



1

提纲

- 消息路由的使用场景
- 消息路由的种类
- 配置、测试、跟踪
- Use Case

2



3

什么时候使用消息路由器

- 业务场景
 - 消息转发
 - 消息路由
 - 消息分发
 - 业务编排
 - 报表：集成平台，抓各个系统的数据攒成一个报告（院长报告）
 - 互联互通中抓其他数据，不是典型场景
 - 托卡支付？
- 数据转换，应用于上面所有场景。（包括过滤、校验、转换、汇总。。。）

4

什么是消息路由器（消息路由引擎）

Message Router是HealthConnect系统内置的可直接使用的产品组件。它是一个BusinessProcess类，客户不需要了解或者调整Router的内部机制，只需要通过写Message Rule（消息规则）来配置Ensemble消息的发送和路由。

5

什么是消息路由规则

- 用图形化工具创建
- 供路由引擎使用
- 通过对消息源、消息类型、消息内容进或多个BP、BO
- 可以认为是一个XML文档
- Business Rule继承Ens.Rule.RuleDefinition.
- 修改规则不需要重启业务组件



6

其他的业务规则

除了消息业务规则，系统还提供了一遍业务规则，告警业务规则，他们和消息路由没有关系，而且使用也不像消息规则这么常用。我们会在介绍DTL部分介绍。

7

消息路由的种类

HealthConnect内置了4种不同的路由器(路由引擎)，用于路由不同类型的消息

- 普通消息路由器(General Message Router)
 - Enslib.MessageRouter.RoutingEngine
- 虚拟文档消息路由器(VDoc Message Router)
 - Enslib.MessageRouter.VDocRoutingEngine
- HL7消息路由器(HL7 Message Router)
 - Enslib.HL7.MessageRouter.RoutingEngine
- 警告消息路由器(Alert Message Router)

8

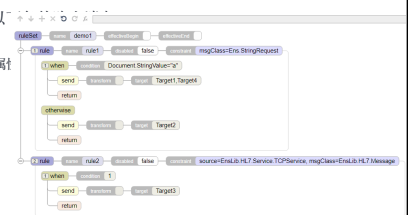
普通消息路由(Message Router)

9

普通消息路由器

使用消息路由规则，按照以下

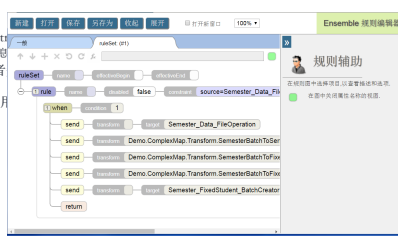
- 源业务组件名字
- 消息类型 (Messages class)
- 消息中的值，通常是消息中属性



10

创建消息路由规则的要点

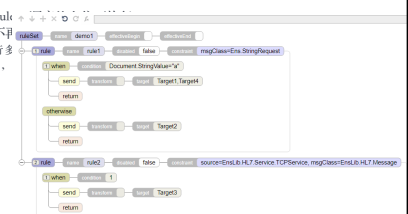
- 通常的设计包含：
 - 多个rule，每一个Rule通过Constraint
 - 普通路由规则Constraint可用消息
 - 每个包含一个when，一个或者
 - Send中可以使用数据转换。
 - Rule可以设置“delete”动作调用



11

Rule的执行原则

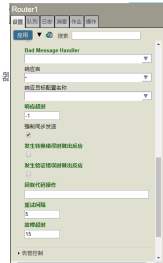
- 一个RuleSet里面可以有多个Rule
- 碰到return，退出整个ruleset，不执行后面的Rule
- 如果希望在一个ruleset里面执行多个Rule，需要在每个Rule里面设置“continue”动作
- 如果一个Rule里面有多条when，判断。



12

路由器的配置

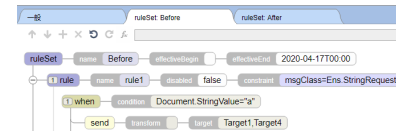
- 强制同步发送
 - 路由器默认发送异步消息。如果需要收到返回消息，需要选中该选项。
- 响应自
 - 从下拉列表选中其中的组件，这些组件返回的消息将被路由器返回给给路由表(BS)。*表示选中
- 响应目标配置名称
 - 所有返回router的消息将转发给该表中的目的地组件。



