

## Internal Firm Rating Model

Anatoliy Antonov, Yanka Yanakieva, Snezhina Petrova

**Abstract:** *Internal Rating Models are intended to evaluate the rating of counterparts using set of objective data and subjective criteria. The paper considers a firm rating model and its implementation using a rule-based approach and the CLIPS language to script the programming interface and the business logic. The script is interpreted by C++ application.*

**Key words:** *Artificial intelligence, Expert system, Rating Systems, Credit Risk, Rating Estimation Model, Computer Systems and Technologies*

### INTRODUCTION

The management of credit risk is one of the most significant activities of banks with lending transactions or of credit institutions. They use quantitative techniques (such as scoring models), on the one hand, and full reliance on the personal experience and expertise of loan and credit officers, on the other. Many banks have in recent years made considerable progress in enhancing these traditional, qualitatively-oriented internal assessments of credit risk by expanding their capabilities for quantifying the credit risk associated with their exposures.

The Basel Committee on Banking Supervision published 2001 a new consultative document [3], known as BASEL II defining two approaches for credit risk estimation - Standard approach and IRB approach (The Internal Ratings-Based Approach). The Standard approach uses external ratings provided by rating agencies such as Moodys and Standard and Poor (SnP). Basel II encourages banks to implement internal rating systems under the condition that banks satisfy minimum requirements to use the IRB approach.

According to the definition of Basel II the term rating system comprises all of the methods, processes, controls, and data collection and IT systems that support the assessment of credit risk, the assignment of internal risk ratings (minimum 7 for non-defaulted borrowers and one for those that have defaulted). Each rating grade defines certain level of risk and quantifies loss or default estimates. A bank must have specific rating definitions, processes and criteria for assigning exposures to grades within a rating system. The rating definitions and criteria used for assigning grades to counterparts must be both plausible and intuitive and must result in a meaningful differentiation of risk. Rating criteria and procedures must differ for different types of borrowers or facilities.

Credit scoring models and other mechanical rating procedures generally use only a subset of available information. Sufficient human judgement and human oversight is necessary to ensure that all relevant information, including that which is outside the scope of the model, is also taken into consideration, and that the model is used appropriately [5].

This paper considers a practical solution of the implementation of IRB approach following the BASEL II requirements. The solution consists of rule-based rating model and Windows application that executes the model script. The model uses CLIPS language to describe questions and results designing the user interface and the rule-based reasoning mechanism. The C++ Application works as interpreter and does not depend on model data (questions and results) and rules.

### COMMON REQUIREMENTS TO RATING MODELS

The most internal rating models used to estimate the credit standing of counterparts against their ability to serve loans and other debt transactions are developed according to the needs of a specific bank institution. Studying the "Best Practice" documents of BASEL II and following the experience of many banks a set of common requirements could be outlined concerning rating models with quantitative and qualitative estimations:

- Use credit risk specific rating criteria and security data
- Use balance data analysis, calculate balance parameter and partial notes

- Use industry branch related criteria and influences
- Use management specific factors and criteria
- Produce scoring and rating for counterparts
- Apply correction on the estimated rating by credit officers
- Integrate early warning system and corresponding KO-criteria
- Weight the separate factors, build rating classes, perform discriminance analysis using unified master rating scale
- Apply default probabilities to map scoring results to rating levels of an rating agency.

Users prefer stand-alone rating systems or rating systems that are easy to integrate in existing software. Their requirements to the "user-friendly" graphical interface correspond to the consultative papers of BASEL II Accord and could be outlined as follows:

- Uncomplicated keyboard data input
- Balance data transfer from other systems
- Flexible definition of rating criteria
- Flexible scoring and rating calculation mechanism
- Customizing the model by updating of questions and rules
- Intuitive operating and handling
- Explanations, notes and hints concerning the input data and the result interpretation
- Documentation of the results

### **MODEL DESCRIPTION**

The firm rating model is based on the assessment of the firm in 3 directions: balance, management and industry branch. Balance sheet data are entered by the user within Balance Data Model session and stored to the data base. A set of 5 main indicators is calculated from balance sheet data using arithmetic expressions [2].

