Building Portfolio Analysis System Using Ontology-based Information System Builder

Samuil Nikolov
Eurorisk Systems Ltd.
31, General Kiselov Str., 9002 Varna, Bulgaria
E-mail: samuil at euroriskssystems dot com

Abstract

This publication describes the way to build a portfolio analysis system using a specially developed tool that is used to quickly and easily build information systems. The tool is developed in C++ and uses CLIPS expert system environment. Building the portfolio analysis system requires disseminating the problem into an ontology, creating small script files that describe each ontology class and defining their data using the information system builder.

Keywords
Information systems, portfolio analysis, software tools

INTRODUCTION

The goal of this publication is to introduce a way for building financial portfolio analysis systems using an ontology-based information system(IS) builder. According to the latest definition in [1] by Tom Gruber, ontologies define a set of representational primitives with which to model a domain of knowledge or discourse. The use of ontologies in information systems was first introduced by Wand and Webber where they defined a formal model of information systems [6]. The term ontology-based information system was first introduced by Guarino [2] in his analysis in what ways an ontology can be used in information systems. The approach described in the current publication is using ontologies in both user interface and in application components as described in Guarino's publication.

The used information system builder is facilitating disseminated ontology data in the form of separate script files, each representing one of the ontology classes. The classes contain the business logic of the information system and describe its user interface. The general principles of operation [3] and the database structure [4] are presented in other publications with participation of the author and are not subject of this paper. The developed IS builder allows creation of a complete information system by creating a set of CLIPS programs that define all the classes of the target ontology. The script files are written in CLIPS rule based programming language. They are loaded in a CLIPS environment used by a C++ core developed by the author. CLIPS was chosen as a widely used productive development and delivery expert system tool which provides a complete environment for the construction of rule and/or object based expert systems thus allowing effortless C++ integration. To build the target portfolio analysis system, one must first define the underlying ontology, then build class definition script files and as last step – define their instances inside the information system builder.

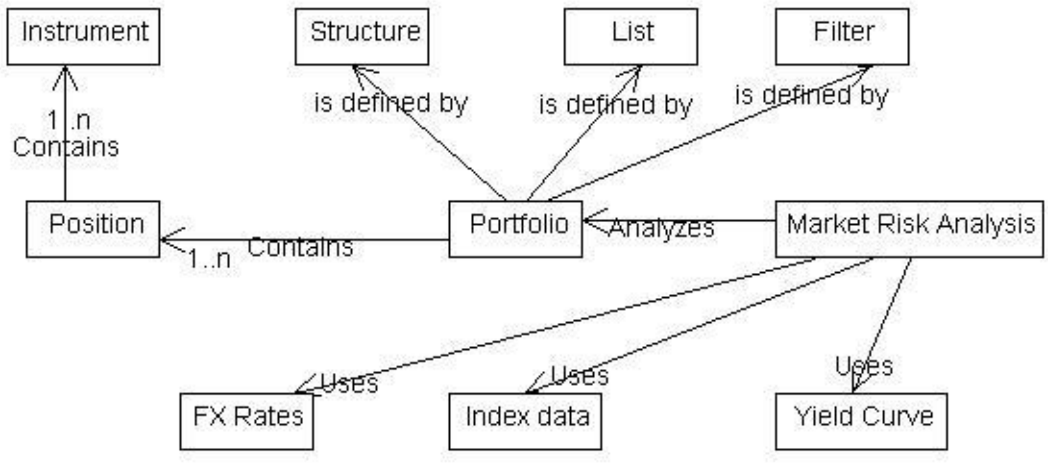
1. Defining Portfolio Analysis Ontology

The first step in building a portfolio analysis system using the IS builder is disseminating a portfolio analysis into a set of concepts or ontology classes. Analysing a portfolio contains several points:

- Clearly defining all parts of the portfolio
- Clarifying the environmental conditions, the analysis will be using
- Deciding the kind of portfolio analyses performed

Having these in mind, a simplified version of the example ontology that will be used is shown on figure 1:

Figure 1: Diagram of the ontology used for creating the portfolio analysis system



A financial portfolio contains a set of financial positions that can include one or more financial instruments. For instance, a plain vanilla swap position contains two instruments – one fixed rate loan and one floating rate loan. In the illustrated ontology, the "instrument" concept defines how will the payments concerning a contract be made – examples of instruments are loan, deposit, bond contracts, stocks. The "position" concept defines the direction of the contract - long or short, its currency and nominal value. Most of the portfolios are not a simple set of positions but a hierarchical structure of sub-portfolios. The ontology class "structure" from figure 1 defines how many and what sub-portfolios the "portfolio" concept contains. The subset from the all positions in the system that define each sub-portfolio is determined using "list"s and "filter"s by either directly choosing position identifiers or specifying some of their parameters for selection. Another part of the portfolio analysis are the environmental conditions that will be used. In figure 1 there are three external conditions, in this case market factors, that will be used in the example market risk analysis - Currency rates, Market Index data and Yield curves. Other analyses can make use of another set of external conditions, that also have to be defined in the ontology.