13

RuleSet和Effective Dates

- 一个业务规则可以包含多个RuleSet，在同时只能有一个生效。
- 添加第2个RuleSet在“一般”页面
- 创建两个RuleSet，配置相应的生效时段，可以实现路由业务在一个时间点的自动切换。



14

其他的细节

- Rule
 - 配置Rule名字便于跟踪管理

Rule可以设置"delegate"动作调用另一个Rule

15

HL7消息路由(HL7 Message Router)

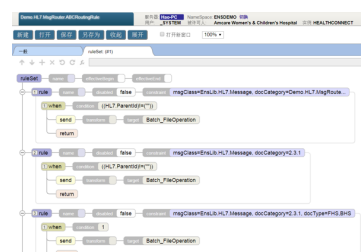
16

提纲

- 业务规则的基本概念
- 规则集 (RuleSet)
- 规则执行的逻辑和活动
 - When, return, assign, send...
- 普通业务规则(General Business Rule)
- 路由业务规则 (Message Rule)
- HL7消息路由规则
- 其他
 - Rule Log
 - UtilityFunction

17

HL7例子



18

HL7消息路由

EnsLib.HL7.MsgRouter.RoutingEngine

- 为HL7消息提供的路由引擎
- 提供可选的消息校验功能
- 根据规则发送HL7消息
 - 可以发送多个目的地
 - 可以收到并返回响应消息到调用方
 - 可以收到响应消息转发其他目标
 - 可以同时做上面两个操作

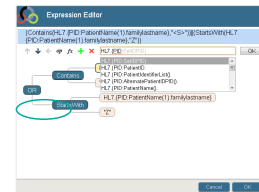


19

在Condition中使用自动提示

编辑器提供自动提示

- 必须首先输入“HL7.”



20

HL7 Message Rule

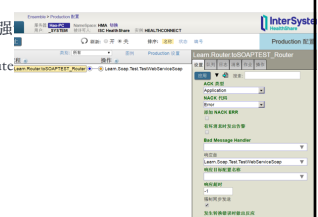
- 约束 (Constraints) 定义了大众化的限制条件:
 - Source, Schema 和 Message Type. (HL7)
- HL7虚拟文档的寻址方法



21

演示- HL7消息路由

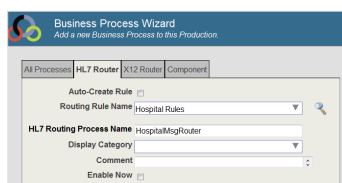
- 直接添加到production
- 强制同步发送, 至少HL7消息, 如果不强在产品里面测试是发送的。
- 响应自给出从哪里来的响应会被送回route



22

通过Wizard创建HL7 Router

可使用创建向导选择HL7 Router

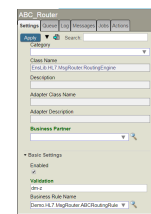


23

HL7消息验证

- Validation 为空不校验
- 典型设定值为“dm-z”
 - d: 校验文档类型DocType
 - m: 校验消息结构: 仅仅segment级别验证不包含field级别验证
 - z: 校验Z-segments

更多设置参考文档



24

25

Class `EnsLib.MsgRouter.VDocRoutingEngine` [Extends EnsLib.MsgRouter.RoutingEngine](#)

一般情况可以直接使用上面的类，

特殊情况需要自己创建类继承以上类

`validation = 1 (default=0)`

26

27

28

- Ensemble ☐ View ☐ Business Rule Log.
 - 每次尝试匹配规则时都会记录.
 - Reason 为空表示没有匹配到规则.

匹配的具体规则, 为空则表示没有匹配的
规则。

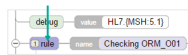
The screenshot shows the AWS IAM console 'Rules' tab. The table lists rules with columns: Name, Type, Action, Condition, and Status. A red box highlights the 'Rule name' column header. To the right, a 'Rule details' panel shows the 'Execution ID' and 'Time Executed' for a specific rule.

