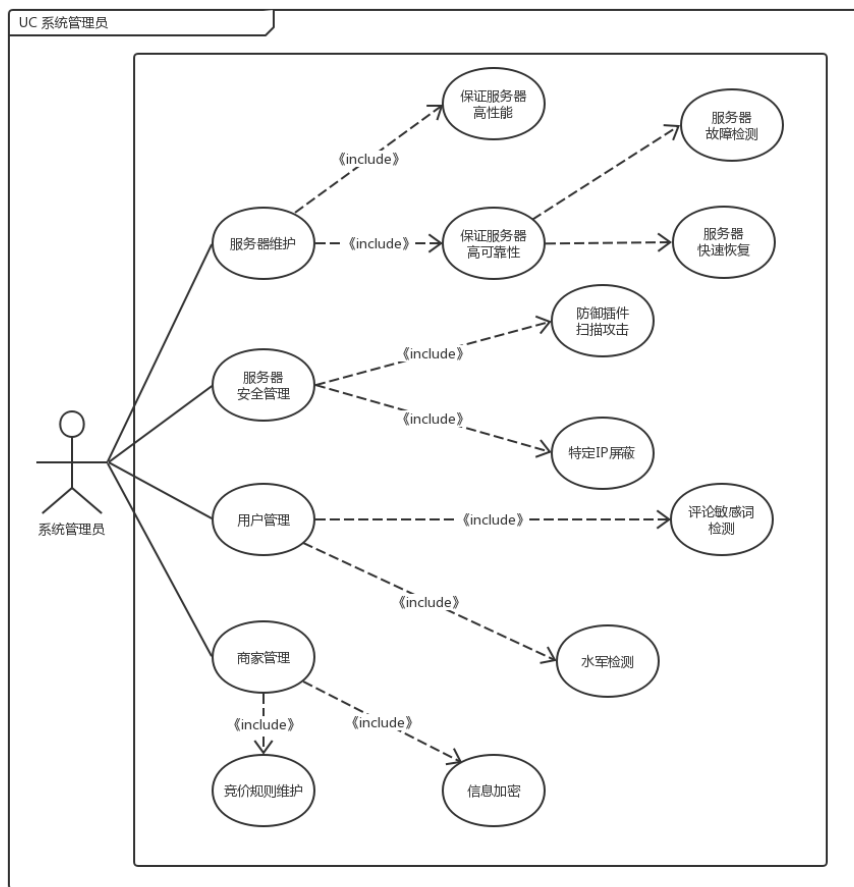
R23	保证系统数据和信息安全	5	C47	数据库信息加密	
			C48	加密商家隐私信息	

2.2 用例视图

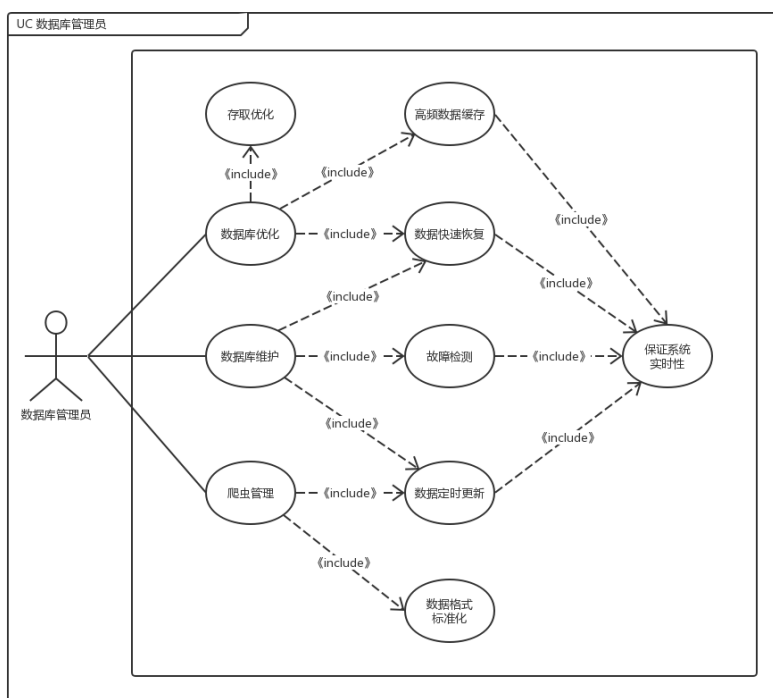
用户用例视图：



系统管理员用例图：



数据库管理员用例图：



2.3 场景描述

场景 ID		S1
商业目标		屏蔽有负面新闻的商品
相关需求与设计约束		R1：提供高易用性搜索 C2：及时屏蔽负面商品
场景内容	刺激	发生商品相关的负面事件
	刺激源	外部环境，包括舆论、用户投诉、商家投诉等
	环境	用户正常使用系统时
	制品	搜索结果过滤模块
	响应	过滤模块增添的屏蔽列表中增添新的商品标签
	响应度量	任何符合屏蔽标准的商品都不能出现在搜索结果中

场景 ID		S2
商业目标		搜索时处理同义词
相关需求与设计约束		R1：提供高易用性搜索功能 C1：兼容同义词表，维护词汇表
场景内容	刺激	需要引入新的同义词匹配关系
	刺激源	外部环境，多表现为新商品的上市
	环境	用户正常使用系统时
	制品	同义词匹配模块
	响应	同义词词汇表增加新的映射关系
	响应度量	热门商品的同义词匹配必须实现映射

场景 ID		S3
商业目标		搜索结果可排序，且允许竞价规则调整顺序
相关需求与设计约束		R2：搜索结果可调整 C3：搜索结果可按价格、销量、评分等排序 C4：允许竞价规则调整顺序 C5：排序方案可拓展
场景内容	刺激	在竞价规则上商品的排名出现变更
	刺激源	商家的竞价
	环境	用户正常使用系统时
	制品	搜索结果排序模块
	响应	搜索结果在现有排序基础上根据新的竞价调整
	响应度量	普通排序的顺序要完全正确，竞价规则造成的顺序影响误差不能超过±1 位

场景 ID		S4
商业目标		评论敏感词检测
相关需求与设计约束		R5：完备的评论管理系统 C9：自动检测敏感词汇 C10：自动化的通知模块
场景内容	刺激	检测到某用户的某评论具有敏感词汇
	刺激源	用户
	环境	用户正常使用系统时

	制品	评论文本检测模块
	响应	根据敏感词通知管理员或商家
	响应度量	所有出现在敏感词汇表中的敏感词均能做出相应的操作与处理

场景 ID		S5
商业目标		网络支付信息加密
相关需求与设计约束		R8：便捷可靠的购物链接个支付功能 C16：支付过程加密 C17：与银行系统及其他购物网站制定严格可靠的接口
场景内容	刺激	用户希望通过爱比价网系统进行支付购买
	刺激源	用户
	环境	用户正常使用系统时
	制品	网络支付加密模块
	响应	对支付过程、链接、表单信息等进行加密
	响应度量	加密过程符合互联网加密标准

场景 ID		S6
商业目标		多浏览器兼容
相关需求与设计约束		R14：兼容多浏览器 C28：兼容各平台不同浏览器 C29：兼容方式可更新
场景内容	刺激	不同用户使用不同浏览器访问系统时
	刺激源	用户

	环境	用户正常使用系统时
	制品	浏览器兼容模块
	响应	返回请求网页
	响应度量	网页展示出现兼容性问题概率小于 1%

场景 ID		S7
商业目标		服务器故障快速检测
相关需求与设计约束		R11：实现高可靠性服务器系统 C22：故障一小时内发现 C23：故障时不影响用户正常访问
场景内容	刺激	某个服务器发生故障时
	刺激源	工作中的服务器
	环境	系统正常运作时
	制品	服务器集群管理模块
	响应	返回服务器的具体故障信息
	响应度量	服务器故障需要在 1 小时内发现

场景 ID		S8
商业目标		保证服务器在并发上的高性能
相关需求与设计约束		R12：实现高性能服务器系统 C24：高峰时期可容纳 500 万用户
场景内容	刺激	高峰时期约 500 万用户同时访问该系统
	刺激源	用户

	环境	用户正常使用系统时
	制品	分布式服务器与集群架构
	响应	负载均衡策略对访问进行分流
	响应度量	用户量小于 500 万时需保证正常使用，大于 500 万时的并发性 错误率不能高于 10%

场景 ID		S9
商业目标		保证服务器故障后可
相关需求与设计约束		R11：实现高可靠性服务器系统
场景内容	刺激	某个服务器发生故障时
	刺激源	工作中的服务器
	环境	系统正常运作中
	制品	服务器集群管理模块
	响应	故障服务器重新启动并配置入集群
	响应度量	单个服务器故障发现后需要可以在 12 小时内恢复

场景 ID		S10
商业目标		检测插件扫描攻击并防御
相关需求与设计约束		R16：实现入侵检测与攻击隔离功能 C32：自动检测各类入侵
场景内容	刺激	出现恶意的攻击或入侵
	刺激源	恶意的系统使用者

	环境	系统正常工作时
	制品	入侵检测模块
	响应	检测到入侵或攻击，收集信息并通知管理员
	响应度量	入侵检测响应时间小于 10 秒，屏蔽 90%以上插件扫描攻击

场景 ID		S11
商业目标		屏蔽特定 IP 访问
相关需求与设计约束		R16：实现入侵检测与攻击隔离功能 C33：自动隔离相应 IP
场景内容	刺激	特定 IP 试图访问本系统时
	刺激源	被列入屏蔽列表的特定 IP
	环境	系统正常工作时
	制品	IP 屏蔽模块
	响应	拦截特定 IP 的访问
	响应度量	被屏蔽的 IP 应无法对系统进行访问

场景 ID		S12
商业目标		敏感评论管理
相关需求与设计约束		R5：完备的评论管理系统 C9：自动检测敏感词汇
场景内容	刺激	用户发表的评论带有敏感词汇时
	刺激源	用户
	环境	用户和系统管理员正常使用该系统时

	制品	评论管理模块
	响应	管理员删除敏感评论
	响应度量	所有敏感评论需要即时通知管理员

场景 ID		S13
商业目标		水军检测与屏蔽
相关需求与设计约束		R5：完备的评论管理系统 C10：自动化的通知模块
场景内容	刺激	水军多次发布相同内容的评论时
	刺激源	不良的网站使用者
	环境	系统正常工作时
	制品	水军检测模块
	响应	自动检测水军并通知管理员是否封禁
	响应度量	1 小时内连续发布 5 条以上相同内容的用户被判断为水军

场景 ID		S14
商业目标		数据库故障快速检测
相关需求与设计约束		R9：实现高可靠性数据库系统 C19：保证数据库可维护性
场景内容	刺激	数据库发生故障
	刺激源	数据库系统
	环境	数据库系统发生故障时
	制品	数据库管理模块

	响应	检测到数据库故障并通知数据库管理员
	响应度量	数据库发生故障后 1 小时内需要发出通知

--	--

场景 ID		S15
商业目标		数据库故障快速恢复
相关需求与设计约束		R9：实现高可靠性数据库系统 C18：故障后 12 小时内恢复 C19：保证数据库可维护性
场景内容	刺激	数据库发生故障
	刺激源	数据库系统
	环境	数据库系统发生故障时
	制品	数据库管理模块
	响应	对数据库进行恢复
	响应度量	数据库发生故障后 12 小时内可以恢复

场景 ID		S16
商业目标		数据库更新时不影响系统正常使用
相关需求与设计约束		R9：实现高可靠性数据库系统
场景内容	刺激	数据库正在进行更新
	刺激源	数据库系统
	环境	系统正常使用且数据库需要更新时
	制品	数据库更新模块

	响应	数据库更新同时用户正常访问系统
	响应度量	数据库更新造成的数据错误不能超过 1%且可恢复

场景 ID		S17
商业目标		爬虫兼容各网站数据格式
相关需求与设计约束		R7：可兼容可拓展的信息爬取模块 C13：兼容不同网站的数据格式
场景内容	刺激	爬虫爬取到其他购物网站的数据并准备标准化
	刺激源	信息爬取模块
	环境	信息爬取模块正常运作时
	制品	爬虫数据标准化模块
	响应	数据被标准化为本系统数据库可用的格式
	响应度量	数据格式出错率不能超过 1%且可恢复

