

XML PRESENTATION OF FINANCIAL INSTRUMENTS

Sevdalina Trifonova
Technical University of Varna

Samuil Nikolov
Technical University of Varna

Abstract

This paper presents a methodology based on the XML technology, for describing financial products, their cash flow streams and result of different risk analysis. The instrument's parameters and results are presented via two XML structures. The first structure is used for description of financial instruments and dynamical GUI generation. The second one for the purposes of import and export of financial instruments from the system. XML presentation of financial instruments gives an ability to create instrument templates for offered financial products by a client.

Keywords: Computer System and Technologies; Extensible Markup Language (XML); Financial Instruments; GUI; Software applications.

INTRODUCTION

Financial institutions offer a wide range of financial products to their clients. The most popular debt financial instruments are bond (market instrument), credit and deposit (over the counter instruments, contracts for borrowing/lending money between financial institutions and their clients). But there are a lot of complex instruments and derivatives on the market which consist of more than one payment structures.

Financial institutions manage risks (credit, market, operational, liquidity) according to the supervisory requirements (standard approaches) or internal models and internal rules. They estimate value at risk, expected profit or loss, do simulations with different future scenarios on interest rates, foreign exchange rates, credit quality and prices

Hard coded systems are not adequate for describing this variety of conditions because the appearance of a new condition leads to changes in procedures and GUI.

The main aim was to find out common way to describe the instruments structure and their parameters, risk values and risk reports. XML offers a simple intuitive way of describing and creating tags, DTDs and XML Schemas.

This capability has resulted in a huge number of markup languages specifying the inherent commonalities present in various market sectors.

One of the markets that has taken this descriptive ability to heart is the Capital Market. Currently there exist different standards for XML presentation of:

- financial instruments on capital market FIXML is used to describe information exchanged on standardized instruments while FpML describes information for over the counter derivative instruments – Swap, FRA) [8]
- based on eXtensible Business Reporting Language (XBRL) - Common Reporting Framework (COREP) is the common solvency ratio reporting framework for credit institutions and investment firms under future EU capital requirements regime. Financial Reporting Framework (FINREP) is designed for credit institutions that use IAS/IFRS for their published financial statements.[7]

But these standards cover different scope than the methodology described in the paper. The focus in this paper is presentation of financial products with complex structure offered to banks' client – retail or corporation.

EXPOSITION

1. XML presentation of financial instruments describing parameters of the contract and GUI

The parameters of a cash flow instrument can be subdivided into several main groups:

- parameters of a contract – instrument's identification information, issuer, contra-gent, currency, market
- amortization schedule – amortization type, amortization frequency, tranches
- interest rate schedule – interest rate payments frequency, period start, end, payment dates

Figure 1 presents the structure of the model 'Cash Flow Instrument' in Risk Framework (application for calculation of financial risk) [5],[9]. Instrument definition is presented via a hierarchy of dialogs, group boxes and elements generated by script based user interface definition [4]. The instrument's data is subdivided into 5 main groups:

- Instrument Definition - contains the instrument's identification information and result of the latest valuation – theoretical value, market price, risk parameters as duration, base point sensitivity etc.
- Amortization Schedule – used for definition of the debt and amortization cash flows
- Interest Rate Schedule – used for definition of the interest rates and shows interest rate cash flows and related information
- Cash Flow Schedule - shows the common cash flow - amortization and interest rate payments
- Instrument Details - contains conventions (date roll, financial center etc.) for payment and fixing, interest rate and valuation attributes (discount curve, spread, activated scenario)
- Administration of the session – created, modified, last valuation, responsible person

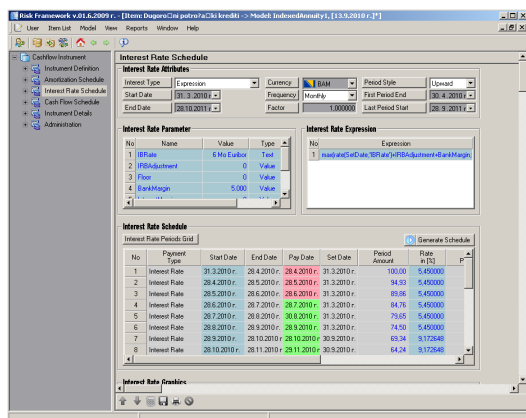


Fig. 1. Definition of a cash flow instrument

The structure of the XML starts with root element <REPORT></REPORT>. Then follows information about the user, last modification date, module in Risk framework [9], session identifier etc