#### *Balance Analysis*

The indicators (in %) are entered by the user or fetched from the database within the firm rating system. The indicator data are converted to partial scoring results within range from 0 to 100 using limits depending on the firm branch. Indicator values from the last 3 years are considered in the assessment of the balance to take in account the absolute or relative trend of the 5 indicators. The historic component of the balance assessment encloses trend estimation of another specific indicator "Firm Result" calculated from balance data using arithmetic formula. Correction of the evaluated scores by the credit risk officer is allowed too. The result of balance assessment is expressed in scores and grade whereas the grade is obtained from the scores using following expression :

$$(100 - \text{scores}) * 0.05 + 1.$$

#### *Management*

Questions for the management assessment are divided into 3 groups, the user enters a grade, possible values are 0, 1, 2, 3, 3.5, 4, 5 and 6. The grades are converted to scores. KO criteria specify questions with high priority which define low limit for the group score result. The result of management assessment (expressed in scores and grade) is evaluated from the group partial scores and the group average grade.

#### *Industry Branch*

The result of branch assessment (expressed in scores and grade) is calculated from the industry branch rating.

#### *Mapping to Rating*

The total result is calculated in scores from the scores of balance, management and

branch assessment and converted to a grade, which is finally mapped by separate rules to external SnP and Moodys rating scale.

### **MODEL IMPLEMENTATION**

The requirements above lead to flexible rule-based model script allowing simple development and configuring of a software rating tool. The model development includes the following steps:

1. Selecting a set of appropriate basic exact criteria that describe a counterpart related to his ability to cover his outstanding obligations.
2. Defining a set of higher-level criteria including top-level results (scoring, rating level). The criteria and results form a variable (fact) space.
3. Creating a rule set that order criteria and results within a dependence tree allowing for a reasoning process. Input criteria, results and rules are written down in CLIPS and stored as rating model in a script file.
4. Saving the developed model in a corresponding script file.

#### **1. MODEL STRUCTURE**

The Rating model script contains all definitions of templates, facts and rules needed to produce counterpart rating according to the implemented model. The model script is independent from counterpart data and it can be common for a large set of counterparts that have similar behavior and that can be described by the same set of criteria and rules. The model scripts are saved in text files with the extension .clp.

The Firm Rating model exposes a standardized section structure:

Access Functions  
CLIPS Functions  
TEMPLATES for Structure (classes, groups, fields, texts)  
TEMPLATES for simple Variables  
Fact Definition (Lists and Properties of GUI elements)  
FACT Hierarchy  
RULE Section

The FACT Hierarchy section contains the basic fact set used to describe input data and results of rating process. The following fixed fact hierarchy is used:

```
dialogroot
  dialogclass
    dialoggroup
      dialogtext
      -----
      dialogfield 1
      dialogfield 2
      -----
    dialoggroup
    -----
  dialogclass
  -----
```

The root includes classes, the classes include groups, the group include fields (questions and results) and texts. Dialog root, dialog classes and dialog groups don't contain any variable values. They are used only to structure the GUI and the user interaction. Input and result variables are present in dialog fields. Rules can apply on any field variables regardless of fact hierarchy (s. Fig.1).

#### **2. FACT DEFINITION**

The TEMPLATES for Structure section contain templates for fact structuring providing templates dialogroot, dialogclass, dialoggroup, dialogfield and dialogtext. This section exists by default and the model developer doesn't change it. The Rating facts (classes,

groups and fields) build a hierarchical structure.