场景 ID		S18
商业目标		大容量数据库存取优化
相关需求与设计约束		R10：实现高性能数据库系统 C20：至少容纳 1000 万条数据 C21：保证数据快速存取
场景内容	刺激	大容量数据库存取数据
	刺激源	数据库访问接口
	环境	数据库系统正常运作时
	制品	数据库管理模块

	响应	数据库实现存取速度优化
	响应度量	用户一次访问响应时间小于 1s

场景 ID		S19
商业目标		实现高频数据缓存
相关需求与设计约束		R10：实现高性能数据库系统 R13：保证系统使用和数据更新实时性 C26：响应时间小于 1s
场景内容	刺激	对数据库进行访问
	刺激源	数据库访问接口
	环境	数据库系统正常运作时
	制品	数据库管理模块
	响应	数据库实现高频数据缓存
	响应度量	用户一次访问响应时间小于 1s

场景 ID		S20
商业目标		保证系统高实时性
相关需求与设计约束		R13：保证系统使用和数据更新实时性 C26：响应时间小于 1s C27：数据即时更新
场景内容	刺激	用户访问爱比价网站
	刺激源	用户
	环境	用户正常访问该系统时

	制品	服务器集群模块、数据库管理模块
	响应	实现系统高实时性
	响应度量	用户一次访问响应时间小于 1s

3 建立初始体系结构

3.1 逻辑视图

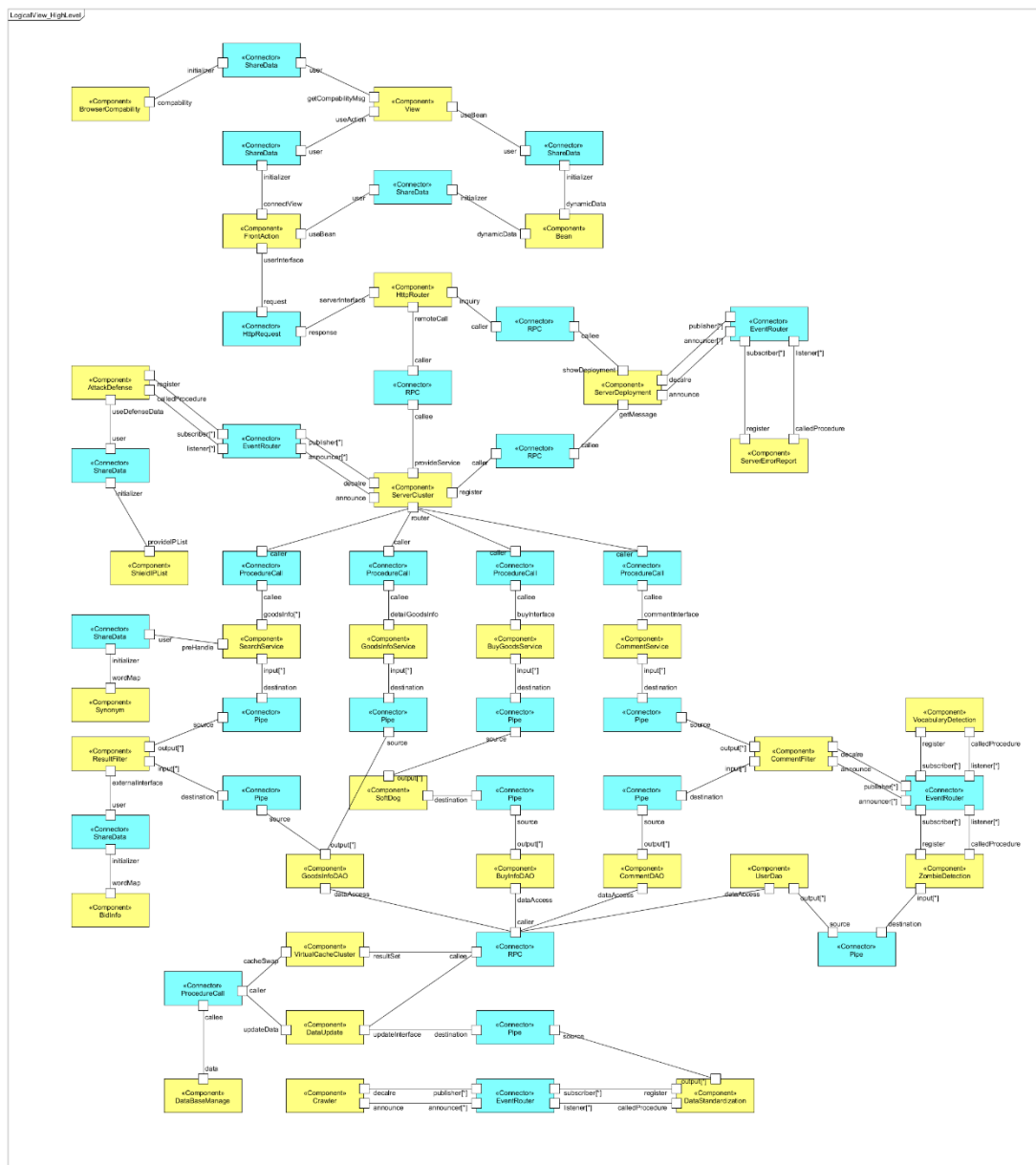
3.1.1 部件与连接件



- 1) 部件 BrowserCompability ,用于以不同方式渲染前端组件和数据从而兼容多浏览器 ,端口 compability 负责与其他部件进行 XML 格式的数据传递
- 2) 部件 View ,用于渲染、展示 WEB 应用的前端界面,端口 getCompabilityMsg 用于接收浏览器兼容处理后的 XML 数据,端口 useAction 负责创建可交互组件,端口 useBean 负责使用动态数据
- 3) 部件 Bean ,存储了当前页面的动态数据信息,端口 dynamicData 负责传递 bean 对象持有的动态数据
- 4) 部件 FrontAction ,负责管理前端用户与后端交互的接口,端口 useBean 负责使用动态数据,端口 connectView 负责与界面交互,端口 userInterface 为面向服务器端的用户接口
- 5) 部件 HttpRouter ,负责路由 HTTP 请求,端口 serverInterface 为服务器端的接口,端口 remoteCall 负责远程调用服务器端的服务,端口 inquiry 负责查询服务配置中心的配置
- 6) 部件 ServerCluster ,为分布式服务器集群,可以实现均衡负载,端口 provideService 用于提供相应服务,端口 router 负责路由发送,端口 declare 与 announce 为进行 EventRouter 监听与广播的接口(此后相同端口不再作额外解释)
- 7) 部件 ServerDeployment ,为服务配置中心,管理集群服务器的分配状况,端口 showDeplyment 负责展示当前配置状态,端口 getMessage 负责接收新的服务器注册请求
- 8) 部件 ServerErrorReprot ,用于监听服务配置中心并进行相应的错误报告
- 9) 部件 AttackDefense ,用于监听服务器请求,防御对服务器的插件扫描攻击,屏蔽短时间内频繁访问的 IP,端口 useDefenseData 用于接收当前的屏蔽列表数据
- 10) 部件 SheildIPList ,负责管理 IP 屏蔽列表,端口 provideIPList 负责传输屏蔽列表信息
- 11) 部件 SearchService ,负责管理搜索相关服务,实现后端数据与前端请求、视图的交互,端口 preHandle 用于进行搜索前的预处理工作
- 12) 部件 GoodsInfoService ,负责管理物品信息展示与比价服务,实现后端数据与前端请求、视图的交互
- 13) 部件 BuyService ,负责管理购买相关服务,实现后端数据与前端请求、视图的交互
- 14) 部件 CommentService ,负责管理评论相关服务,实现后端数据与前端请求、视图的交互
- 15) 部件 Synonym ,负责存储同义词对照表,端口 wordMap 用于接收词汇并返回相应的同义词列表
- 16) 部件 ResultFilter ,负责对搜索结果进行排序或一些拓展处理,端口 externalInterface 可用于拓展结果处理的方法与种类
- 17) 部件 BidInfo ,负责存储竞价与比价信息,端口 info 用于传递存储的信息数据
- 18) 部件 SoftDog ,意为加密狗,用于在购买服务中对用户、商家的信息加密
- 19) 部件 CommentFilter ,为评论信息过滤器,可以监听评论内容并作出相应广播操作
- 20) 部件 VocabularyDetection ,用于存储敏感词列表,同时监听出现的敏感词、作出相应通知
- 21) 部件 ZombieDetection ,为水军防御模块,可以监听用户的 IP 和操作内容,及时通知管理员
- 22) 部件 GoodsInfoDao ,为物品信息数据访问对象,用于进行数据库存取操作
- 23) 部件 BuyInfoDao ,为购买信息数据访问对象,用于进行数据库存取操作
- 24) 部件 CommentDao ,为评论信息数据访问对象,用于进行数据库存取操作
- 25) 部件 UserInfoDao ,为用户信息数据访问对象,用于进行数据库存取操作
- 26) 部件 VirtualCacheCluster ,为虚拟高频数据缓存集群,用于分布缓存被高频访问的数据,保证数据库存取性能,端口 resultSet 负责传递结果集,端口 cacheSwap 用于与数据库交互
- 27) 部件 DataUpdate ,用于进行安全的数据库数据更新,保证 ACID 原则、避免并发性错误,端口 updateData 用于接收更新数据,端口 updateInterface 为数据库更新接口
- 28) 部件 DataBaseManage ,负责管理数据库存储,端口 data 负责数据传输
- 29) 部件 DataStandardize ,为爬虫的数据标准化工具,负责统一各网站的数据格式
- 30) 部件 Crawler ,意为爬虫,可以自动爬取各网站的购物数据信息,并定时通知数据库进行数据更新
- 31) 连接件 ProcedureCall ,为简单的程序调用,caller 代表调用方, callee 代表被调用方

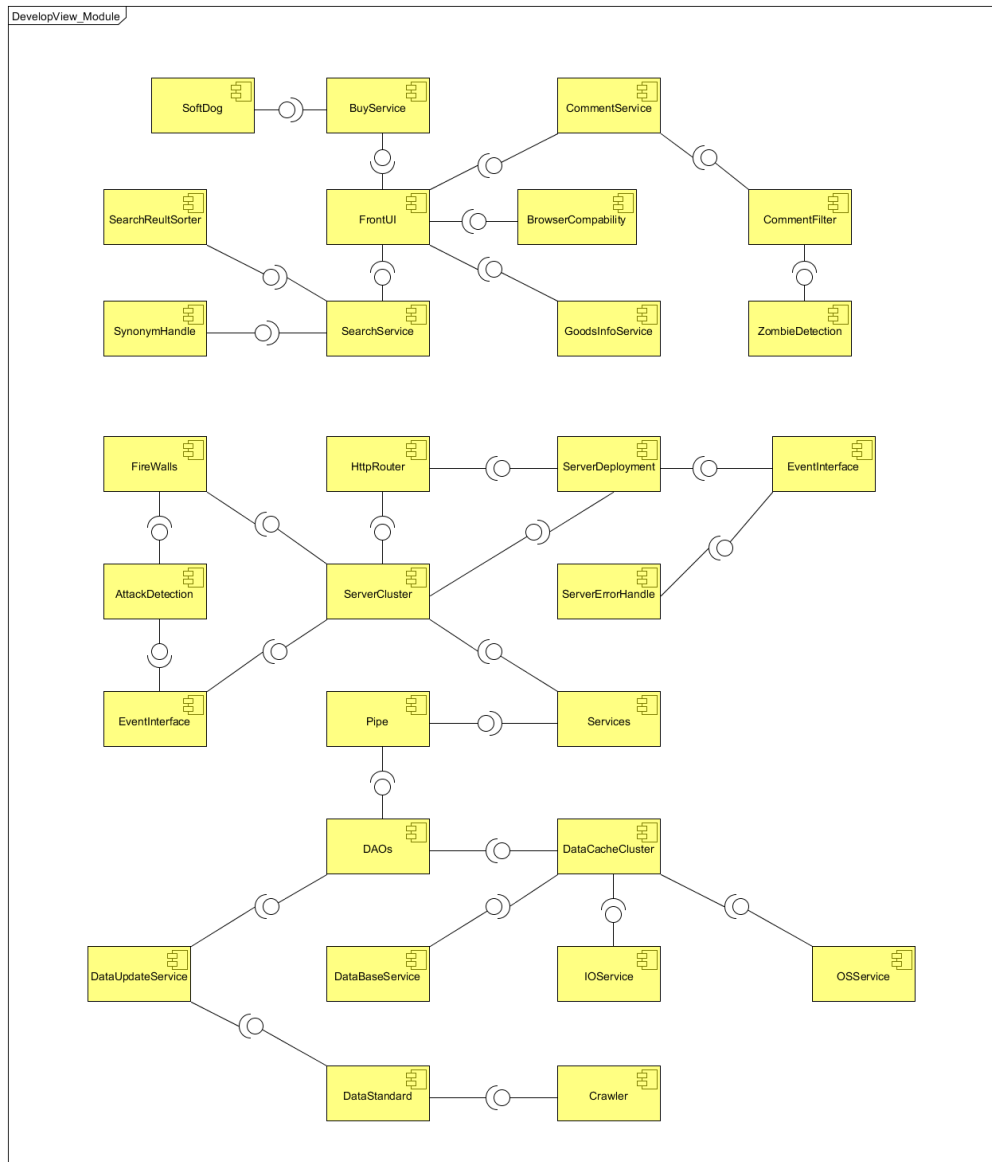
- 32) 连接件 RPC，描述了远程过程调用机制，remote=true 表示远程调用可用
- 33) 连接件 HttpRequest，描述了描述了 http 超文本通信协议，用于客户端浏览器和服务器通信，以两个角色 request-response 请求响应的方式相互通信
- 34) 连接件 EventRouter，描述了事件路由机制，由角色 publisher 声明并广播特殊时间，角色 listener 作出响应操作
- 35) 连接件 ShareData，描述了共享数据机制，角色 initializer 将设置的数据共享给一个或多个其他角色 user 使用
- 36) 连接件 Pipe，描述了数据传输中的管道机制