<Date d="7" m="9" y="2010" /> - Last edition date

<User Name="Trifonova" ID="User 007" PreName="Sevdalina" /> - User ID

<Modul_Type>Instrument_Data</Modul_Type> - Module of the Risk Framework System

<Item Name="IRBRS" ID="1094_6-0762090062" /> - Item description

<Session_Name>IndexedAnnuity1</Session_Name> - Session identifier

<Model_Name>Cashflow Instrument</Model_Name> - CLIPS model name

<Unit Name="ALM" ID="001005" /> - branch to which the instrument belongs

The GUI is based on a box model shown on fig.2. Root level template control is the root element. It includes a set of dialog controls. The main 5 groups are presented as dialog box template controls, including a set of group boxes.

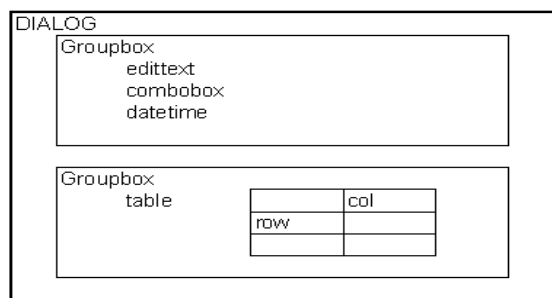


Fig. 2. Box model of GUI

The part of XML describing the parameters of cash-flow instrument starts with:

```
<ROOT>
<DIALOG>
  <GROUPBOX>
    <EDITTEXT/>
    <COMBOBOX/>
    <LISTBOX/>
  </GROUPBOX>
  .....
</DIALOG>
.....
</ROOT>
```

Majority of the controls in cash flow instrument are described with tags : <EDITTEXT/>; <COMBOBOX/>; <BUTTON/>; <DATETIME/> etc. Attributes of these tags are:

ID –identifier of the control

Name – user friendly name, used for label assigned to the control

Type – data type (string, number, date etc)

Value –value of the parameter

Min –minimal allowed value

Max- maximal allowed value

TreeLevel – describes the hierarchy from root element to the current control

The functionality is implemented into the CLIPS model and the relationship between CLIPS model and XML presentation of the instrument is based on unique identifier of the control.

Table data are presented via the XML structure:

```
<GRID ID="identifier"
NAME="description" TreeLevel='1.1.2' >
  <ROW>
    <COL type='data type'>
      'Cell Value'
    </COL>
    .....
  </ROW>
  .....
</GRID>
```

Using this XML structure it is possible to describe not only financial instruments but market data and valuation results, also in table format required by the supervisor. Thus presentation of all information into the system can be shown in Internet browser via uniform XML transformation.

1.3.1. Interest Rate Attributes			
Name	Value	Min	Max
1.3.1.1. Interest Type	Expression		
1.3.1.2. Interest Type Save	Expression		
1.3.1.3. Interest Start Date	01/03/2010		
1.3.1.4. Interest End Date	01/04/2014		
1.3.1.5. Interest Currency	BAM		
1.3.1.6. Interest Frequency	Monthly		
1.3.1.7. CF Factor	1.00	-1000000000000.00	10000000000000.00
1.3.1.8. Interest Period Style	Upward		
1.3.1.9. First Period End Date	01/04/2010		
1.3.1.10. Last Period Start Date	01/03/2014		

1.3.2. Interest Rate Parameter			
No	Name	Value	Type
1	IRRate	5 Mo Euribor	Text
2	IRBAdjustment	-1.00	Value
3	Floor	3.50	Value
4	BankMargin	-3.000	Value
5	InterestMargin	3.600	Value
6	FixSchedulePeriod	6	Value
7	FixScheduleStart	31.03.2010	Date

Fig. 3. Result of XML Transformation to HTML

Usually financial instruments can be described with basic constructs for amortization (annuity, bullet, regular) and for the interest rate (fixed or floater rate based on an inter-banking rate). The amortization and rate payment dates are determined according to standard frequency conventions (Monthly, Quart, Annual etc.).

