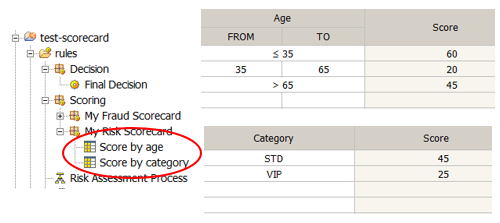
# Content

* **scorecard-bom**: a project with a BOM to implement scorecards
* **test-scorecard**: a sample ruleproject that uses scorecard-bom to implement some simple scorecard tables
* **test-xom**: java classes used to map the BOM of test-scorecard.

# Intructions

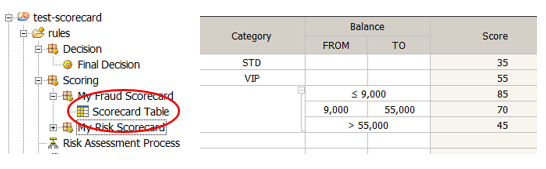
1. To install, you just need to import ‘scorecard-bom’ rule project in your workspace and reference it from other rule projects where you want to implement scorecards.
2. Each scorecard consists in a set of decision tables in the same rule package. For instance:



This will compute a score named “My\_Risk\_Scorecard”. You can get access to that score using this verbalization:

***if*** the score of **"My\_Risk\_Scorecard"** is more than 120 ...

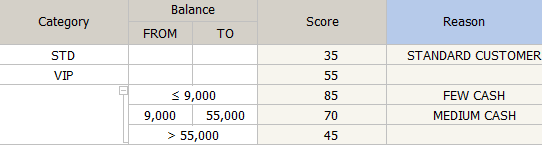
1. You can also implement a scorecard in a single table like this:



This will compute a score named “My\_Fraud\_Scorecard”.

NOTE: Note that each row defines a segment for just one attribute. This is the normal semantic of a scorecard. However, in fact, these are standard DT and you can do anything you want, but you will break the scorecard paradigm.

1. You can also add reasons to the score. For instance:

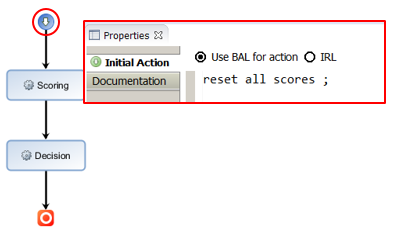


You can later access the reasons of a specific scorecard with this BOM verbalization:

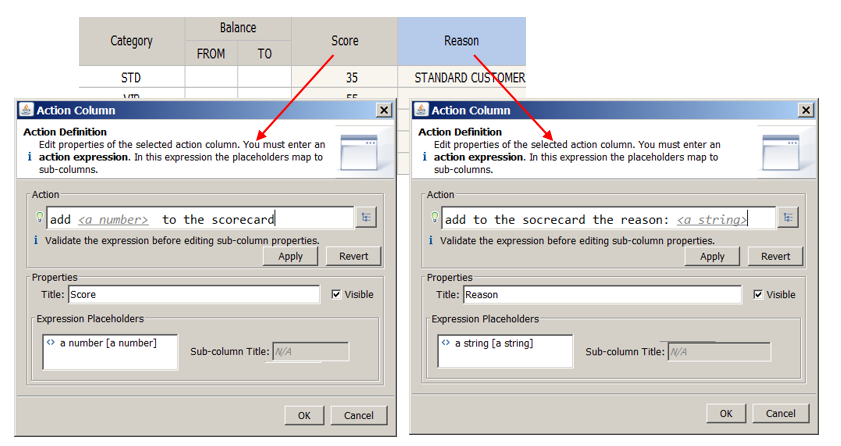
**for** **each** **string** **in** the reasons of the scorecard **"My\_Fraud\_Scorecard"** **:**

- ***print*** **" - "**+ **this string** ;

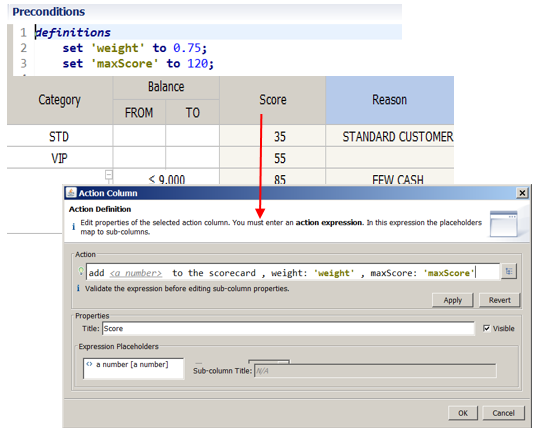
1. **Very Important**: you need to initialize the scores, typically at the beginning of the ruleflow (note that if you miss this configuration, you will get a Null Pointer Exception). See an example on how initialize scores:



1. In a scorecard, condition columns are configured as usual based on your BOM. However you configure action columns using the scorecard BOM. These are the details on how to configure the action columns of previous examples:



1. The scores can weight different for each attribute of the scorecard and a max score can be set for the score cars. The recommended way of doing this is configuring each DT of the scorecard with a “definitions” part where the weight and maxScore are defined. See the example:



1. This is a rule that uses all the available BOM verbalization provided to use scorecards (the rule itself does not have too much sense, just as a reference):

***if***

the score of **"My\_Scorecard"** is more than 100

***and*** the reasons of the scorecard **"My\_Scorecard"** contain **"YOUNG CUSTOMER"**

***then***

add 99 to the scorecard ;

add 99 to the scorecard , weight: 0.75 , maxScore: 150 ;

add to the scorecard the reason: **"YOUNG CUSTOMER"**;

reset all scores;

# Limitations

* You can use blanks in the name of a rule package that represents a scorecard, for instrance “My Scorecard”. However, you should replace blanks by underscores to refer to the scorecard (for instance to get the computed score: the score of **"My\_Scorecard"**)
* The following verbalizations:

add 99 to the scorecard ;

add 99 to the scorecard , weight: 0.75 , maxScore: 150 ;

add to the scorecard the reason: **"YOUNG CUSTOMER"**;

should only be used in the package of a scorecard. Actually the name of the scorecard where the score will be computed is deduced from the package name. So, put all DTs that correspond to the same scorecard in the same package.

* After a score is computed by the ruleflow, you can get access to the score even out of the scorecard packages. In this case you should provide to the verbalization the id scorecard name you want to access (is deduced from the package containing the rule).
* After you call the “reset all scores” verbalization you will no longer get access to the computed scores (it will be cero). This is intended to be used only once at the beginning of the ruleflow.

# Contact

Eduardo Izquierdo

[eduardo.izquierdo@es.ibm.com](mailto:eduardo.izquierdo@es.ibm.com)