


规则引擎基础知识

作者: [bugless](#)

原文链接: <https://ld246.com/article/1627548055112>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

 <https://b3lofile.com/file/2021/07/solo-fetchupload-4865804408570797737-0e283874.jpeg?imageView2/1/interlace/1/format/jpg>

规则引擎基础知识

创建时间: 2021-07-29 15:07 星期四

一、什么是规则引擎

全称为 **业务规则管理系统**，英文名为 BRMS(即 Business Rule Management System)。规则引擎由推理引擎发展而来，是一种嵌入在应用程序中的组件，实现了将业务决策从应用程序代码中分离出来，并使用预定义的语义模块(注: <https://ld246.com/forward?got=https%3A%2F%2Fbaidu.com%2Fitem%2F%25E9%25A2%2586%25E5%259F%259F%257%2589%25B9%25E5%25AE%259A%25E8%25AF%25AD%25E8%25A8%2580%2F2826893> 领域特定语言)编写业务决策(业务规则)。由用户或开者在需要时进行配置、管理。需要注意的是规则引擎并不是一个具体的技术框架，而是指的一类系，即业务规则管理系统。

规则引擎实现了将业务决策从应用程序代码中分离出来，接收数据输入，解释业务规则，并根据业务规则做出业务决策。规则引擎其实就是一个输入输出平台，通过接受动态数据流入根据内部的规则出决策结果的处理器，以抽离业务逻辑保证其独立维护和动态更新。

二、为什么需要规则引擎

2.1 从开发人员视角来看

在没有规则引擎的时代，有些逻辑比较复杂的业务，只有不断的增添 if-else 去满足我们这个复的业务场景，对于开发者来说还好，对于后面接手的同学一看到到处都是 if-else，体验过的同学就会知，当然 if-else 可以通过一些模式去优化，比如使用策略模式，或者使用一些注解进行扩展点优化，样的确可以解决一部分代码不清晰的问题，但是依然无法解决开发缓慢，需要上线等问题。举个例子在风控系统中，因为风控的逻辑在不断的发生一个改变，如果我们在代码中去写死，那么发生一个改就改一下代码，上一下线，这明显是我们不能接受的。所以我们需要规则引擎去改变这个现状，通过效可靠的方式去做这些业务规则的改变。

核心问题域有了：大量无法避免的 if-else 充斥在我们的系统中，对于系统的维护造成了威胁。系统架构的质量属性：可扩展性、研发质量&效率 严重受损)：

1、无法直观表达现有业务逻辑，新人入手困难

2、新增&改动逻辑困难，极难扩展

3、容易出现 case 遗漏，质量堪忧

4、兜底&通用处理成本高，系统鲁棒性差

5、每次变更逻辑时都需要经历一次完整的研发-测试-发布-回测-灰度 等漫长的研发活动，效率本堪忧

所以隔离这部分无法避免的业务决策逻辑，让逻辑变得纯粹可独立维护，所有逻辑一目了然动态配，就是规则引擎要做的事儿。

2.2 从业务人员角度来看

业务人员独立配置业务规则，开发人员无需理解，让业务人员的规则和真正的实际情况一致。

增加业务的透明程度，业务人员配置了之后其他业务人员也能够知道，以前只能通过代码扣扣相。

规则高效改动和上线，一般业务人员提出需求之后都是希望能尽快上线，但是之前都需要有代码发，项目上线等环节，现在业务人员配置好了之后即配即用。

减少业务人员和开发人员的矛盾，开发人员通常会因为一些时间因素或者一些理解不到位导致业务人员的规则实现有偏差，最后业务同学会对开发同学产生一些小小的矛盾，这下完全业务配置解除开之后，只要不断的升级规则引擎，业务规则就不会再对开发人员有依赖。

使用规则引擎的优势

1、业务规则与系统代码分离，实现业务规则的集中管理

2、在不重启服务的情况下可随时对业务规则进行扩展和维护

- 3、可以动态修改业务规则，从而快速响应需求变更
- 4、规则引擎是相对独立的，只关心业务规则，使得业务分析人员也可以参与编辑、维护系统的业务规则
- 5、减少了硬编码业务规则的成本和风险
- 6、使用规则引擎提供的规则编辑工具，使复杂的业务规则实现变得简单