2. Creating script files for ontology classes

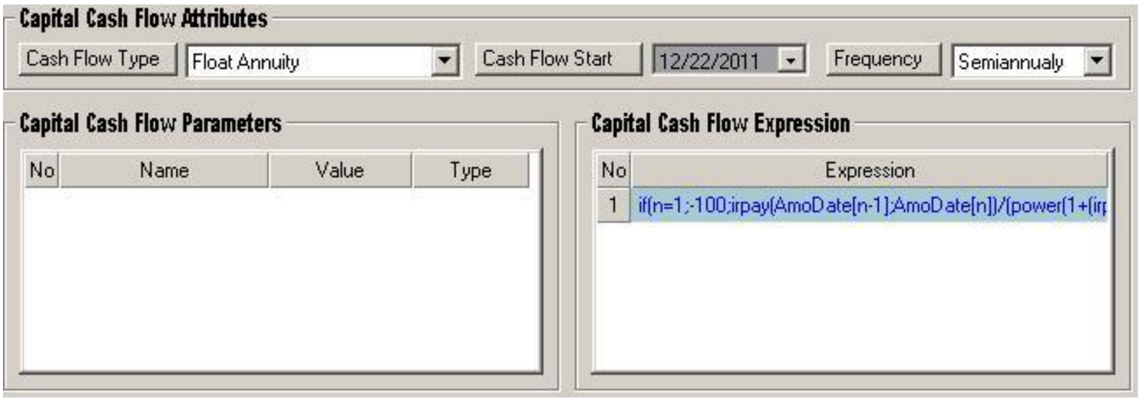
Each of the ontology classes from figure 1 must be defined in its own script file. The scripts contain information about what will be stored in the database for the specific class, how it will be visualised in the IS builder system in execution time and what operations should be performed on the visualized fields when the class is activated – this option is used to obtain some results from the data input by the user. The structure of the script file is not subject of the current publication and will not be discussed. As an example, to create a script file for the "instrument" class, one must define for storing and visualization several attributes among which:

- Starting and maturity dates of the instrument
- Amortization start date, frequency and type (regular, bullet, annuity, custom expression-defined, etc.)

- Interest rate payments start, frequency, irregular period start or end, type (floating, fixed, capped or floored floater, custom expression-defined, etc.)

Parts of the user interface generated for the instrument class that is used in the portfolio analysis system is shown on figures 2 and 3. They contain fields that are used to input contract-specific values for the "instrument" object being defined in execution time. Besides being visualised, the information from these fields is also stored in a database for use by the portfolio analysis. Some of the scrips may contain calculation rules that can yield results from the input fields. For instance, the "instrument" script of our portfolio analysis system contains rules that generate payment schedules and calculate the main parameters of the instrument's cash flow for a certain evaluation date – present value, accrued interest, internal rate of return, etc.

