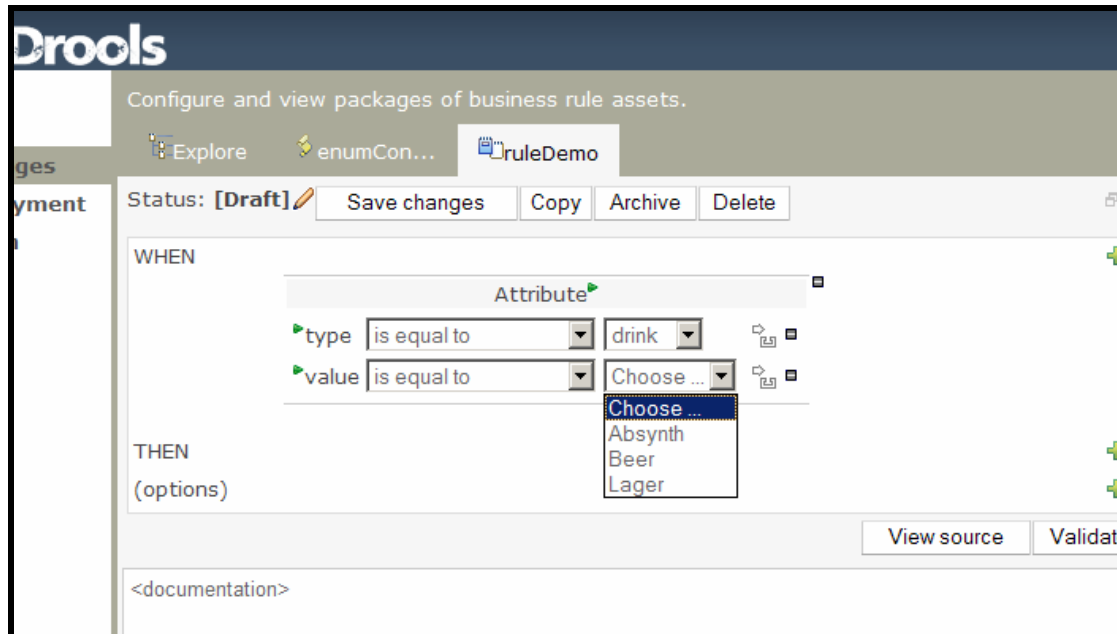


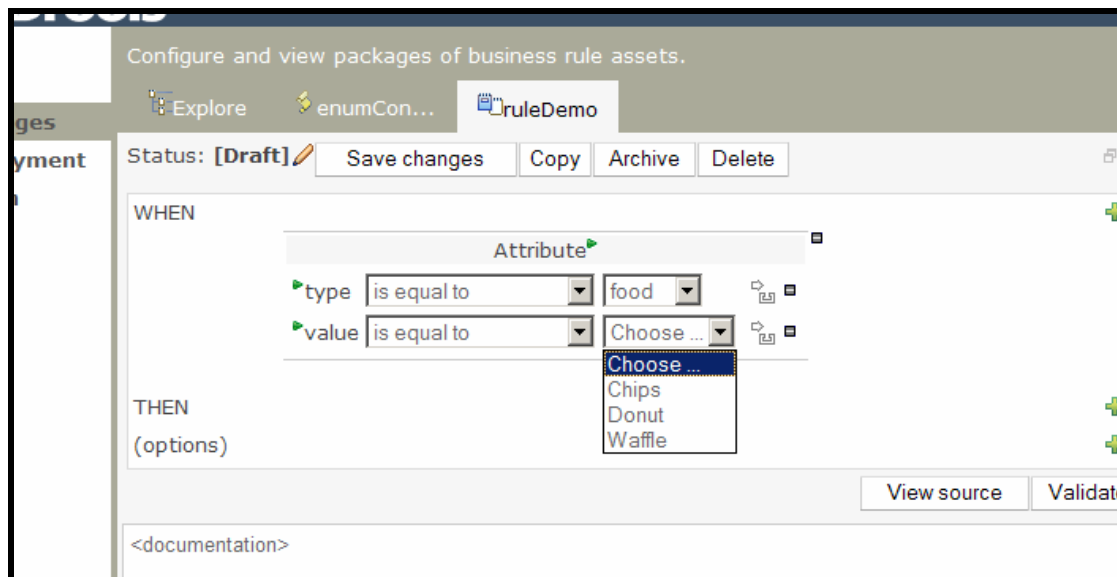
Conditional Enumerations in JBRMS v4.0.2

This facility allows you to populate the drop down list of a second fact model field based on the selected value of the first fact model field.

In this illustration, there is a drop down list for Attribute.type. Having chosen, **type** "drink" – the **value** drop down is populated with a selection of healthy drinks :



In this illustration, having chosen, **type** "food" – the **value** drop down is populated with a selection of healthy meals:



This facility expands on the already powerful option of Enumerations described here

<http://wiki.jboss.org/wiki/Wiki.jsp?page=JBRMSV4.01150CreateEnumerationsViaDatabaseAccess>

The advantage of this facility is that instead of a flat fact model:

```
FactFlat
. drink
. food
. hobby
. etc
```

A simple fact model can be used :

```
Attribute
. type
. value
```

This means if a new field, say music is required:

- A new fact model **does not have to be built** and loaded into JBRMS
FactFlat
 . drink
 . food
 . hobby
 . music
 . etc
- The execution code **does not have to be changed** to use the new fact model

Instead, the same simple fact model Attribute can be used with type="music", value="Techno".

For the flat fact model, a **simple enumeration** is created as a Technical Asset for **each field** of the fact :

```
'FactFlat.drink' : ['A=Absynth', 'B=Beer', 'L=Lager']
```

```
'FactFlat.food' : ['C=Chips', 'D=Donut', 'W=Waffle']
```

For the simple fact model, 2 enumerations are required:

1. Technical Asset for type – simple enumeration:

'Attribute.type':(new com.mycom.ListLoader).demoType()

Typical code for **type** drop down list :

```
public ArrayList demoType(){  
  
    ArrayList<String> aL=new ArrayList<String>();  
  
    aL.add("dk=drink");  
    aL.add("fd=food");  
    aL.add("il=illness");  
    return aL;  
}
```

2. Technical Asset for value – conditional enumeration :

For conditional enumerations, a single method call returns a **Map** containing enumerations for all values :

Key <String>	Value <List>
"Attribute.value[type=dk]"	['A=Absynth', 'B=Beer', 'L=Lager']
"Attribute.value[type=fd]"	['C=Chips', 'D=Donut', 'W=Waffle']
"Attribute.value[type=il]"	['L=Liver Failure', 'O=Obese', 'D=Death']

So the Technical Asset is :

=(new com.mycom.ListLoader).demoValue()

Typical code for **type** drop down list :

```
public Map<String,List> demoValue(){  
  
    Map<String,List> mapTypes=new HashMap<String,List>();  
  
    List<String> lDrink=new ArrayList<String>();  
    lDrink.add("A=Absynth");  
    lDrink.add("B=Beer");  
    lDrink.add("L=Lager");  
    mapTypes.put("Attribute.value[type=dk]",lDrink);  
    List<String> lFood=new ArrayList<String>();  
    lFood.add("C=Chips");  
    lFood.add("D=Donut");  
    lFood.add("W=Waffle");  
    mapTypes.put("Attribute.value[type=fd]",lFood);  
    return mapTypes;  
}
```