<h2 id="三-规则引擎执行过程示意图">三、规则引擎执行过程示意图</h2>

<p></p>

<h2 id="四-规则引擎应用场景">四、规则引擎应用场景</h2>

<p>对于一些存在比较复杂的业务规则并且业务规则会频繁变动的系统比较适合使用规则引擎，如下</p>

- 1、风险控制系统----风险贷款、风险评估
- 2、反欺诈项目----银行贷款、征信验证
- 3、决策平台系统----财务计算
- 4、促销平台系统----满减、打折、加价购

<h2 id="五-市面上常见的规则引擎">五、市面上常见的规则引擎</h2>

<blockquote>

<p>只记录常见的规则引擎会，不断更新，对相关项目的实际使用另行更新。</p>

</blockquote>

<table>

<thead>

<tr>

<th>规则引擎推荐</th>

<th>规则引擎推荐</th>

</tr>

</thead>

<tbody>

<tr>

<td>stepchain- 通用业务流程流水线处理框架</td>

<td>OpenL Tablets- 商业规则引擎</td>

</tr>

<tr>

<td>aswan- 陌陌风控系统静态规则引擎</td>

<td>Interleave- 业务流程自动化应用</td>

</tr>

<tr>

<td>RuleEngine- 轻量级规则引擎模块</td>

<td>Activiti- 业务流程管理(BPM)和工作流系统</td>

</tr>

<tr>

<td>URule- 纯 Java 规则引擎</td>

<td>Intalio BPM- 业务规则引擎</td>

</tr>

<tr>

<td>Easy Rules- Java 规则引擎</td>

<td>Prova- 事件处理规则语言</td>

</tr>

<tr>

<td>Aristotle- Ruby 商业逻辑引擎</td>

<td>JRuleEngine- java 规则引擎</td>

```

</tr>
<tr>
<td>JBoss BPM Suite- 业务流程管理平台</td>
<td>InfoSapient- 开源规则引擎</td>
</tr>
<tr>
<td>JBoss BRMS- 业务规则平台</td>
<td>JEOPS- Java 的演绎法规则引擎</td>
</tr>
<tr>
<td>JBoss OptaPlanner- Java 规划引擎</td>
<td>Mandarax- Java 规则引擎</td>
</tr>
<tr>
<td>Tohu- </td>
<td>JRules- 业务规则引擎</td>
</tr>
<tr>
<td>A2D- 规则引擎</td>
<td>JLisa- 商业规则引擎</td>
</tr>
<tr>
<td>Esper- 复杂事件处理组件</td>
<td>NxBRE- 业务规则引擎</td>
</tr>
<tr>
<td>NRuleEngine- 简单 C#规则引擎</td>
<td>OpenRules- 商业规则管理框架</td>
</tr>
<tr>
<td>Simple Rule Engine- 推理规则引擎</td>
<td>Drools Expert- 规则引擎</td>
</tr>
<tr>
<td>Hammurabi- Scala 规则引擎</td>
<td>Jess- 规则引擎</td>
</tr>
</tbody>
</table>
<p>重要推荐</p>
<table>
<thead>
<tr>
<th>序号</th>
<th>名称<br></th>
<th>简介</th>
<th>文档</th>
<th>备注</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><a href="https://ld246.com/forward?goto=https%3A%2F%2Fwww.drools.org%2F" target=

```

= "_blank" rel="nofollow ugc">Drools </td>
<td>drools 是一款由 JBoss 组织提供的基于 Java 语言开发的开源规则引擎，可以将复杂多变的业务规则从硬编码中解放出来，以规则脚本的形式存放在文件或特定的存储介质中(例如存放在数据库中)，使得业务规则的变更不需要修改项目代码、不用重启服务器就可以在线上环境立即生效。

使用最为活跃、广泛并且开源。
</td>
<td>Drools 中文网 | 基于 java 的功能强大的开源规则引擎

Drools - Business Rules Management System (Java™, Open Source)
</td>
<td>推荐书籍《Drools 规则引擎技术指南》</td>
</tr>
<tr>
<td>2</td>
<td>Easy Rules</td>
<td>EasyRule 是轻量级的规则引擎 API。它提供 Rule 抽象来创建带有条件和动作的规则，以及 RulesEngine 通过一组规则运行以测试条件和执行动作的 API。