发给哪里

29

包含条件和调试条目。

如果RuleLogging包含d，则将记录规则中的调试操作。



Rule Log		
Visual Trace	Rule Definition	
Execution ID:	19	
Session ID:	53	
Time Executed:	10/14/2010 11:30:53.417	
Rule Name:	Hospital/HospitalRoutingRule	
Rule Set:	(none)	
Reason:	rule1 altered?	
Error#:	No	
Error Message:	(none)	
Return Value:	sent To Pharmacy Hospital ER	
Activity Name:	Send To H.E.M. High/Lowest Routing	
Effective Begin Date/Time:	10/1/2005 0:00:00	
Effective End Date/Time:	10/1/2025 0:00:00	
Detailed Logging		
Log Level	Source	Message
Condition		Executing ruleSet1: effectiveBegin = (not specified)
Debug		H.D.MHS1.D1 = "PHARMACY"
Condition	rule1	Constraints for rule "rule1" passed
Condition	rule1.altered?	Has H.D.MHS1.Processing? a = 1, D.MHS1.S = "PHARMACY"

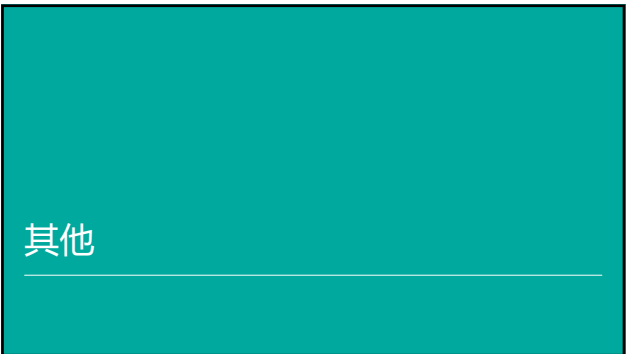
30

规则日志设置(2015.1+)

在Business Process中配置.

设置	错误(Errors)	返回值(Return Values)	Debug动作(Debug Actions)	规则中的条件(Conditions in Rule)
Empty	✓			
e	✓			
r (default)	✓	✓		
d	✓	✓	✓	
c	✓	✓		✓
a	✓	✓	✓	✓

31



32

消息规则的扩展

- 可能需要铺盖的引擎方法:
- (以Vdoc消息路由为例)
 - OnValidate()
 - OnRequest()
 - OnResponse
 - OnInit()
 - OnTimeOut()
 - ...

参考范例:

```
Method OnValidate(pDoc As EnsLib.HL7.Message, pValSpec As %String, Output pStatus As %Boolean) {
    Set valid = ##super(pDoc, pValSpec, .res)
    set pStatus = res
    // Perform custom steps here
    quit 0
}
```

33



35

Business Rule类型

- 普通业务规则(Generic Business Rule)
- 路由业务规则(Message Routing Rule)
 - HL7路由业务规则
 - Vdoc路由业务规则
- 托管告警业务规则

37

规则命名规范

对Rule Name 属性, 可以:

- 引用全名.
 - 例如, Demo.TRU.Process.OrderProduct.CalculateShipping.
 - 可以引用命名空间内的任何已定义的规则.
- 不含包名的类名.
 - 例如, CalculateShipping.
 - 调用时将自动扩展其类名, 包含当前Business Process的类路径.

Rule Name

CalculateShipping

Name of Business Rule to invoke

Rule Editor

Rule Name

Demo.TRU.Process.OrderProduct.CalculateSh

Name of Business Rule to invoke

Rule Editor

40

应用函数(Utility Functions)

内置的functions，可以在Business Rule的conditions段落中使用。

- 如: Contains(), DayOfWeek(), Max(), StartsWith().
- 能添加用户定义的函数。



44

创建应用函数U

继承Ens.Rule.FunctionSet.

创建class method，返回特定的值.

参见文档: Developing Ensemble Productions □ Less Common Tasks □ Defining Custom Utility Functions.

```

Class Utility.MyFunctions Extends Ens.Rule.FunctionSet
{
    //Accepts time in 24-hour format (e.g. 13:15). Returns the number of seconds
    //since midnight for that time
    ClassMethod GetSeconds(time As %String) As %Numeric
    {
        quit $ZTIMEH(time)
    }
}

```

45