



# VRS 产品特点

杭州旗正信息技术有限公司

---

地址：杭州市文三路 199 号 7 号楼 512 室

电话：(860571) 56771736

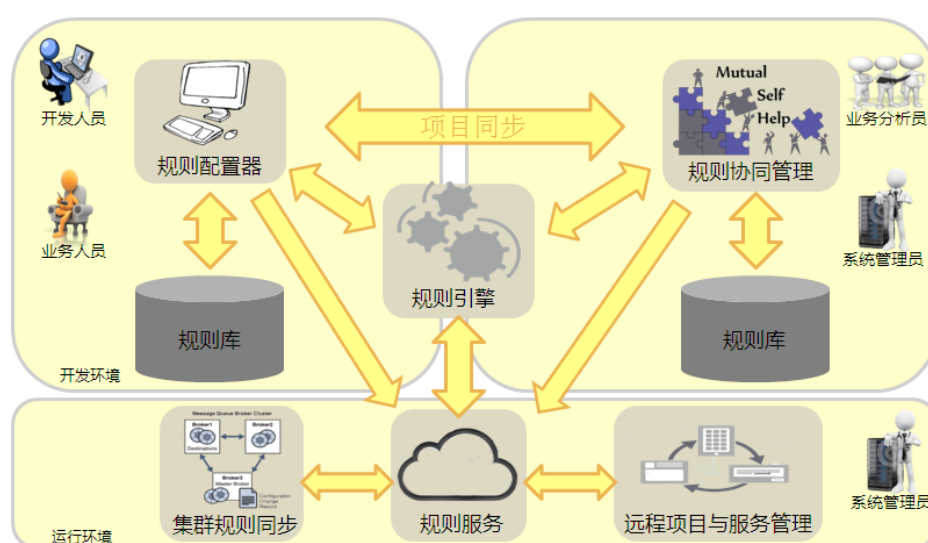
传真：(860571) 81110230

网址：[www.flagleader.com](http://www.flagleader.com)

E-mail：[sale@flagleader.com](mailto:sale@flagleader.com)

## 完整规则管理系统

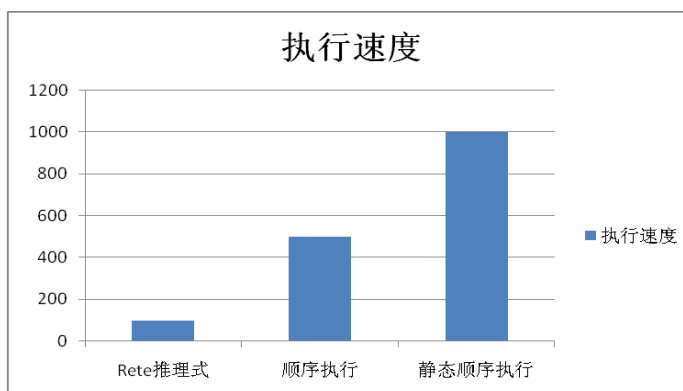
VRS 以规则库为基础，规则编辑器为规则录入窗口，规则引擎为动力，辅以规则协同管理、远程项目与服务管理、集群规则同步等功能模块。帮助企业打造一套完整的商业规则定制与管理平台。