通过注解、链式编程、表达式 yml 文件 等方式灵活选择使用
</td>
<td>The simple, stupid rules engine for Java
</td>
</tr>
<tr>
<td>3</td>
<td>Bstek 锐道：URULE PRO 规则引擎 </td>
<td>URule Pro 是一款由上海锐道信息技术有限公司自主研发的纯 Java 规则引擎，提供规则集、决策表、交叉决策表（决策矩阵）、决策树、评分卡、复杂评分卡、规则流等八种类型的业务规则设计工，采用业内首创的纯浏览器编辑模式，打开浏览器即可开始复杂规则的设计与测试</td>
<td>Urules 文档
</td>
<td>测试账号：uruletest10086
测试密码：dakou2010
</td>
</tr>
<tr>
<td>4</td>
<td>Aviator</td>
<td>Aviator 是一个高性能、轻量级的 java 语言实现的表达式求值引擎，主要用于各种表达式的动求值。</td>
<td>AviatorScript 文档 </td>
</tr>
<tr>
<td>5</td>
<td>VisualRules</td>
<td>旗正 VisualRules 是由国家科技部和财政部的创新基金支持，专门针对国内规则引擎市场空白情况，结合国内项目的特点而开发的一款业务规则管理系统(BRMS)产品。</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>iLog</td>

<td>BM WebSphere ILOG JRules 是目前业界领先的业务规则管理平台。与传统的由 IT 人员用硬
码来维护规则的做法不同，ILOG JRules 让业务用户能够在不依赖或者有限依赖于 IT 人员的情况下
快速创建、修改、测试和部署业务规则，以满足经常变化的业务需求。ILOG JRules 提供了一整套的
具，帮助开发人员和业务人员进行规则的全生命周期管理。 </td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>OpenL Tablets</td>
<td>OpenL Tablets 业务规则引擎（BRE）和业务规则管理系统（BRMS），属于商业规则引擎。 </
d>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>QLExpress</td>
<td>由阿里的电商业务规则、表达式（布尔组合）、特殊数学公式计算（高精度）、语法分析、脚
二次定制等强需求而设计的一门动态脚本引擎解析工具。在阿里集团有很强影响力，同时为了自身
断优化、发扬开源贡献精神，于 2012 年开源。
</td>
<td><a href="https://ld246.com/forward?goto=https%3A%2F%2Fgitee.com%2Fcuibo119%2
QLExpress" target="_blank" rel="nofollow ugc">QLExpress: 规则引擎 (gitee.com)</td>
<td></td>
</tr>
</tbody>
</table>
<h2 id="6-大厂实践">6、大厂实践</h2>
<p><a href="https://ld246.com/forward?goto=https%3A%2F%2Ftech.meituan.com%2F201
%2F06%2F09%2Fmaze-framework.html" target="_blank" rel="nofollow ugc">从 0 到 1: 构建
大且易用的规则引擎 - 美团技术团队 (meituan.com)</p>
<p><a href="https://ld246.com/forward?goto=https%3A%2F%2Fzhuannlan.zhihu.com%2Fp%
F371831214" target="_blank" rel="nofollow ugc">谈一谈规则引擎在活动系统中的落地 - 知乎 (zh
hu.com)</p>
<p><a href="https://ld246.com/forward?goto=https%3A%2F%2Fzhuannlan.zhihu.com%2Fp%
F47472836" target="_blank" rel="nofollow ugc">网易考拉规则引擎平台架构设计与实践 - 知乎 (z
ihu.com)</p>
<p><a href="https://ld246.com/forward?goto=https%3A%2F%2Fzhuannlan.zhihu.com%2Fp%
F140916822" target="_blank" rel="nofollow ugc">复杂风控场景下，如何打造一款高效的规则引
- 知乎 (zhihu.com)</p>
<p><a href="https://ld246.com/forward?goto=https%3A%2F%2Fzhuannlan.zhihu.com%2Fp%
F364546754" target="_blank" rel="nofollow ugc">B 站新一代 golang 规则引擎的设计与实现 -
乎 (zhihu.com)</p>
<blockquote>
<p>标 题：规则引擎基础知识

作 者：<a href="https://ld246.com/forward?goto=http%3A%2F%2Fbugless.site
2F" target="_blank" rel="nofollow ugc">bugless

出 处：<a href="https://ld246.com/forward?goto=http%3A%2F%2Fbugless.site" target="

blank" rel="nofollow ugc">http://bugless.site

创建时间：2021-07-29 16:39:08

关于博主：坐标广州，喜欢折腾各种东西的工程师，如有问题探讨可以直接下方留言。

声援博主：如果您觉得文章对您有帮助，可以评论、订阅、收藏。您的鼓励是博主的最大动力！</p>
</blockquote>