Figure 2: UI generated for amortization schedule attributes



Capital Cash Flow Attributes

Cash Flow Type	Float Annuity	Cash Flow Start	12/22/2011	Frequency	Semiannualy

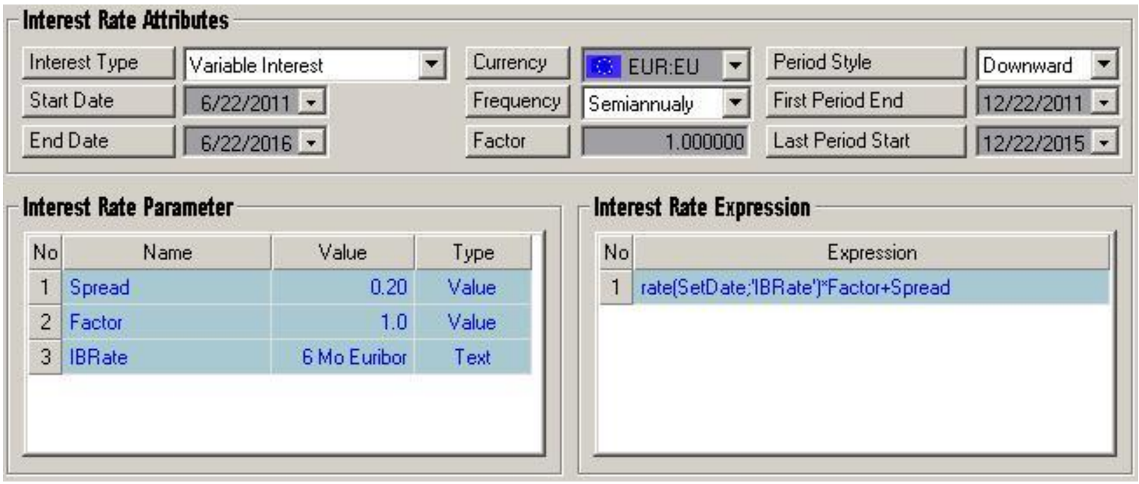
Capital Cash Flow Parameters

No	Name	Value	Type

Capital Cash Flow Expression

No	Expression
1	if(n=1;-100;irpay(AmoDate[n-1];AmoDate[n])/(power(1+(irp

Figure 3. UI generated for interest rate schedule attributes



Interest Rate Attributes

Interest Type	Variable Interest	Currency	EUR:EU	Period Style	Downward
Start Date	6/22/2011	Frequency	Semiannualy	First Period End	12/22/2011
End Date	6/22/2016	Factor	1.000000	Last Period Start	12/22/2015

Interest Rate Parameter

No	Name	Value	Type
1	Spread	0.20	Value
2	Factor	1.0	Value
3	IBRate	6 Mo Euribor	Text

Interest Rate Expression

No	Expression
1	rate(SetDate;'IBRate')*Factor+Spread

Similar UI definition scripts are prepared for all the defined classes of the ontology. Position class has fields for inputting nominal, direction, currency, price paid for the position, a table containing the identifiers of the involved instruments, etc. Another feature of the IS builder used is that it is able to export the data input in the fields in various external formats like XML [5].

3. Defining instances of the ontology classes

To calculate the market risk of their portfolio, the users of the information system builder must create, define and store to the database instances of all the classes from the ontology. This involves running the information system builder, loading each of the script files, entering data for most of the fields in the UI and storing the data in the database. For example, in the "Instrument" instances, one has to input debt amortization type, debt and interest payment frequencies, interest size etc. While defining the "Position" instances the user has to link the described position to one or more database-stored instances of the instrument class and then specify the volume traded, whether the position is long or short, the currency used for trading. The classes "Filter", "List" and "Structure" define the contents and the sub-portfolio structure of a "Portfolio" instance. Filters define rules for selecting particular instances of the "Position" class from the database while lists specify them directly by their unique identifiers which usage and forming is explained in [3]. The classes "FX rates", "Index data" and "Yield Curve" are used to store corresponding market data for specific market centres and dates. After creating instances of all classes of the ontology, an instance of the class "Market Risk Analysis" is opened. It contains analysis parameters like evaluation date, risk measurement horizon, confidence level and number of runs for the Monte Carlo Simulation of the market variables and lots of output fields, tables and charts for showing the results of the analysis. On activation, the rules inside this class load one portfolio instance from the database using the functions added to CLIPS from the C++ core. Then its structure and the contained positions and instruments are loaded according to the data stored inside the portfolio instance performing complex filtering in the process if necessary. After loading all necessary data, the market risk of the defined portfolio is calculated, and various results of the calculation are visualized on the output controls. Those include the total value at risk (VaR), currency VaR, interest VaR, capital VaR, their incremental and marginal versions and many more. The same portfolio definition ontology with some minor additions is used in several other portfolio analyses like asset and liability management, credit risk assessment, interest income analysis, cash flow analysis, regulatory requirements.

CONCLUSIONS AND FUTURE WORK

The described way for building portfolio analysis systems using an ontology based information system builder is used in a commercial system. Currently, it is used for creating financial analysis systems, but as seen from the method described in the publication, it can be easily and quickly suited to any other field if a detailed ontology is available. Building even a complicated portfolio analysis system requires several quick and simple steps like disseminating the problem into basis concepts, defining the visualization of the attributes of the concepts and at last, defining specific instances of these concepts inside the information system builder.

[1] T. Gruber. «Ontology, » in the Encyclopedia of Database Systems, Ling Liu and M.Tamer Özsu,Ed.New York: Springer-Verlag, 2009, pp. 1963-1965
[2] N. Guarino. «Formal ontology and information systems, » in Proc. FOIS'98, 1998, pp. 3-15
[3] S. Nikolov and A. Antonov. «Framework for building ontology-based dynamic applications, » in Proc. CompSysTech, 2010
[4] S. Nikolov. "Storing data of ontology-based dynamic applications," in Proc. 6th International Scientific Conference Computer Science, 2011, pp. 134-139
[5] S. Trifonova and S. Nikolov. „XML presentation of financial instruments," in Proc. Unitech, 2010
[6] Y. Wand and R. Weber. “An ontological model of an information system,” IEEE transactions on Software Engineering