3.1.2 配置描述

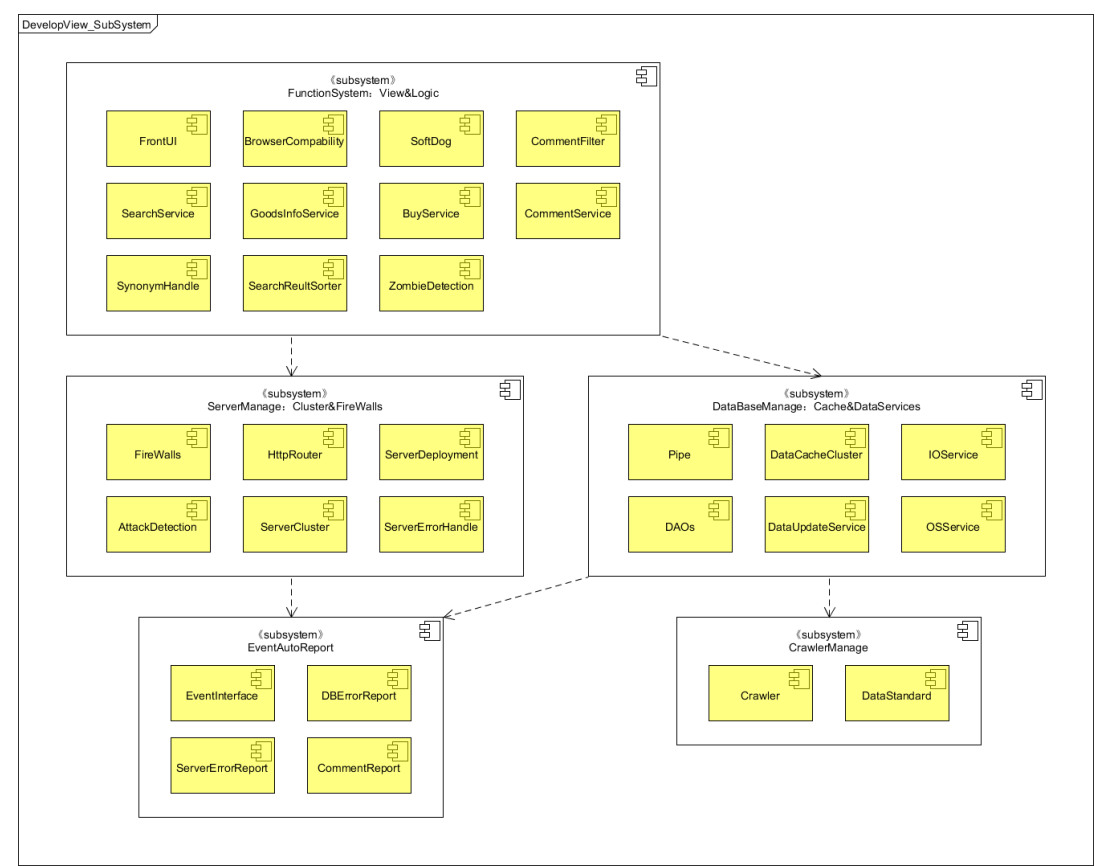


3.2 开发视图

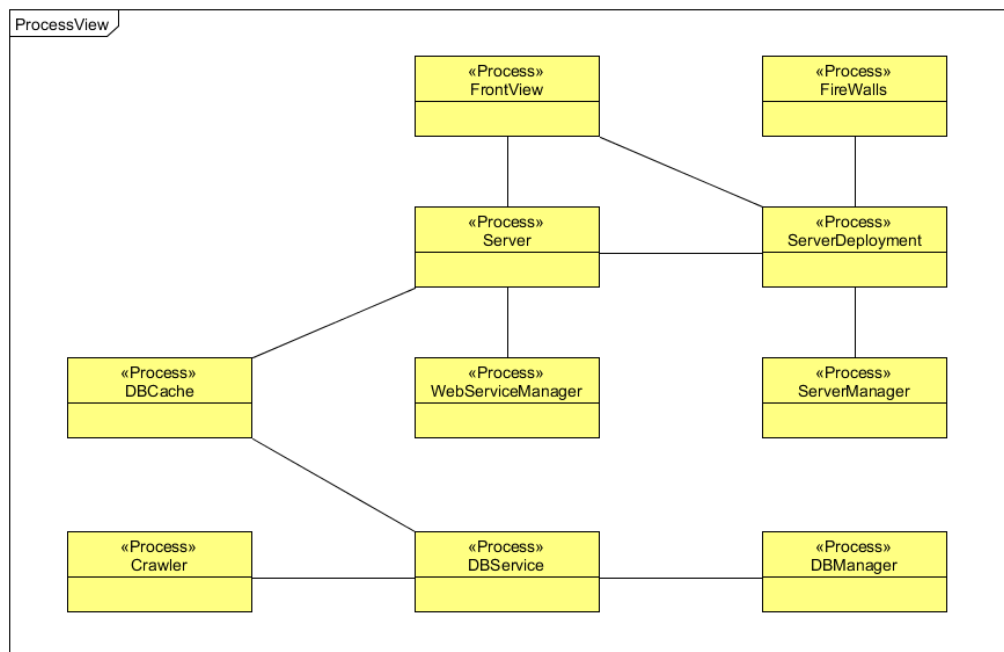
3.2.1 模块组织



子系统组织

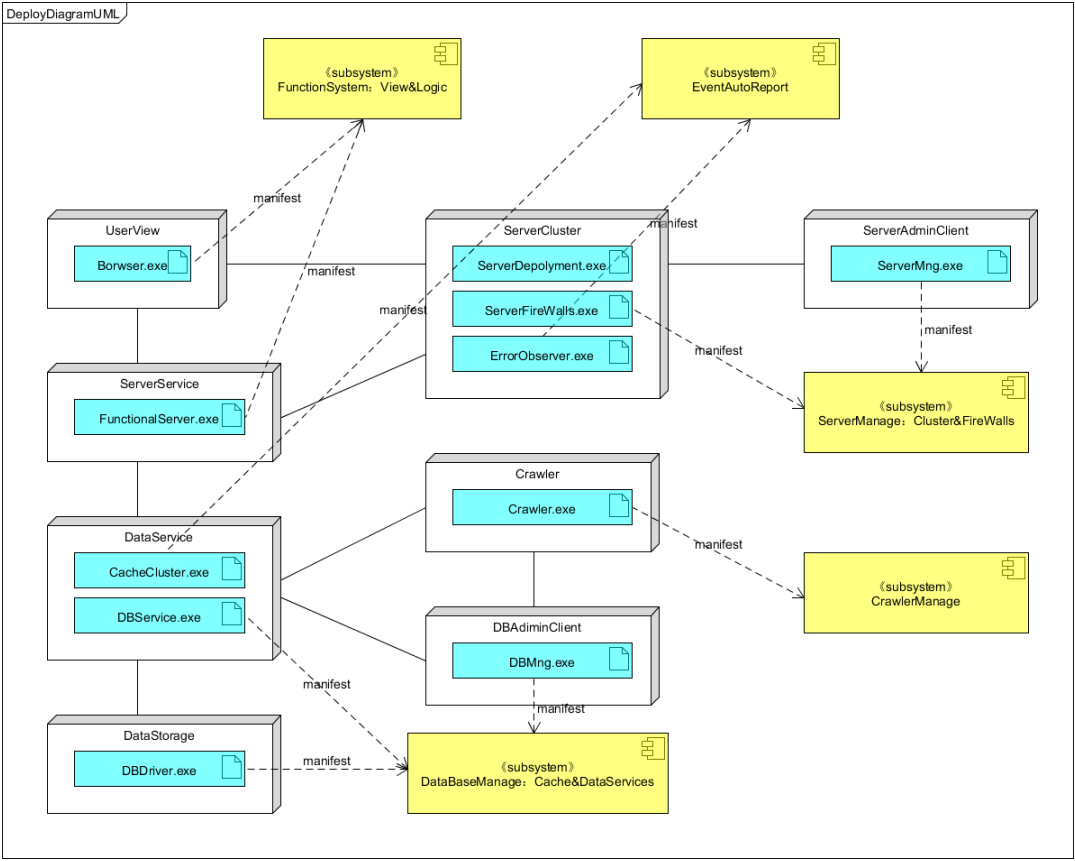


3.3 进程视图



进程	部件、连接件
FrontView	View、BrowserCompability、Bean、FrontAction、HttpRequest
Server	Service、Synonym、ResultFilter、SoftDog、BidInfo
ServerDeploymeny	HttpRouter、ServerCluster、ServerDeploy、RPC
FireWalls	AttackDefense、ShieldIPList、EventRouter
WebServiceManager	CommentFilter、VocabularyDetection、ZombieDetection、 ProcedureCall、EventRouter
ServerManager	ServerErrorReport、EventRouter
DBCache	VirtualCacheCluster、RPC
DBService	Pipe、Dao、DataUpdate
DBManager	DataBaseMange、PC
Crawler	Crawler、DataStandarize、EventRouter

3.4 部署视图



4 设计决策

4.1 关于风格的决策

4.1.1 数据抽象和面向对象风格

风格编号	Style1
相关功能	提供搜索、提供商品信息、提供评论平台、提供比价信息、用户管理、评论管理、用户与商家信息加密

质量要求	搜索功能高易用性、数据完整可靠、可拓展性、快速开发
决策依据	<ol style="list-style-type: none"> 1 面向对象风格有利于信息隐藏与模块化，实现程序的高内聚低耦合 2 数据抽象有利于进行模块的分解与对接，从而实现并行开发，提高开发效率与质量 3 开发人员易于理解，提高开发效率
影响视图	逻辑视图、开发视图