But the products often consist of different non-standardized amortization plans that match clients' work-cycle and complicated definition of interest rates (interest rate is a function of more than one reference interbanking rate, or optional rights to achieve maximal profit and to limit loss (cap option defines maximal interest rate, which is payable, floor – minimal interest rate)

Non-standardized amortization and interest plans are defined via a list of user defined parameters and formulas for respectively amortization and interest payments. The formulas are defined according to the context-free grammar rules. [3]

Description of the non-standardized instruments requires good knowledge of financial instruments offered by the bank on one side, and from the other side it takes time.

The described XML structure allows standardization of those non-standard instruments and easier use by bank employees. This is achieved by providing template XML descriptions of the complex financial instruments that contain the proper formulas and standard parameter setups – like daycounting conventions and contract's currency. The bank employees need only to specify some contract-specific parameters like start and maturity dates, frequency, the interbanking index used to calculate the interest rate, its maximal and minimal values if these are applicable. The template contains all the necessary formulas and frequency settings according to the bank's product.

An example with a particular non-standard contract will be given to illustrate the usability of the proposed solution. The contract has equal annuity payments which are split between repaying the debt part and paying interest rate. It is linked to bank-specific rate, which is calculated using an interbanking index, but with a complicated bank-defined formula. The formula makes use of the country's interest margin, which is dependent on the country's credit rating and the bank's margin which is dependent

on the current state of the bank's finances as well as on the internal rating given to the specific customer.

The XML template for such an instrument, besides the standard settings and default values for most parameters, contains the two formulas - one for amortization payments calculation and one for interests nested within XML grid descriptions to enable multiline formulas:

```
<GRID ID="IRATE_EXPRESSION_GRID"
Name="Interest      Rate      Expression"
TreeLevel="1.3.3.1.">
  <ROW>
    <COL type="string">Expression</COL>
  </ROW>
  <ROW>
    <COL type="string">
max(rate(SetDate;'IBRate')+IRBAdjustment+
BankMargin;Floor)+ InterestMargin
    </COL>
  </ROW></GRID>
```

The interest rate is being calculated according to the formula, supplied by the bank, that takes into account the parameters IBRate, IRBAdjustment, BankMargin, Floor and InterestMargin. The parameters are free defined in grid

```
IRATE_PARAMETER_GRID      <GRID
ID="IRATE_PARAMETER_GRID"
Name="Interest      Rate      Parameter"
TreeLevel="1.3.2.1.">
  <ROW>
    <COL type="string">IBRate</COL>
    <COL type="string">6 Mo Euribor</COL>
    <COL type="string">Text</COL>
  </ROW>
  <ROW>
    <COL type="string">
      IRBAdjustment</COL>
    <COL type="string">0.7</COL>
    <COL type="string">Value</COL>
  </ROW>
  .....
</GRID>
```

The employees do not need to know the complex formulas that are involved in calculating the actual principal and interest payments, they just need to insert the parameters at the proper places, even default values are filled in template.

Risk Framework [5] system gives an ability to reduce dramatically the exchanged information

for financial instruments. The template contains all the necessary formulas and frequency settings according to the bank's product while the XML used for importing data contains only the actual (non-default) contract parameters. The import mechanism merges the two XML files to form a full instrument description, thus easily defining an otherwise nonstandard financial instrument involving complicated formulas to determine its interest and amortization payments.

2. Financial instruments – import and export based on XML

Risk Framework [5], [9] is platform independent and supports different database vendors. Many differences between vendors and necessity to migrate information from one vendor to other lead to development of the special Export- Import module based on the other XML format for presenting the information.

XML is vendor independent and system independent. While data is being exchanged using XML, there will be no loss of data even between systems that use totally different formats. Many of the vendors has ability to export – import information from tables to XML, but each vendor uses own format.