## 速度最快规则引擎

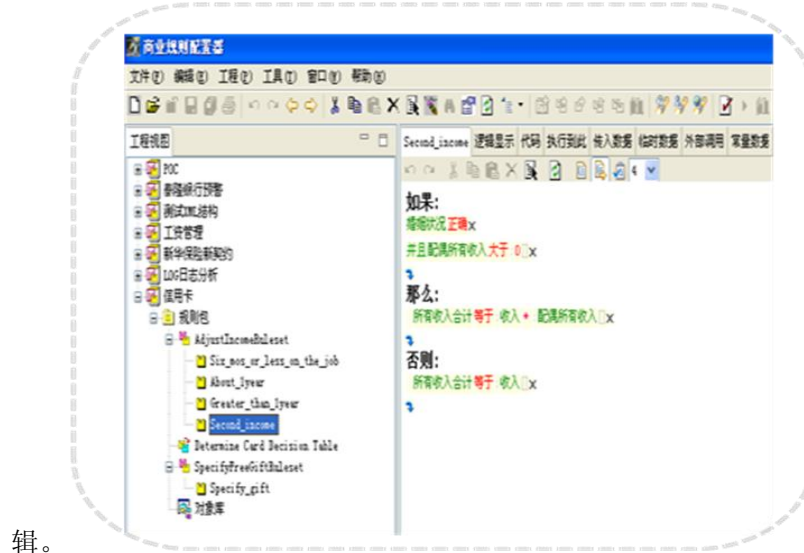
规则引擎产品有三种执行方式：基于 Rete 算法推理式、顺序执行、静态顺序执行。基于 Rete 算法推理式是绝大多数产品采用的一种方式。此种方式基于 Rete 算法进行模式匹配，在规则运行过程中耗费大量资源进行推理式解析。规则执行速度和效率低下。

VRS 采用静态顺序执行方式，将规则完全编译成可执行的代码进行执行。这样就最大限度的保证了规则执行的速度。同时由于所有的业务规则全部对应的生成了可执行代码，这样使得用于加载和调用规则的服务程序非常简单。只需要实现注册、查找、调用等工作，极其精简的规则引擎除了可以保证单次调用规则的性能之外，还使得系统不再受限于同时在线数，规则引擎不会因为同时调用的负载高，而影响性能。



## 独立编辑器，全中文配置

VRS 采用独立编辑器，全中文配置的方式进行业务逻辑的编写。规则配置界面使用表格来定义业务数据对象（BOM），这些业务数据对象包括单个数据，表格数据，Excel 数据，数据库表数据，XML 数据，SOAP 数据等。同时提供类自然语言来描述操作这些数据逻辑。这些类自然语言和平时业务人员使用的业务语言及其类似，使得业务人员可以直接配置这些逻辑。



## 数据库对象动态 ORMMapping 技术

传统规则引擎产品采用一般的 ORMMapping 技术(如 Hibernate)，将数据库的表等映射成 java 类等。这种方式可以对象化的方式来操作数据库，但是这种方式却使得当变动数据库结构时，必须改动类，这样操作数据库的逻辑不能像其他的业务逻辑一样可以动态变化。

VRS 采用动态的 ORMMapping 技术，数据库的表映射到一个统一的接口类。这样操作数据库的逻辑就和其他的业务逻辑没有区别，同时数据结构的变化也可以像业务逻辑一样可以动态变化。

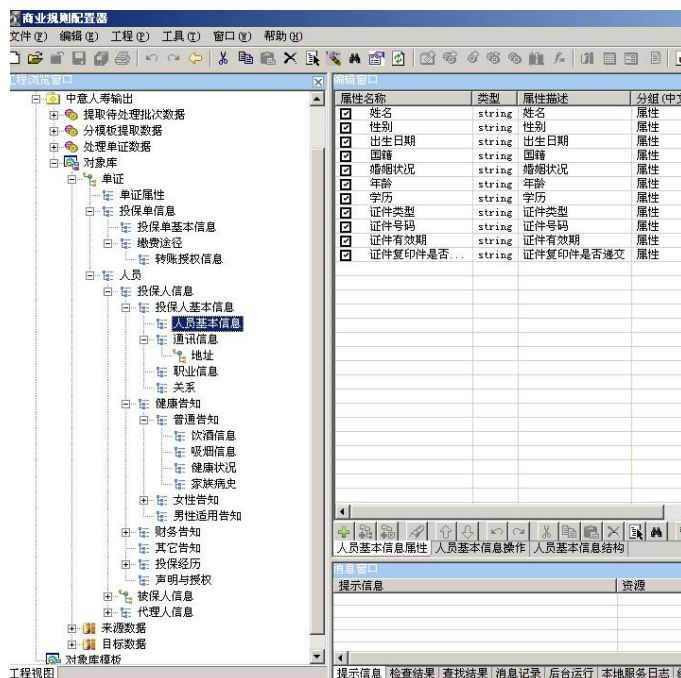
动态 ORMMapping 技术，解放了数据库表结构变化导致需要修改程序并重启服务器的问题，使得数据库表结构的变更就像规则引擎应用业务逻辑的变更一样，实现了完全的动态。