4.1.2 管道-过滤器风格

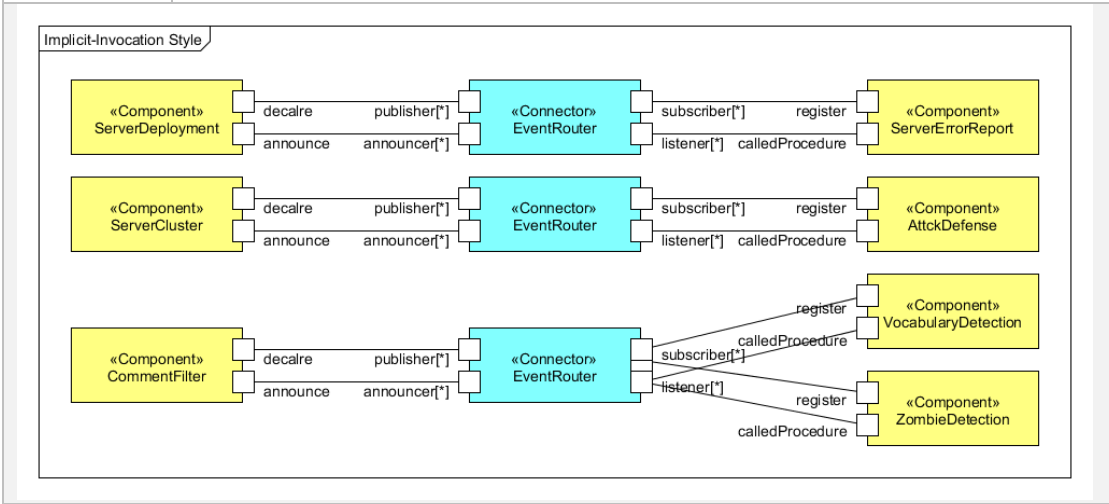
风格编号	Style2
相关功能	提供搜索、提供购买功能、用户管理、评论管理、用户与商家信息加密、搜索结果排序
质量要求	数据传输保证安全性、保证存取效率、良好的可拓展性、良好的可复用性、快速开发
决策依据	<ol style="list-style-type: none"> 1 过滤器并不关心上游和下游信息的标识，因此该风格具有良好的隐藏性和高内聚、低耦合的特点，适应于大型系统开发 2 将整个输入、输出行为视为多个过滤器的简单合成，有利于开发人员理解，提高开发效率和质量 3 过滤器和管道模块复用性高，可以较为灵活的组合，在大型系统中可有

	<p>效地避免重复代码</p> <p>4 允许进行一些吞吐量、死锁的分析，同时支持并行执行，适应于本高并发、高吞吐量的系统</p> <p>5 新的过滤器可简单替换旧的过滤器，有利于本系统的多次迭代开发过程</p>
影响视图	逻辑视图、开发视图

4.1.3 基于事件的隐式调用风格

风格编号	Style3
相关功能	插件扫描防御、IP 屏蔽、评论敏感词检测、服务器故障检测、数据库故障检测、水军检测
质量要求	敏感词可拓展、IP 屏蔽列表可拓展、故障快速响应、插件攻击快速响应、特定事件及时作出通知、快速开发
决策依据	<p>1 隐式调用风格为软件重用提供了强大的支持，只需将新的模块注册到系统的事件中而不需要修改事件发出者的部分，有利于系统的高效率、高质量开发</p>

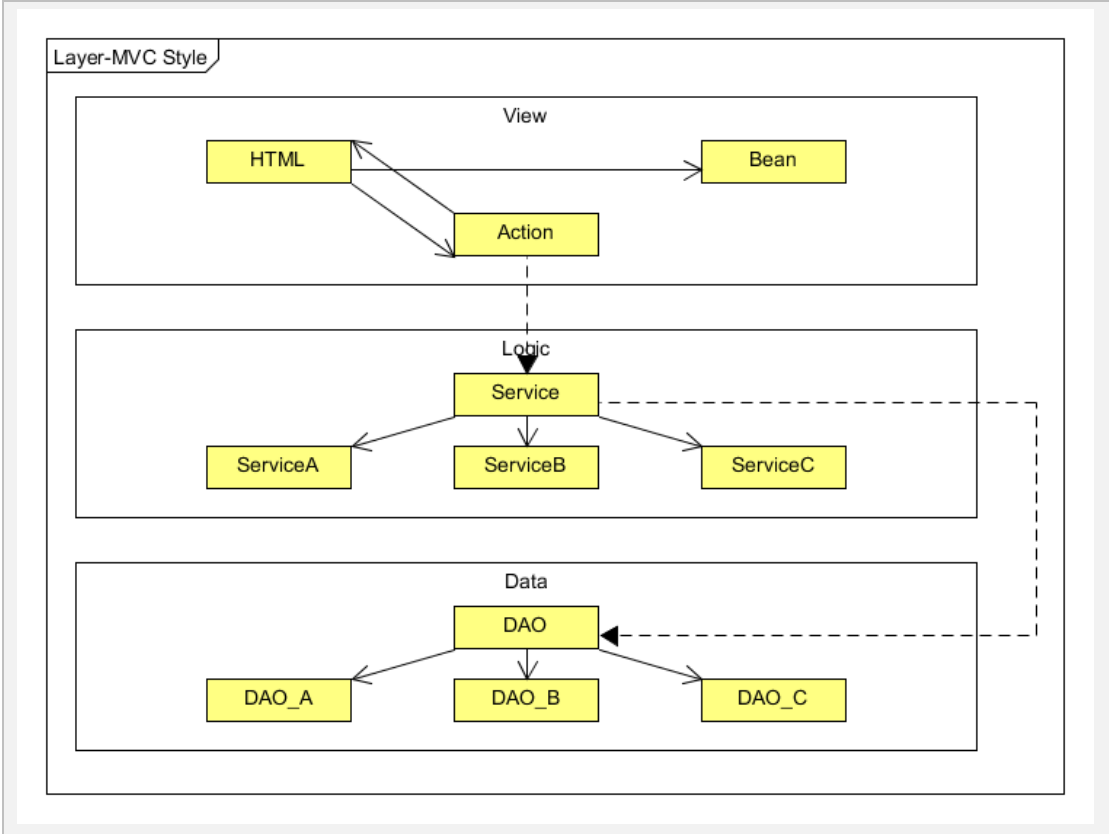
	<p>2 用一个模块替代另一个模块时,不会影响模块间的接口,在软件改进上的优越性有利于本系统的多次迭代开发过程</p> <p>3 本系统中有很多高优先级的需求,如故障检测、攻击检测、敏感词检测等,若使用隐式调用风格,形成事件的发起者-监听者的关系,既可以实现良好的封装,又便于开发者理解,同时保证系统在一些关键事件监听上的实时性与独立性</p> <p>4 隐式调用风格中,将故障检测、攻击检测等模块独立成单独的进程,实现系统各组件的独立性和并行性,提升可靠性和性能</p>
影响视图	逻辑视图、开发视图、进程视图、部署视图



4.1.4 分层+MVC 风格

风格编号	Style4
相关功能	提供搜索、提供商品信息、提供评论平台、提供比价信息、提供购买渠道、用户管理、评论管理
质量要求	快速开发、服务器高性能、服务器高可靠性、数据库高性能、数据库高可靠

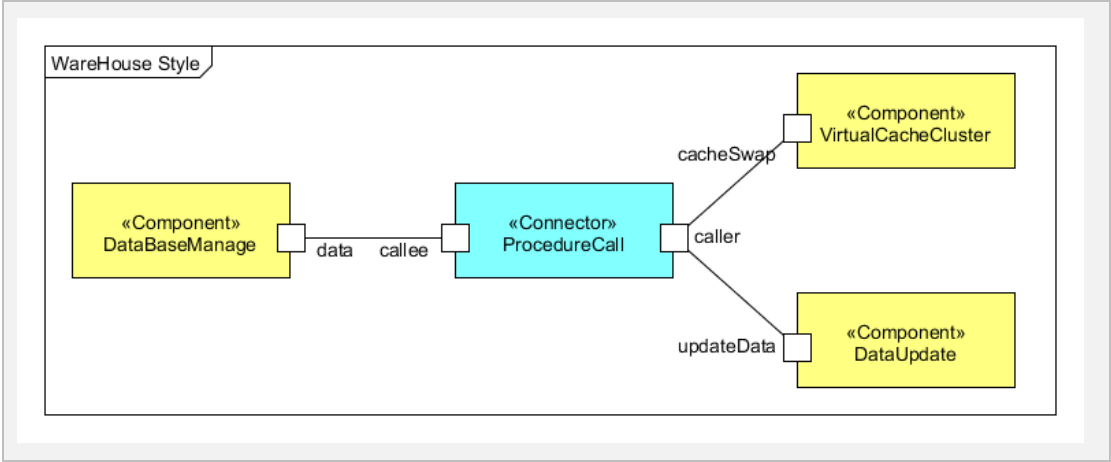
	性、保证系统的易用性良好
决策依据	<ol style="list-style-type: none"> 1 层次结构可以将本复杂系统按递增的步骤进行分解，有利于开发人员的理解，提高开发效率与质量 2 支持重用、拓展性良好 3 简单的分层结构对于本系统来说依然过于抽象，将分层风格与 MVC 风格相结合，细化调用关系，将系统的高层设计在 VIEW-LOGIC-DATA 的基础上更加层次化
影响视图	逻辑视图、开发视图



4.1.5 仓库风格

风格编号	Style5
相关功能	数据库管理、数据库故障检测、数据库及时更新

质量要求	<p>保证数据库存取效率、保证数据库可靠性、保证数据库对高并发的适应性、</p> <p>保证系统实时性</p>
决策依据	<p>1 分离数据源与控制，实现系统的高可靠性</p> <p>2 提高系统的可拓展性，只需要在数据源的基础上增加新的控制模块</p> <p>3 利用并行提高数据的性能，保证其在高峰期的吞吐量</p>
影响视图	逻辑视图、开发视图

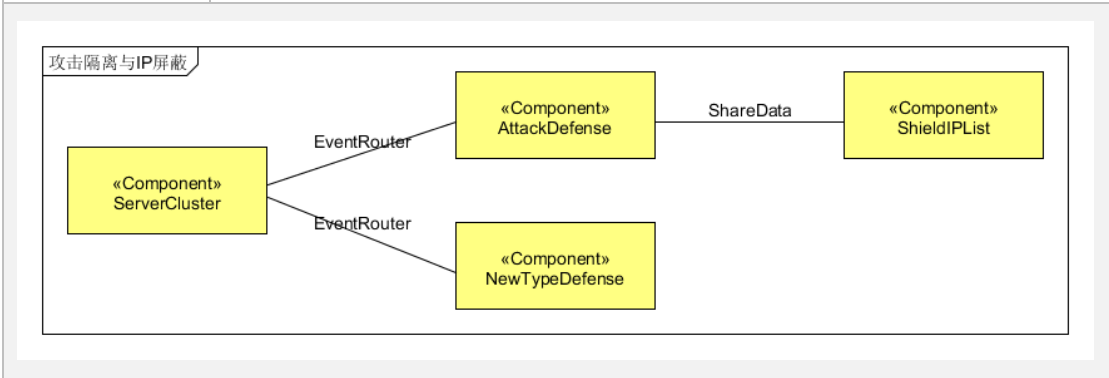


4.2 相关设计决策

4.2.1 设计决策 1——攻击隔离与 IP 屏蔽

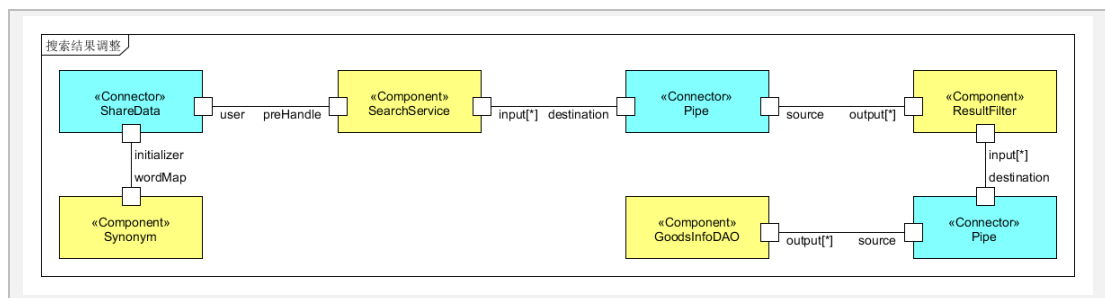
需求&约束	<p>R16：实现入侵检测和攻击隔离功能</p> <p>C32：自动检测各类入侵</p> <p>C33：自动隔离相应 IP</p> <p>C34：可拓展的检测和隔离模块</p>	
候选方案	CS1.1	<p>封装一个访问检测模块，每次请求在路由时先交给该模块进行</p> <p>过滤</p>