The structure of DB is shown on the fig.4. The data of financial instruments are subdivided into three groups:

- head information (RF_HEAD) – instrument id, model name, financial center, user and time stamp of creation, last modification or deletion
- body (RF_BODY)- details of the deal. Each parameter of the deal is placed on one row of the table. The columns of the table describes the ID of parameter, its data type, minimal, maximal allowed value, value of the parameter etc
- table data presentation with description of the header (RF_TAB_H) and the values in the cells (RF_TAB_B)

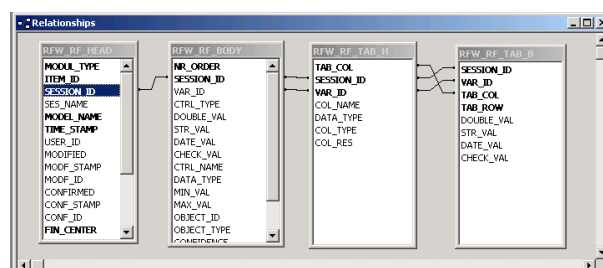


Fig. 4. Database tables and relationship

The designed XML structure repeats the logic of DB structure. Root element is <SS>. The length of the tag names is maximum three symbols. The aim is to reduce the size of the output file.

<H> tag is the open tag for head data. Some of the others are:

```
<H>
  <MT v="Instrument_Data" /> - module
  <II v="1094_6-0762090062" /> -identifier
  <SN v="IndexedAnnuity1" /> -model name
  <TS v="2010-08-19" /> - timestamp
  <UI v="Imp" /> -user id
  <FC v="001005" /> -financial center
  owner of the instrument
  <B> - start of body
    <BR> -element definition
    </BR>
  </B>
</H>
```

New tag
 is opened for each parameter of the financial instrument:

```
<BR>
  <NO v="14.0" /> - sequence number
  <VI v="InstrumentType" /> - identifier
  <CT v="COMBOBOX" /> - GUI
  control type
  <DV /> -date value
  <SV v="Annuity Loan" /> -string
  value
  <DTV /> -datetime value
  <CV /> -check box value
  <CN v="Instrument Type" /> - user
  friendly name
  <DT v="L" /> -code of data type, List
  of string values
  <MIV v="0.0" /> - minimal value
  <MAV v="0.0" /> -maximal value
  <TL v="1.1.2.2." /> - tree level,
  placement in hierarchy of dialog and
  groupboxes
  <T></T> - table
</BR>
```

Table data are presented in substructure:

```
<T> -table
  <TH> - table header
  <HR>
    <TC v="1.0"/> - table column
    <CN v="Duration"/>-name
  </HR>
```

```
</TH>
<TB> - table data
  <CL>- cell
    <TC v="1.0"/> - table column
    <TR v="1.0"/> -table row
    <DV v="7.32"/> -double value
    <SV /> - string value
    <DTV/> -date value
  </CL>
</TB>
</T>
```

CONCLUSION

This article has proposed XML definition for presenting complex financial instruments. Such an approach has a range of advantages including:

- Provides methods for describing financial products based on platform, system and vendor independent format;
- Provides the opportunity to create templates for any of the financial products offered by a bank. It makes the system more efficient and reduces operational risk.
- Provides the possibility to export and import financial instruments.
- Currently the validation of XML is subject to document type definition(DTD), but a XML schema is in process of development

The methodology, presented in this article has been tried out by financial institutions in Bulgaria and Bosnia and Herzegovina and is in a stage of testing. The approach is also set to be used in financial analyses of contract portfolios.

REFERENCE

- [1] Fabozzi Frank J., Peterson Pamela P., Financial Management and Analysis, Second Edition, John Wiley & Sons, Inc., 2003.
- [2] Fabozzi Frank J., Fixed income Mathematics, 1993, Probus Publishing Company, USA
- [3] Trifonova S.,Nikolov N., Nikolov VI., Debt Instruments –Definition and Evaluation of Cash Flows via Expressions, Informatics in the Scientific Knowledge, 2010, pp 182,194.
- [4] Paskalev Pl., Nikolov VI., Multi-platform, script-based user interface, CompSysTech, 2004, pp IIIB.14-1, IIIB.14-6.

- [5] Asset Liability Management in Risk Framework-
http://www.eurorisksystems.com/download/Presentation_ALM_EuroRisk_EN.zip
- [6] World Wide Web Consortium (W3C), XML Standards <http://www.w3.org/standards/xml/>
- [7] Extensible Business Reporting Language, COREP and FINREP projects – <http://www.eurofiling.info/documents>
- [8] FIXML, Financial Information eXchange (FIX) protocol - <http://www.fixprotocol.org>
- [9] Nikolov S., Antonov A., Framework for building ontology-based dynamic applications, ACM International Conference Proceeding Series (ICPS), Vol. 471, p. 83-88