## 数据对象动态可变

VRS 采用 map 的方式来传递规则包的数据对象，这样就保证了对象结构的可变性。一般的规则引擎采用类对象的方式来传递需要处理的数据，这样如果改变对象的类型或者结构，就需要改动变量传递类的结构。采用 map 方式规避了这方面的问题，使得数据对象的结构也是可变的。

这是为了解决当业务需求变更时，除了只是调整逻辑之外，还相应需要变更数据结构这类需求。比如某个规则服务，提供给外部的系统调用，采用可变参数对象接口，就可以使得后期可以根据需要，变更数据结构。



## 安全考虑，版本控制

VRS 配置器由于大量开发人员与业务人员同时参与使用，处于安全考虑，所有使用人员必须经过用户名和密码的身份认证。这种身份认证可以和业务系统结合，并且通过规则的权限分配，用户只能查看和修改自己具有权限的规则，从而最大限度的保证规则修改的安全性。

同时集中统一的管理也为规则包的整体版本控制提供了可能。很多实际业务要求，同一规则服务，会在不同的时间段或者不同的条件下，会调用不同的版本（也就是具体其中某几个规则会有所不同），而这几个版本可能是同时都要有效的。因此需要针对规则包进行统一的版本控制。



## 编辑、修改、测试自行完成

VRS 可以在编辑环境下，直接对规则包进行整体测试。测试时只需输入所需的参数的值，就可以查看到输出结果，以及可以看到整个规则执行路径以及对应的数据变化情况。这样就可以非常方便的规则包进行测试检查，或者对下面的某个规则或者规则集单独进行检查。

当规则包被调用时，有时候用户需要知道这次调用到底激活了那些规则，他们的先后执行顺序，以及调用此规则时，对业务对象的修改情况（进规则之前，业务对象什么状态，经过规则执行之后，业务对象变成了什么状态）。VisualRules 可以将所有当次运行的这些规则执行轨迹记录下来，供用户进行查阅，或者存储在数据库中，供以后查阅。这在用户进行规则的查错时非常有用，可以马上定位到底是运行到那个规则时，发生了错误。

## 标准接口，轻松集成

VRS 提供完整的标准化接口，集成工作弹指之间：

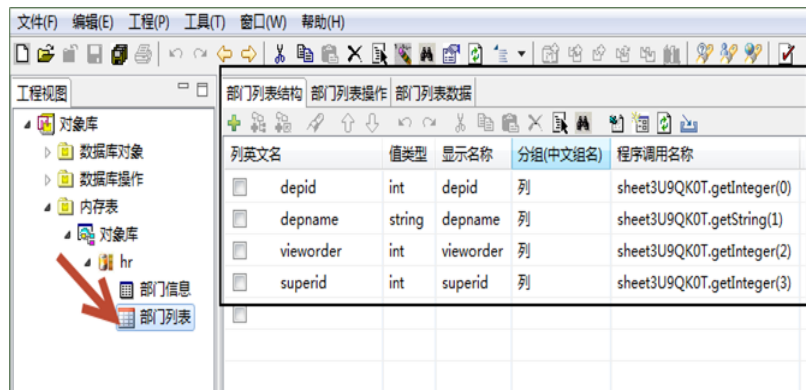
- **Socket 接口：**socket 通讯方式接口，传输性能好、安全性高；
- **Servlet 接口：**通过 `HttpRequest` 提交等方式来访问 VRS 中的规则包，返回的数据结构以及 xml 结构方式来存取数据，实现以 Json 结构来返回数据；
- **Soap 接口：**外部程序只需要引用 `engine.dll` 以及配置规则服务器的 ip 地址，就可以直接调用 VRS 中的规则包；
- **本地 Java 接口：**外部 java 程序只需引用 `engine.jar` 包，就可以访问。