	CS1.2	利用隐式调用机制,将 Http 请求注册为事件,服务器作为事件的 announcer,而将扫描防御模块作为事件的 listener 进行监听,若发生符合条件的事件再进行检测与相应处理
选择选项	CS1.2	
选择理由	<ol style="list-style-type: none"> 1. 封装单独的过滤模块不利于 HTTP 请求过滤方式的拓展 2. 每次请求均进行一次完整的检验操作影响服务器的响应性能,可能会阻碍保证系统实时性的约束 3. 利用隐式调用机制分离服务器的提供服务和安全监测部分,保证系统组织的高内聚性,同时只需增加新的 listener 便可实现拓展,因此使检测方式拥有良好的可拓展性 4. 隐式调用机制分离出多个独立的进程并行操作,可以有效提高服务器的性能,保证系统实时性 	
影响	逻辑视图、开发视图、进程视图	
详细设计约束	<ol style="list-style-type: none"> 1. 封装每个单一的检测模式的模块,保证高内聚性 2. 面向接口编程,保证新增检测模块时不影响服务器模块本身 	



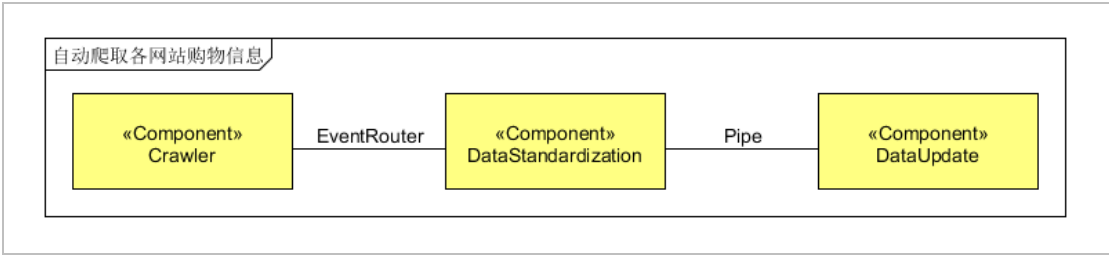
4.2.2 设计决策 2——搜索结果调整

需求&约束	R1：提供高易用性搜索功能 R2：搜索结果可调整 C1：兼容同义词，维护词汇表 C2：及时屏蔽负面商品 C3：结果可排序 C4：允许竞价调整顺序 C5：排序方案可拓展	
候选方案	CS2.1	维护同义词表，搜索前进行预处理，同时建立排序模块，提供各排序方式的接口，提供负面商品屏蔽模块，可过滤负面商品
	CS2.2	使用管道-过滤器机制，在数据层与逻辑层中通过管道连接一个可拓展的过滤器部件，将排序工作交给过滤器部件实现，同时利用数据共享机制，持有动态更新的负面商品信息，并进行相应的过滤、屏蔽操作
选择选项	CS2.1的同义词表结合 CS2.2 的过滤器思想	
选择理由	需求和约束中要求的搜索结果处理类型不同且各有特点，拆分的粒度太细导致难以理解，为搜索服务模块提供过多的接口又不利于信息隐藏和数据封装。因此可以将搜索服务模块的接口抽象为搜索前预处理与一个与管道相连的 input-output 接口，而排序、屏蔽工作则交给管道中间的过滤器执行，不再由搜索服务模块显示调用。这样的封装在保证良好的拓展性的同时，又避免搜索服务模块本身的封装性受到破坏	
影响	逻辑视图、开发视图	
详细设计约束	1. 同义词表通过表驱动和 Map 的形式表现 2. 过滤器部分面向接口编程，保证模块内部不依赖于数据本身的标识，而只关心对数据自身的操作	



4.2.3 设计决策 3——自动爬取各网站购物信息

需求&约束	<p>R7：可兼容可拓展的信息爬取模块</p> <p>C13：兼容不同网站的数据格式</p> <p>C14：自动爬取，定时更新</p>	
候选方案	CS3.1	<p>将信息爬取作为独立可自主运作的模块，而不依赖于其他模块的调用，分离出单独的数据标准化模块，再提供单独的数据库更新模块用于不影响正常访问的数据更新过程</p>
选择选项	选择 CS3.1	
选择理由	<p>爬虫功能需求有一定的自主性和智能型，因此作为单独的模块运行较为合理，而分别封装爬取和标准化部分有利于高内聚低耦合的开发，而本系统对数据的更新要求很高，同时要求不能妨碍用户正常使用，所以有必要拆分出单独的数据更新模块</p>	
影响	逻辑视图、开发视图、进程视图	
详细设计约束	<ol style="list-style-type: none"> 1. 严格规定各模块间的交互接口 2. 数据更新模块保证数据库操作的 ACID 原则 	



4.2.4 设计决策 4——高可靠性服务器系统

需求&约束	R11：实现高可靠性服务器系统 C22：故障一小时内发现 C23：故障时不影响正常访问	
候选方案	CS4.1	对服务器集群设置单独的管理模块，访问时进行服务器状态检测与服务器分配操作
	CS4.2	利用隐式调用机制，将服务器集群的配置状态设置为树状的结构，将叶节点的丢失作为恶劣事件注册，此时，服务器集群配置模块作为 announcer，同时使用新的故障检测模块作为 listener，监听集群中的节点丢失事件，若发生则可以实时响应并作出相应处理
选择选项	CS4.2	
选择理由	1. 考虑到服务器对高并发时的性能具有很高的要求，若将检测子程序作为集群管理模块的一部分，每次配置时进行检测会造成服务器集群的响应速度受损 2. 利用隐式调用机制可以分离出独立的监听进程，利用并行的进程提高服务器响应性能、保证系统实时性	

	<p>3. 并行进程在监听故障时不会受到故障本身影响，因此在服务器集群管理上可以体现较高的可靠性</p>
影响	逻辑视图、开发视图、进程视图
详细设计约束	<p>1. 故障检测模块遵循隐式调用的接口规则</p> <p>2. 故障检测模块需作为单独的进程独立运行</p>

高可靠性服务器系统

```

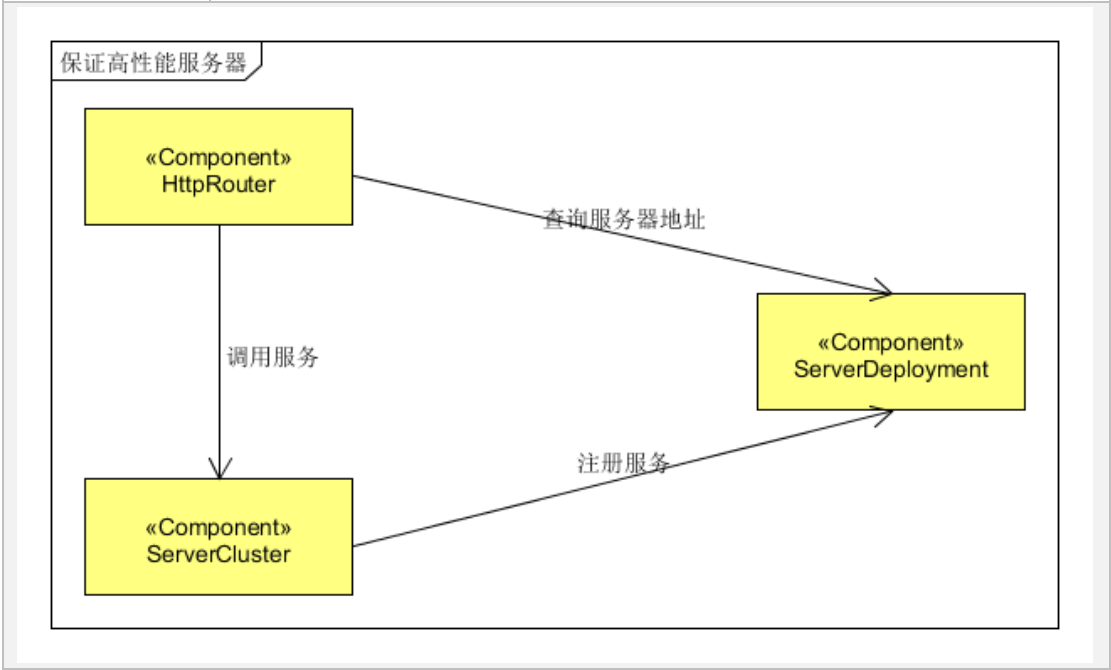
graph LR
    SD["«Component»  
ServerDeployment"]
    ER["«Connector»  
EventRouter"]
    SER["«Component»  
ServerErrorReport"]

    SD -- decare --> ER
    SD -- announce --> ER
    ER -- subscriber[*] --> SER
    ER -- listener[*] --> SER
    ER -- publisher[*] --> SD
    ER -- announcer[*] --> SD
    SER -- register --> ER
    SER -- calledProcedure --> ER
  
```

4.2.5 设计决策 5——高性能服务器系统

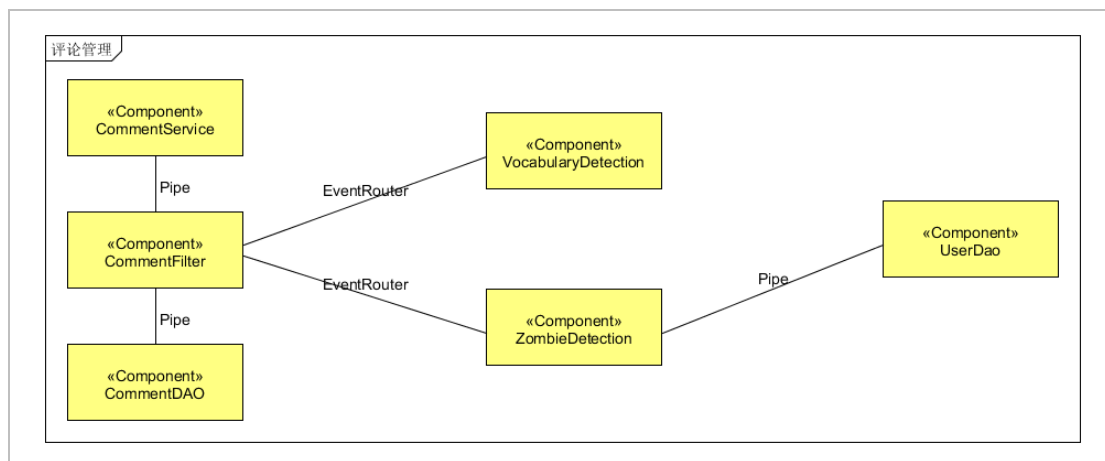
需求&约束	<p>R12：实现高性能服务器系统</p> <p>C24：高峰时容纳 500 万用户</p> <p>C25：不发生并发性错误</p>	
候选方案	CS5.1	使用服务器集群，利用 hashmap 的计算方式将每个 HTTP 请求分散给不同的服务器执行，减少单个服务器的运作压力
	CS5.2	设立单独的路由分配模块、服务信息配置模块和服务器集群模块。服务器集群模块向配置模块注册信息，路由分配模块向信息配置模块请求一个服务器地址并缓存，此后每次直接使用缓存的地址访问相应的集群中的服务器
选择选项	CS5.2	
选择理由	<p>1 简单的 hashmap 分配法虽然简单可行，并可以有效利用服务器集群，但这种数学算法在保证运算效率的同时拥有一定的偶然性，若</p>	