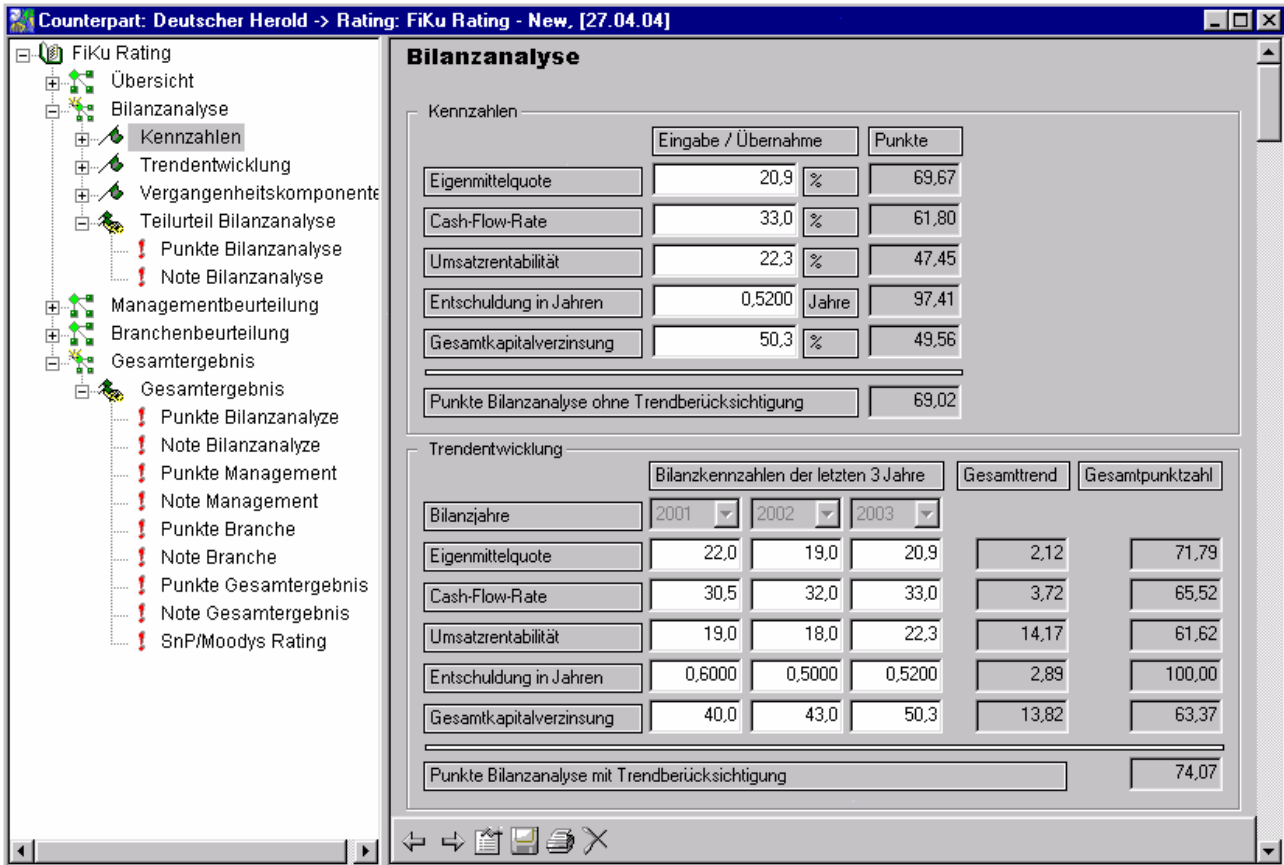


Figure 1: GUI defined by the model script

They correspond to GUI structures window, panel and control. The facts are defined by the statement `deffacts` and `initfacts` and loaded into the fact base after parsing the rating model script. The facts refer to the fact templates and set values (strings) for the template slots. The values define the appearance of the graphical elements (position, style, type, width, height etc.) and remain constant while the user interacts with the GUI and during the inference process. The inference engine works only with the referenced variables for fields linked by the slots named "variableID" that connect facts to simple variable templates. Text fields are provided with slots "question" and "explain" containing descriptions, hints and explanations for the user of the rating model.

Example of field definition:

```
(dialogfield (ID "Score BA")
  (type "EDITTEXT")
  (name "Score Balance Analysis" )
  (control "")
  (x-coord 65)
  (y-coord 405)
  (length 60)
  (height 13)
  (style 2178) ;;ES_AUTOHSCROLL|ES_RIGHT
  (variableID "Score BA")
)
```

### 3. RULE DEFINITION

The RULE Section includes all rules describing dependencies between facts (questions and results). These dependencies are expressed by the general form:

IF facts variables (questions or results) are present in the fact base  
 THEN calculate the value of new or same facts (results) and insert these facts into fact base

The inserted facts can be included in the IF part of other rules causing firing of these rules that insert new facts and so on until high level results are obtained. The inference engine always keeps track of rules, which have their conditions satisfied and thus rules can immediately be executed when they are applicable.

The definition of rules is the most important step during model development. Rules can express dependencies between facts within hierarchic fact derivation tree or trees. Rules are activated (fired) upon presence of a set of facts in the fact base. The reasoning process is data driven, result facts are inserted into the fact base by rules and other rules are activated by inserted facts. Value for result facts can be calculated using a large set of logic or arithmetic functions of CLIPS engine

A part of these rules works with simple variables containing balance sheet data that are combined by arithmetic expressions to produce indicator values. Other rules use indicators to create scores for desired group. High-level rules produce final scoring results for the rated firm and try to map the scoring results to ratings within a selected rating order.