## 独有内存表设计

伴随着各行各业 IT 建设的不段深入，“大数据量”这个我们曾经认为很遥远的事物，现如今时刻出现在我们日常业务中。这些业务数据处理的过程实际上会产生大量的中间数据，数据量大且逻辑复杂不透明。传统的处理方式无外乎在数据库中建大量临时表或编写高效的批处理代码，这两种方式都存在对人员要求高，系统影响大的弊端。

VRS 独有的计算表格(内存表)功能，在内存中模拟数据库表，以直观高效的方式处理和存储中间数据。让业务人员也不再惧怕大数据量，坦然面对。



## 多种样式决策表

VRS 针对国内决策表的特殊情况，设计了多种决策表，包括表格类决策表、多维决策表、关联决策表。

当然也可以根据业务系统的实际需要进行扩展。这些不同类型的决策表，使得业务人员可以更加简单方便的表述业务逻辑。

交叉决策表									
如果	那么								
车型	5万	10万	15万	20万	30万	50万	100万		
6座以下	1307	1974	2294	2510	2913	3691	4655		
6-10座	1312	1981	2302	2519	2922	3704	4672		
10-20座	1308	2128	2496	2735	3190	4064	5345		
20-36座	1966	2946	3474	3857	4538	5631	7671		
36座以上	2995	4826	5418	5973	6862	8912	11723		
6-10座	1270	1916	2227	2437	2827	3583	4714		
10-20座	1414	2134	2480	2714	3149	3990	5246		
20-36座	1961	3015	3525	3880	4528	5772	7593		
36座以上	2707	4276	5043	5598	6587	8464	11133		
6-10座	1243	1875	2180	2395	2767	3507	4612		
10-20座	1364	2089	2429	2657	3082	3906	5139		
20-36座	2036	3074	3573	3910	4535	5748	7561		
36座以上	3055	4811	5599	6064	6903	8922	11340		
2吨以下	1219	1901	2237	2463	2900	3635	4747		
2-5吨	1962	3060	3599	3963	4666	5850	7640		
5-10吨	2251	3511	4131	4549	5366	6714	8769		
20-36座	3006	4813	5662	6235	7341	9202	12018		
10吨以上	1036	1615	1900	2093	2463	3088	4034		
特种车型一	2634	4539	5387	5963	7104	8980	11728		
特种车型二	1363	1705	1983	2193	2508	3402	5131		
特种车型三	624	817	927	1031	1253	1647	2413		

### 多维决策表

关联决策表									
如果	那么								
区域类别	门槛基数	促销奖励系数							
一类	大于等于350000	100%							
	小于350000并且门槛基数大于等于175000	目标达成率×100%							
	小于175000	0%							
二类	大于等于250000	0							
	小于250000并且门槛基数大于等于125000	目标达成率×100%							
	小于125000	0							
三类	大于等于150000	0							
	小于150000并且门槛基数大于等于75000	0							
	小于75000	0							
四类	大于等于100000	0							
	小于100000并且门槛基数大于等于50000	0							
	小于50000	0							

如果	那么
税前工资	所得税
小于等于2000	0
大于2000并且税前工资小于等于2500	(税前工资-2000)×5%
大于2500并且税前工资小于等于4500	(税前工资-2000)×10%-25
大于4500并且税前工资小于等于6500	(税前工资-2000)×15%-125
大于6500	(税前工资-2000)×20%-425