	<p>出现瞬间的高并发事件，会有请求往同一服务器堆积导致集群崩溃的概率，这是应该避免的</p> <p>2 分离服务器集群本身和服务器信息配置模块，使服务器本身专注于提供服务，而服务信息配置模块则负责引导路由和进行集群管理，从而提升系统性能</p> <p>3 路由模块缓存申请到的服务器地址，避免每次请求都向信息配置模块查询，影响系统性能</p>
影响	逻辑视图、开发视图
详细设计约束	<p>1 服务信息配置模块使用树结构，利用节点控制服务器与客户端请求的分配状况</p> <p>2 保证服务器集群的接口只专注于本身提供的服务，保证逻辑组件的封装性良好</p>



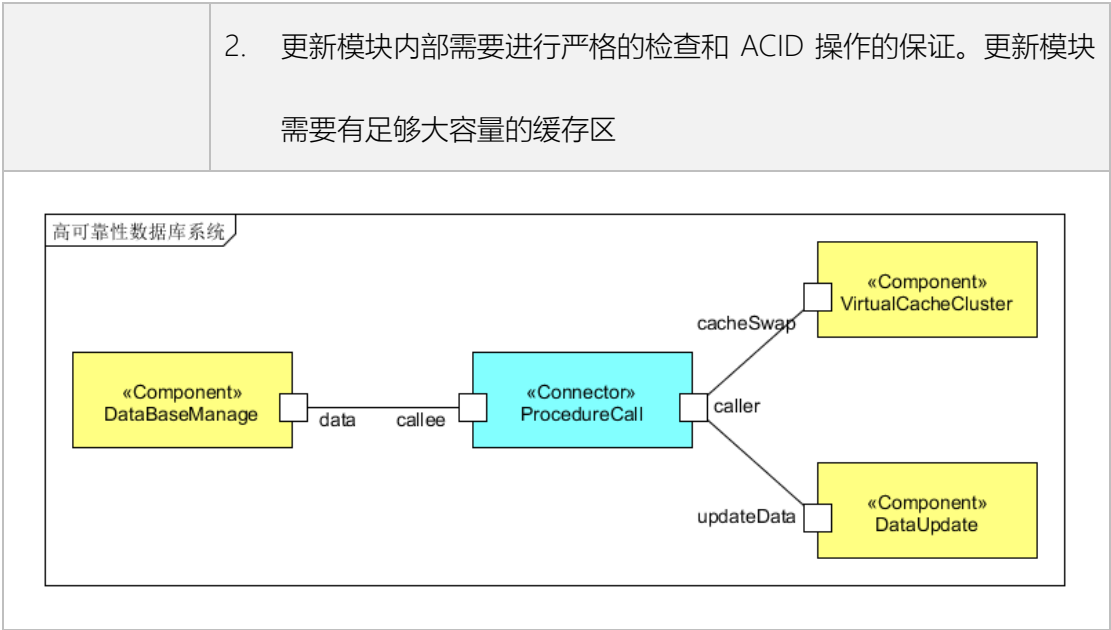
4.2.6 设计决策 6——评论管理

需求&约束	R5：完备的评论管理系统 C9：自动检测敏感词汇 C10：自动化的模块通知	
候选方案	CS6.1	使用管道-过滤器机制，为评论信息数据连接一个过滤器进行敏感词检测，再将检测到敏感词的事件进行注册，为其增添相应的监听模块。使用隐式调用机制通知作出相应的通知或处理操作。再为水军检测模块增添一个管道与用户数据实体对象进行交互。
选择选项	CS6.1	
选择理由	管道-过滤器机制与隐式调用的优势已在之前的决策中有过多次阐述。 在评论管理的敏感词检测中，使用过滤器分离评论服务本身与对其的检测，再利用隐式调用保证检测部分良好的可拓展性。数据传输部分使用管道-过滤器可以实现良好的信息隐藏和模块封装	
影响	逻辑视图、开发视图、进程视图	
详细设计约束	1. 严格面向接口编程 2. 引用关系避免相互依赖	



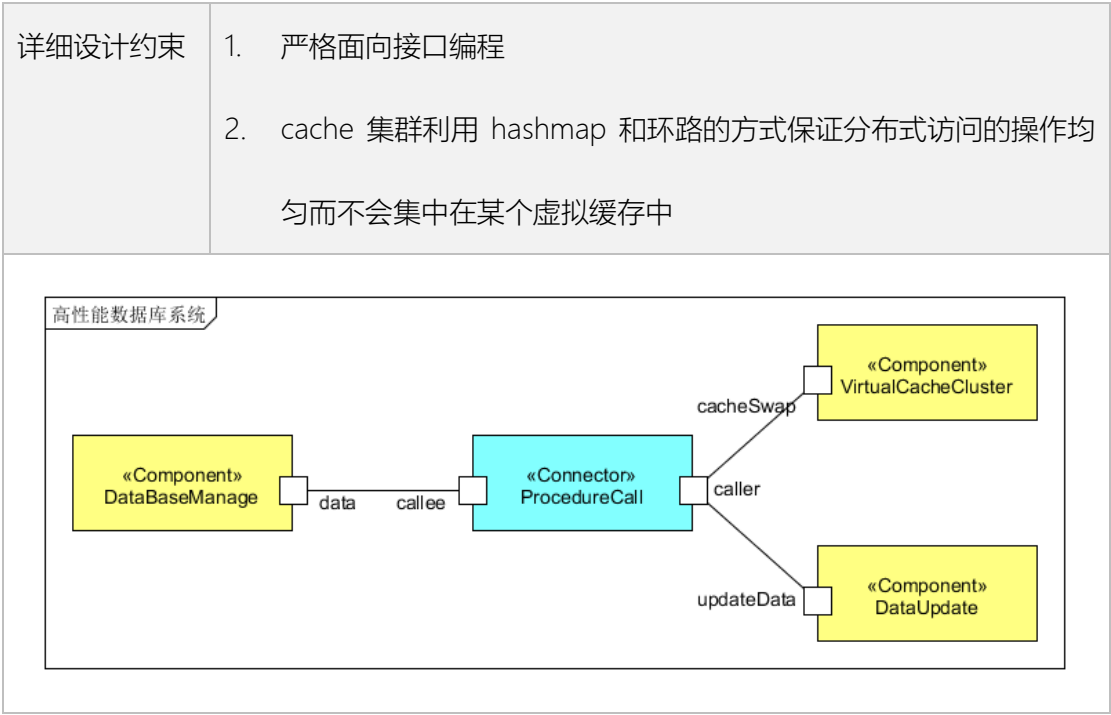
4.2.7 决策 7——高可靠性数据库系统

需求&约束	R11：实现高可靠性数据库系统 C22：故障 1 小时内发现 C23：故障时不影响访问	
候选方案	CS7.1	使用仓库风格，将数据库模块本身与数据库更新模块分离，数据库更新模块用于对更新数据进行缓存、检测、在保证 ACID 原则的同时进行安全更新。同时提供一个虚拟数据缓存模块，作为数据与 DAO 层的中介
选择选项	CS7.1	
选择理由	无论如何将数据库本身暴露在外是极不安全的。因此选择为其增加两个功能模块用于提供数据的访问和更新功能来保证数据库的安全性和可靠性。同时独立更新进程，从而实现数据更新时不影响用户正常访问的约束	
影响	逻辑视图、开发视图、进程视图	
详细设计约束	1. 严格面向接口编程	



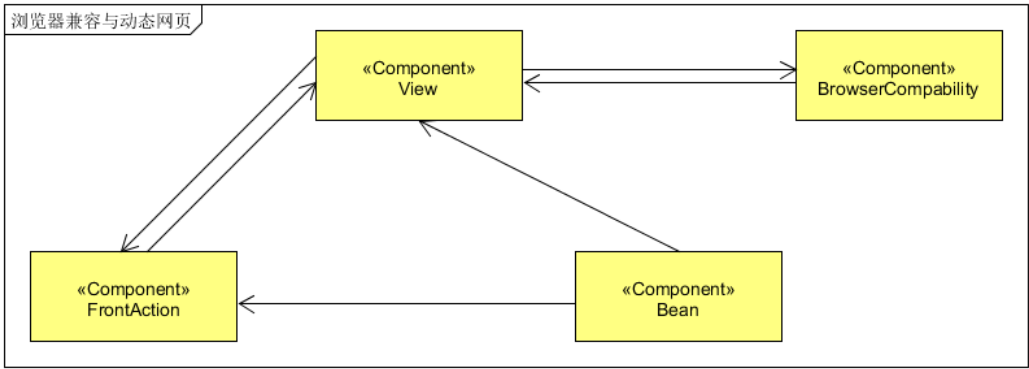
4.2.8 决策 8——高性能数据库系统

需求&约束	R10：实现高性能数据库系统 C20：容纳至少 1000 万条数据 C21：保证数据快速存取	
候选方案	CS8.1	在决策 7 的基础上，将虚拟数据模块拓展为虚拟高频数据缓存模块，模仿计算机系统中的 cache 缓存和数据交换原理和分布式数据存储系统，优化数据库的存取速度
选择选项	CS8.1	
选择理由	考虑到约 2000 条数据被频繁访问，使用一个模仿 cache 缓存的结构进行快速存取和数据库的数据交换操作可以有效地提高存取效率。同时还可以建立分布式访问的 cache 集群。利用这些优化操作提升使数据库性能得以提升	
影响	逻辑视图、开发视图	



4.2.9 决策 9——浏览器兼容与动态网页

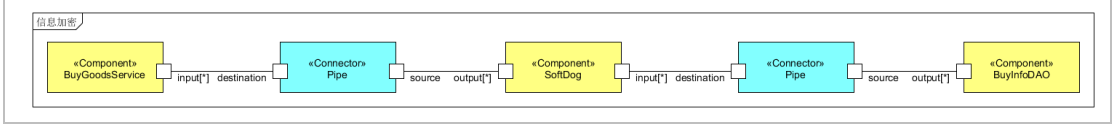
需求&约束	R14：兼容多浏览器	
	C29：兼容方式可更新	
候选方案	CS9.1	分离兼容模块和前端展示模块，利用程序调用机制动态适应不同的浏览器
	CS9.2	将前端分离为展示模块 View，控制用户接口动作的模块 Action、存储数据的模块 Bean 和浏览器兼容模块。浏览器兼容仅和展示模块 View 相交互，提供不同浏览器的兼容配置信息
选择选项	CS9.1 结合 CS9.2	
选择理由	1	前端的数据格式冗余且复杂，简单的两个模块会造成浏览器兼容子程序的解析压力增加，并可能影响组件刷新效率
	2	分离展示组件、数据、动作监听可以使前端的编码规范化且格式化，

	减少兼容模块的交互以提高性能
影响	逻辑视图、开发视图
详细设计约束	1 使用jsp、blade、Vm 等模板替换技术实现数据与组件的分离
 <pre> graph TD View["«Component» View"] <--> BrowserCompability["«Component» BrowserCompability"] View <--> FrontAction["«Component» FrontAction"] Bean["«Component» Bean"] --> FrontAction </pre>	