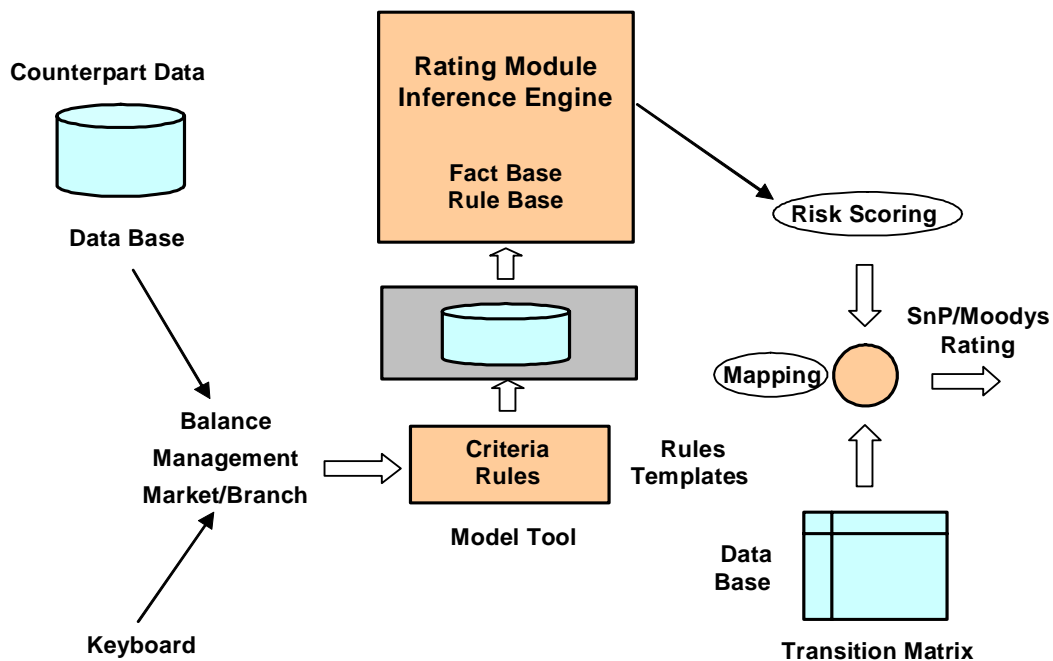


Figure 2: Rating Procedure using the Firm Rating Model

Rule Example:

```
(defrule Rule_Partial_Branch_Score
  (Branch_Points (value ?BP))
=>
  (assert (Branch_Score (value (+ (* (- 100 ?BP) 0.05) 1)) ))
)
```

## **CONCLUSIONS AND FUTURE WORK**

The practical implementation of the developed rule-based firm rating model within the Risk Rating module in German Banks leads to following conclusions:

1. The rule-based rating model builds the kernel of an expert system for the evaluation of the credit rating of a firm counterpart or corporate.
2. The rule-based rating model ensures a run time flexibility allowing the user to change the variables, to adjust or to change the rating questions and rules.
3. The firm rating model works with exact values but in some practical cases there is no information or the available information is not exact. The firm rating model could be enhanced to work with fuzzy logic and confidence distributions expressing inexact or fuzzy values.

The future development of rule-based rating models is bound to the improving of the Risk Rating module functionality, it should be extended to allow including of new GUI elements in the models - for example ActiveX controls, pictures, animations, and why not explanations and hints in sound, to ensure friendly and pleasant work with the rating model.

## **REFERENCES**

- [1] Kasabov K. Nikola, Foundations of neural networks, fuzzy systems and knowledge engineering, 1998, MIT Press, USA.
- [2] RiskMetrics - Technical Document, Third Edition, 1995, New York
- [3] Basel Committee on Banking Supervision, "The internal Ratings-Based Approach", Supporting Document to the New Basel Capital Accord, 31 May 2001
- [4] Петър Благоев, Оценка на кредитния риск, Банкови информационни технологии - ComputerWorld/Пролет 2003
- [5] Basel Committee on Banking Supervision, "The New Basel Capital accord", Consultative Document, 31 July 2003

## **ABOUT THE AUTORS**

Assoc.Prof. Anatoliy Stefanov Antonov, PhD, Department of Computer Systems and Technologies, Technical University of Varna, Phone: +359 52 220003,

E-mail: antonov@eurorisksystems.com

Yanka Nedelcheva Yanakieva, Department of Computer Systems and Technologies, Technical University of Varna, E-mail: yanakieva@eurorisksystems.com

Snezhina Kamenova Petrova, PhD Student at Department of Computer Systems and Technologies, Technical University of Varna, E-mail: sn\_petrova@yahoo.com