4.2.10 决策 10——信息加密

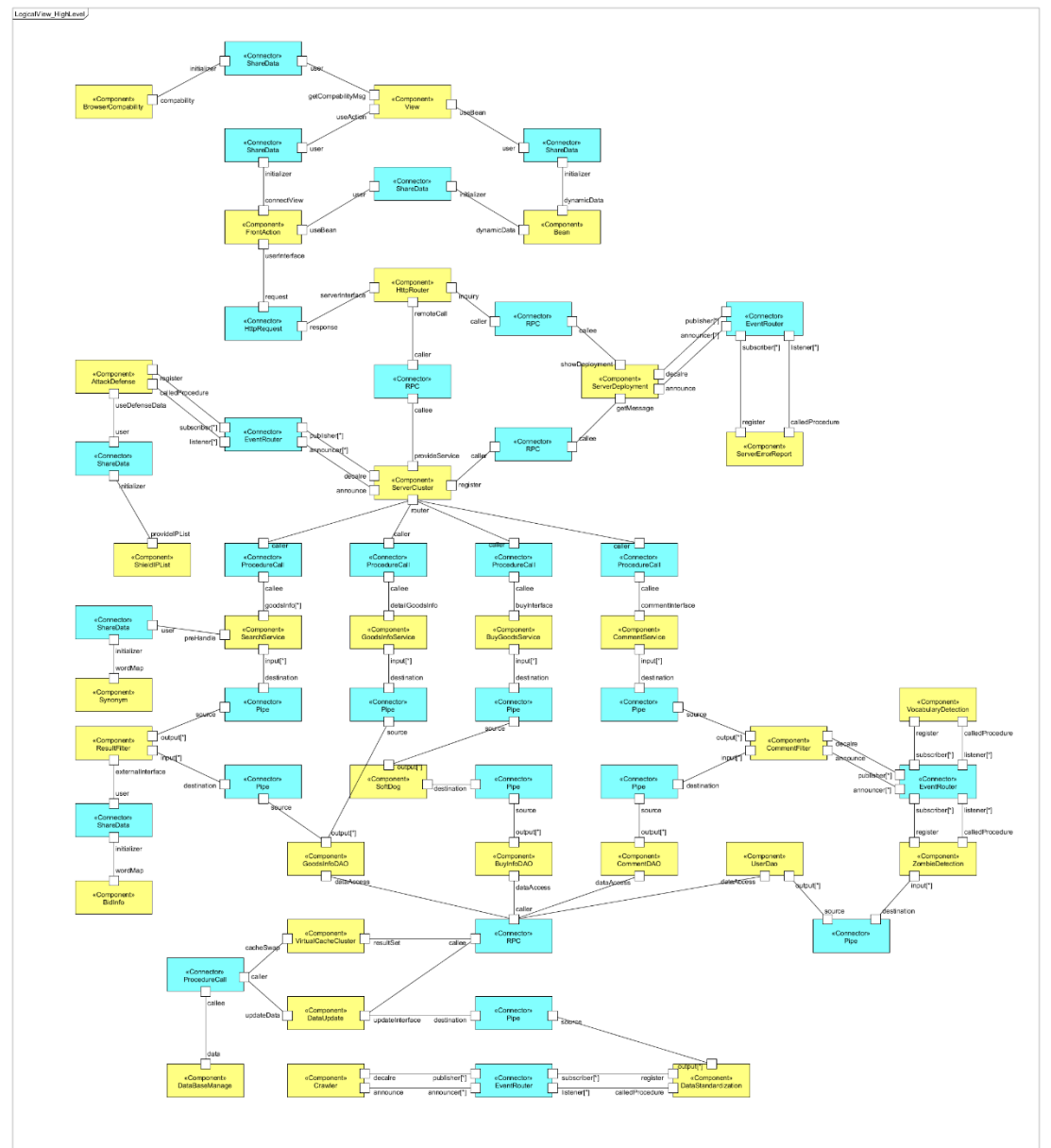
需求&约束	R23：保证系统数据和信息安全 C47：数据库信息加密 C48：加密商家和用户信息	
候选方案	CS10.1	使用管道-过滤器机制对购买商品的数据在传输过程中进行加密，并严格封装数据库的访问接口
选择选项	CS10.1	
选择理由	管道-过滤器机制具有良好的可拓展性和封装性，因此在保证安全的能力上有良好的表现，使用加密模块加密由数据库传递的数据来保证商家和用户的信息不回直接泄露	
影响	逻辑视图、开发视图	
详细设计约束	1. 严格面向接口编程	

2.	保证加密算法安全可靠
----	------------



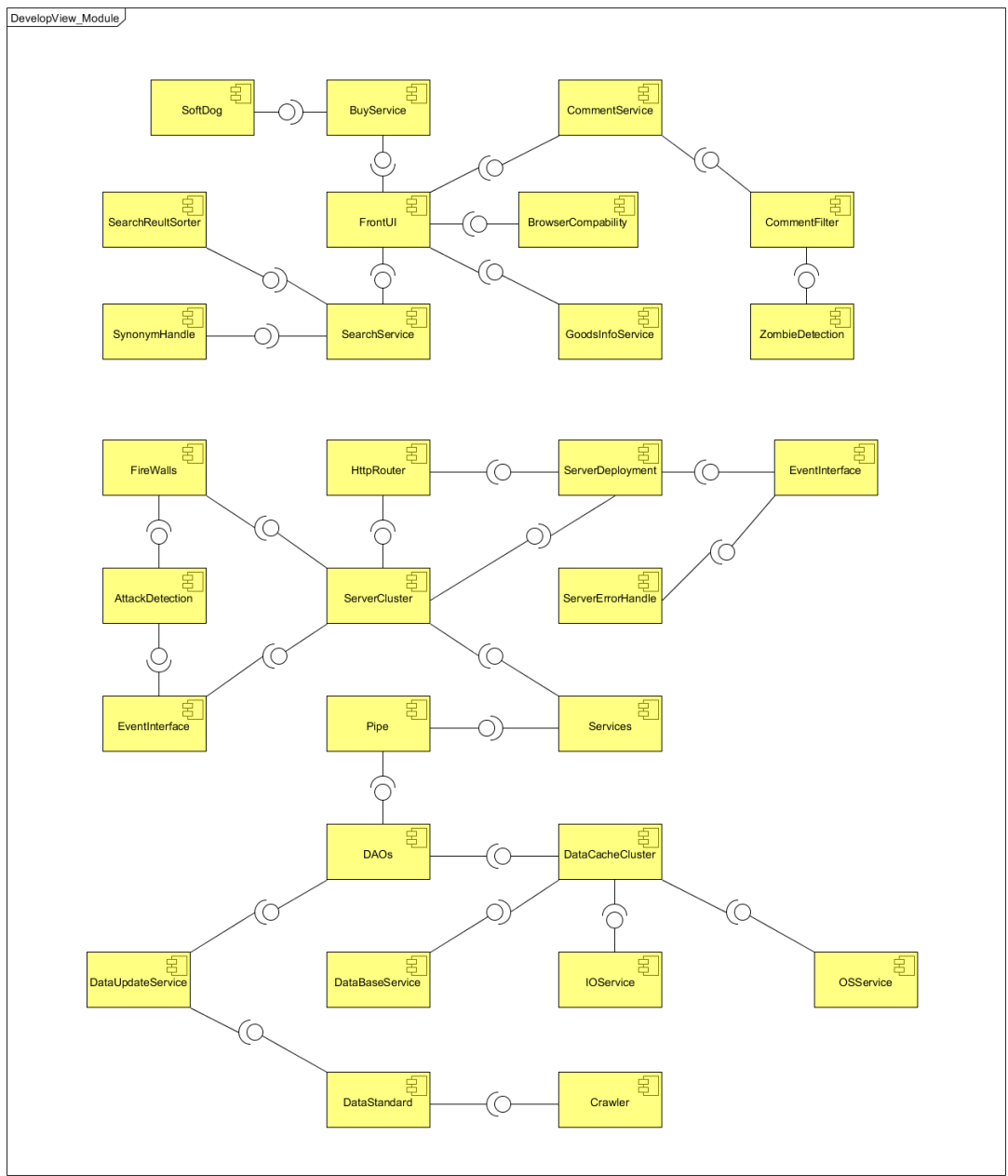
5 最终高层结构

5.1 逻辑视图



5.2 开发视图

5.2.1 模块描述



5.2.2 接口定义

开发视图中各构件间的接口定义如下：

模块名	FrontUI
-----	---------

供给接口			
ID	display	功能描述	将前端的网页信息进行展示
		语法	Void display(xml);
		前置条件	无
		后置条件	无
		不变量	只对数据信息进行展示，而不会改变数据内容
需求接口			
Httpservice、BrowserCompability.comp			

模块名		BrowserCompabilty	
供给接口			
ID	comp	功能描述	处理 xml 数据使其兼容不同浏览器
		语法	Xml borwserName(xml);
		前置条件	待兼容数据符合 xml 格式
		后置条件	向 UI 返回兼容处理后的数据
		不变量	数据信息
需求接口			
无			

模块名		SearchService	
供给接口			
ID	search	功能描述	提供商品搜索服务

		语法	Void search(keyword,sort);
		前置条件	无
		后置条件	返回搜索结果集
		不变量	无
需求接口			
SearchResultSorter.sort、SynonymHandle.handle、DAO			

模块名		SearchResultSorter	
供给接口			
ID	sort	功能描述	对集合进行排序
		语法	Set sort(set, sort);
		前置条件	输入数据格式为集合
		后置条件	返回排序后的集合
		不变量	无
需求接口			
无			

模块名		SynonymHandle	
供给接口			
ID	syn	功能描述	提供关键词同义词表
		语法	Set getSyn(keyword);
		前置条件	无

		后置条件	返回同义词集合
		不变量	同义词映射表
需求接口			
无			

模块名		GoodsInfoService	
供给接口			
ID	show	功能描述	提供商品信息展示服务
	Info	语法	Info showGoodsInfo(goodsId);
		前置条件	输入的 ID 为正整数
		后置条件	返回相应商品详细信息
		不变量	无
需求接口			
DAO			

模块名		BuyService	
供给接口			
ID	buy	功能描述	提供商品购买服务
		语法	Info buy(goodsId);
		前置条件	输入的 ID 为正整数
		后置条件	返回相应的商品购买信息和链接
		不变量	无

需求接口
Dao

模块名		SoftDog	
供给接口			
ID	encrypt	功能描述	提供商品搜索服务
		语法	Info encrypt(Info);
		前置条件	无
		后置条件	返回加密后数据信息
		不变量	无
需求接口			
无			

模块名		CommentService	
供给接口			
ID	comme nt	功能描述	提供评论服务
		语法	Void comment(message);
		前置条件	评论信息不能为空
		后置条件	处理评论信息，作出相应过滤后更新至数据库
		不变量	无
ID	show	功能描述	展示评论信息
		语法	Info showComment(goodsId);

		前置条件	Id 为正整数
		后置条件	返回某商品的评论信息
		不变量	无
		需求接口	
CommentFilter.filt、 DAO			

模块名		CommentFilter	
供给接口			
ID	filte	功能描述	过滤评论信息
		语法	Info filte(message);
		前置条件	无
		后置条件	返回过滤后的评论信息
		不变量	无
需求接口			
DAO、ZombieDetection.detect			

模块名		ZombieDetection	
供给接口			
ID	detect	功能描述	进行评论水军检测
		语法	Result detect(messgae);
		前置条件	无
		后置条件	返回水军检测结果

		不变量	无
需求接口			
DAO			

模块名		HttpRouter	
供给接口			
ID	route	功能描述	对 HTTP 请求进行路由
		语法	Response route(request);
		前置条件	请求符合 HTTP 协议
		后置条件	服务器进行相应服务操作后的返回值
		不变量	无
需求接口			
ServerCluster.service、ServerDeployment.inquiry			

模块名		SearchCluster	
供给接口			
ID	service	功能描述	提供服务器服务
		语法	Response service(resquest);
		前置条件	请求符合 HTTP 协议
		后置条件	服务器进行相应操作后的返回值
		不变量	ClusterStateMessage
需求接口			

Services、FireWalls.detect、EventInterface.announcer
--

模块名		ServerDeployment	
供给接口			
ID	inquiry	功能描述	提供服务器集群分配查询功能
		语法	IP inquiry(clientMessage);
		前置条件	无
		后置条件	分配的服务器的 IP 地址
		不变量	无
ID	register	功能描述	提供服务器集群服务注册功能
		语法	Void register(serverMessage);
		前置条件	无
		后置条件	服务器成功注册
		不变量	无
需求接口			
EventInterface.announcer			

模块名		SearchErrorHandle	
供给接口			
ID	handle	功能描述	监听服务器故障并作出相应处理
		语法	Void handleAndReport(ServerMessage);
		前置条件	无

		后置条件	进行相应的故障处理和通知
		不变量	无
需求接口			
EventInterface.listener			

模块名		FireWalls	
供给接口			
ID	detect	功能描述	防火墙功能，入侵检测，屏蔽插件攻击
		语法	Message detect(request);
		前置条件	发生插件扫描攻击或特定 IP 入侵
		后置条件	返回防御过程的相关数据信息
		不变量	IPList
需求接口			
AttackDetection.detect			

模块名		AttackDetection	
供给接口			
ID	detect	功能描述	提供网络攻击防御功能
		语法	Void detect(request);
		前置条件	发生高频访问入侵时
		后置条件	无
		不变量	无

需求接口
EventInterface.listener

模块名		DataCache	
供给接口			
ID	get	功能描述	提供高频数据缓存
		语法	Data get(message);
		前置条件	数据检索条件输入正确
		后置条件	数据结果集
		不变量	无
需求接口			
DataBaseService.select			

模块名		DataUpdateService	
供给接口			
ID	update	功能描述	提供数据库更新服务
		语法	Void update(data);
		前置条件	数据格式正确
		后置条件	数据成功更新至数据库
		不变量	无
需求接口			
DataBaseService.update			

模块名		DataBaseService	
供给接口			
ID	update	功能描述	提供数据库更新服务
		语法	Void update(data);
		前置条件	无
		后置条件	数据成功更新至数据库
		不变量	无
ID	select	功能描述	提供数据库查询服务
		语法	ResultSet select(sql);
		前置条件	无
		后置条件	返回查询结果集
		不变量	无
需求接口			
无			

模块名		Crawler	
供给接口			
ID	send	功能描述	提供自动爬取的数据信息
		语法	Message send();
		前置条件	无
		后置条件	返回一段时间内爬取的数据信息

		不变量	无
需求接口			
无			

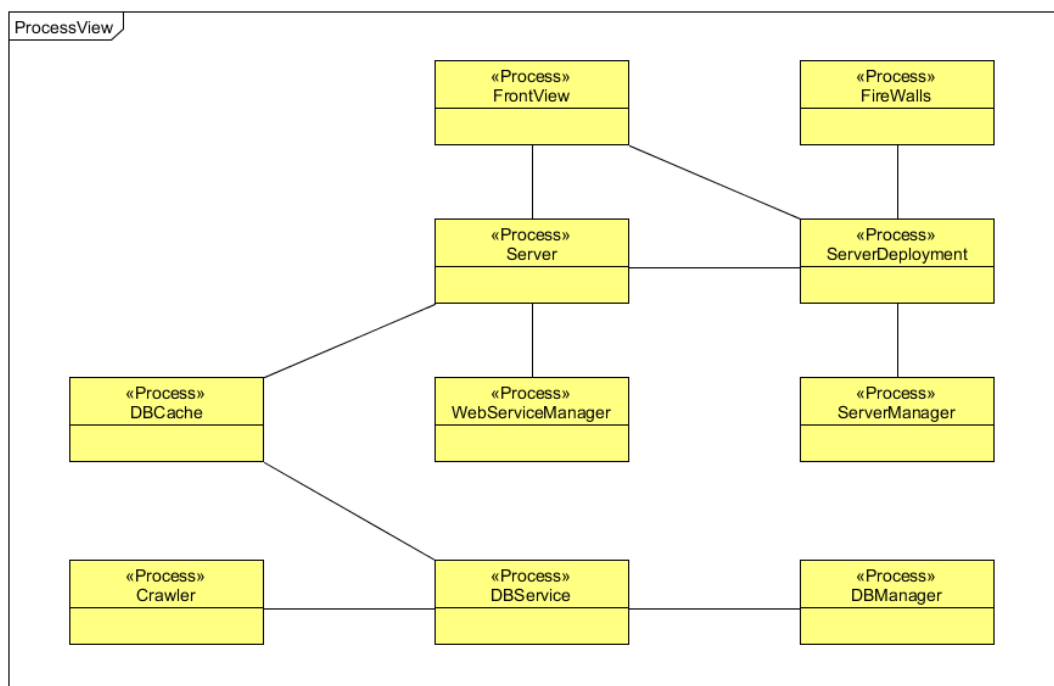
模块名		DataStandard	
供给接口			
ID	sd	功能描述	数据标准化
		语法	Result standard(message);
		前置条件	无
		后置条件	返回标准化后的数据
		不变量	无
需求接口			
Crawler.send			

模块名		EventInterface	
供给接口			
ID	listener	功能描述	注册事件监听
		语法	Void listen(event);
		前置条件	无
		后置条件	成功注册事件
		不变量	无
ID	announ	功能描述	注册事件广播

	ce	语法	Void announce(event);
		前置条件	无
		后置条件	成功注册事件广播
		不变量	无
	需求接口		
无			
模块名		DAOs	
供给接口			
ID	data	功能描述	提供数据持久化对象
		语法	DAO get(message);
		前置条件	数据检索输入条件正确
		后置条件	从数据库中获取的数据持久化对象
		不变量	无
需求接口			
DataCache.get			

5.3 进程视图

5.3.1 进程描述

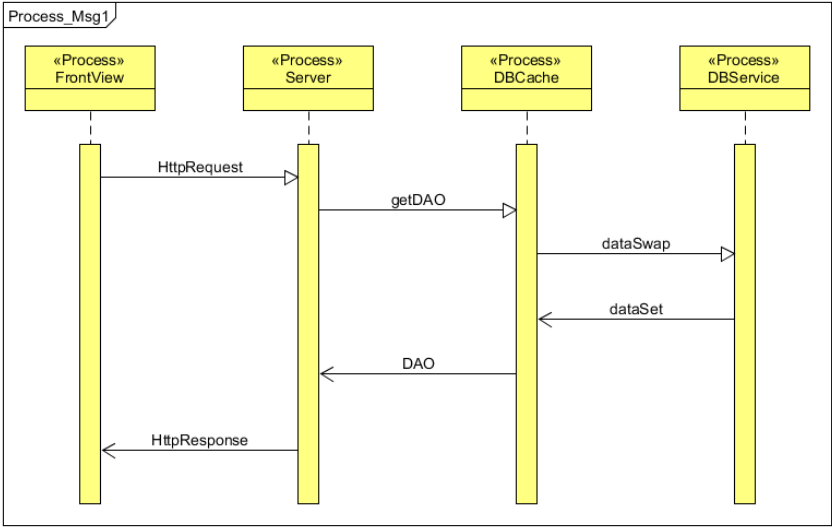


逻辑视图与进程视图的关系如下表所示：

进程	部件、连接件
FrontView	View、BrowserCompability、Bean、FrontAction、HttpRequest
Server	Service、Synonym、ResultFilter、SoftDog、BidInfo
ServerDeployemny	HttpRouter、ServerCluster、ServerDeploy、RPC
FireWalls	AttackDefense、ShieldIPList、EventRouter
WebServiceManager	CommentFilter、VocabularyDetection、ZombieDetection、 ProcedureCall、EventRouter
ServerManager	ServerErrorReport、EventRouter
DBCACHE	VirtualCacheCluster、RPC

DBService	Pipe、 Dao、 DataUpdate
DBManager	DataBaseMange、 PC
Crawler	Crawler、 DataStandarize、 EventRouter

5.3.2 进程通信描述



ID		FrontView	
发送接口			
ID	reque st	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			
FrontView.get			

ID	Server
发送接口	

ID	getD	功能描述	服务器请求获得数据持久化对象
	AO	通信方式	远程过程调用
		接口定义	get(Message);
接收接口			
FrontView.Request			

ID		DBCACHE	
发送接口			
ID	swap	功能描述	与数据库进行数据缓存数据交换
		通信方式	过程调用
		接口定义	Swap(DATA);
接收接口			
Server.getDAO			

ID		DBService	
发送接口			
ID	updat	功能描述	更新数据
	e	通信方式	过程调用
		接口定义	Update(data);
接收接口			
DBCACHE.swap			

其他进程接口描述：

ID		Crawler	
发送接口			
ID	Send	功能描述	传递爬取数据
		通信方式	远程过程调用
		接口定义	Send(data);
接收接口			
无			

ID		DBManager	
发送接口			
ID	handl	功能描述	管理数据库状态
	e	通信方式	过程调用
		接口定义	Handle(message);
接收接口			
DBService.errorReport			

ID		ServerManager	
发送接口			
ID	handl e	功能描述	管理服务器状态
		通信方式	隐式调用
		接口定义	handle(data);
接收接口			

ServerDeployment.errorReport

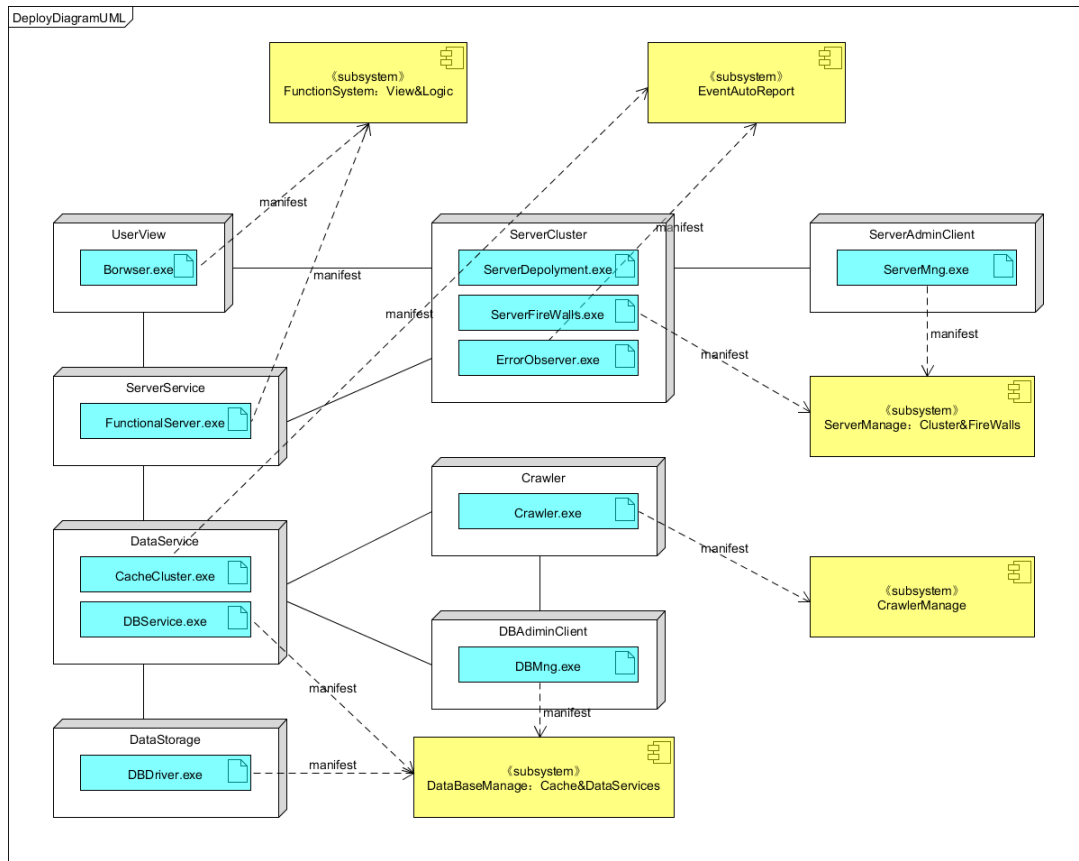
ID		ServerDeployment	
发送接口			
ID	webp	功能描述	配置服务器集群
	ack	通信方式	远程调用
		接口定义	webpack(data);
接收接口			
Server.register			
FrontView.inquiry			

ID		FireWalls	
发送接口			
ID	detect	功能描述	入侵检测与防御
		通信方式	隐式调用
		接口定义	detect(data);
接收接口			
ServerDeployment.request			

ID		WebServiceManager	
发送接口			
ID	handl	功能描述	管理服务相关的事件
	e	通信方式	隐式调用

		接口定义	handle(data);
接收接口			
Server.service			

5.4 部署视图



上图表示了本系统的基础设施以及他们与开发视图的子系统组织的映射关系，共需要 8 种类型的物理节点：

1. UserView，客户端节点，HTTP 请求的发出者
2. ServerService，服务器服务节点，接收 HTTP 请求并提供服务
3. DataService，数据库服务节点，提供数据库存取服务
4. ServerCluster，服务器集群管理节点，用于处理负载均衡和服务器集群的管理
5. ServerAdminClient，服务器管理员用户接口节点，为服务器管理员提供用户接口
6. DBAdminClient，数据库管理员用户接口节点，为数据库管理员提供用户接口
7. DataStorage，数据库存储节点，代表数据库实体
8. Crawler，自动爬虫节点，负责进行网络数据信息的爬取与传输