

THE ARCHITECTURE OF AN EXPERT MEASUREMENT SYSTEM FOR THE DYNAMIC MODELING OF THE STOCK MARKET

Dan NIȚESCU¹

În cadrul activității de cercetare “Sistem de Măsură Expert pentru Modelarea Dinamică a Pieței Bursiere” am urmărit înțelegerea și formalizarea tranzacțiilor pieței bursiere și realizarea unui Model Dinamic care implementează această înțelegere. Trecerea de la Geometria Fractală, care descrie tranzacțiile pieței bursiere într-un mod natural, la modelul dinamic formal bazat pe folosirea Lanțurilor Markov este contribuția mea originală în acest domeniu. Articolul detaliază cum s-a realizat în mod practic o astfel de transformare, care sunt instrumentele informatice de care m-am folosit și nu în ultimul rând criteriile folosite la alegerea lor. Rezultatul muncii mele de cercetare se regăsește în arhitectura sistemului ce integrează toate modulele specifice.

In the research activity framework “Expert Measurement System for the Dynamic Modeling of the Stock Market” I have been seeking the understanding and formalization of the stock market transactions [6] and to realize a Dynamic Model which implements that understanding. The transformation from Fractal Geometry [7], which naturally depicts the stock market transactions, to the formal dynamic modeling, underpinned using Markov chains [8], is my original contribution to this domain. The paper details the way such a transformation has been practically done, the software tools I have been used and last but not the least which were the criteria for using these tools. The result of my research work can be found in the system architecture which is integrating all specific modules.

Keywords: transaction, stock market, fractal geometry, Markov chain, software tools, choosing criterion, system architecture.

1. Introduction

The correct performance of an information system is guaranteed by the modeling process which captures in a suitable manner all the functional conditions imposed by the specification of the system behavior.

Such a modeling task is feasible since there has been proved that any limited domain taken from real life can be faithfully modeled, and the model provides acceptable answers. These answers give supporting evidence to the modeled domain and, therefore, are of significant utility.

¹ PhD Student, Dept. of Computer Science University POLITEHNICA of Bucharest, Romania, nitescu.dan@gmail.com

A model is said predictive if it can produce satisfactory answers to essential aspects of dynamic process prior to the concrete realization of the same process. The utility measure of a predictive model is measured by the quality of the corroboration between the model answer and the real behaviour of the domain which the model is describing. The specific research domain that I have selected is the dynamics of NASDAQ quotes [9].

The project's specification for the actual system is detailed in the following task list:

1. online data acquisition for the stock market transactions;
2. structuring of the acquired data;
3. finding the relations inside the “container structure”;
4. the calculation of the transactions conditions used by the model;
5. forecasting “the transactions price level” by means of the synthesized dynamic model
6. the forecasting presentation as a practical report;
7. comparing the model answer against the effective price level realization for NASDAQ quotes;
8. computing of the system's confidence factor.

Others administrative tasks which directly contribute to the reliable system behavior are:

- a) securing redundant data communication by at least 2 independent net connexions by means of reliable net providers;
- b) securing system's recovery by means of the backup service;
- c) securing system's protection by means of VPN, Virtual Private Network;

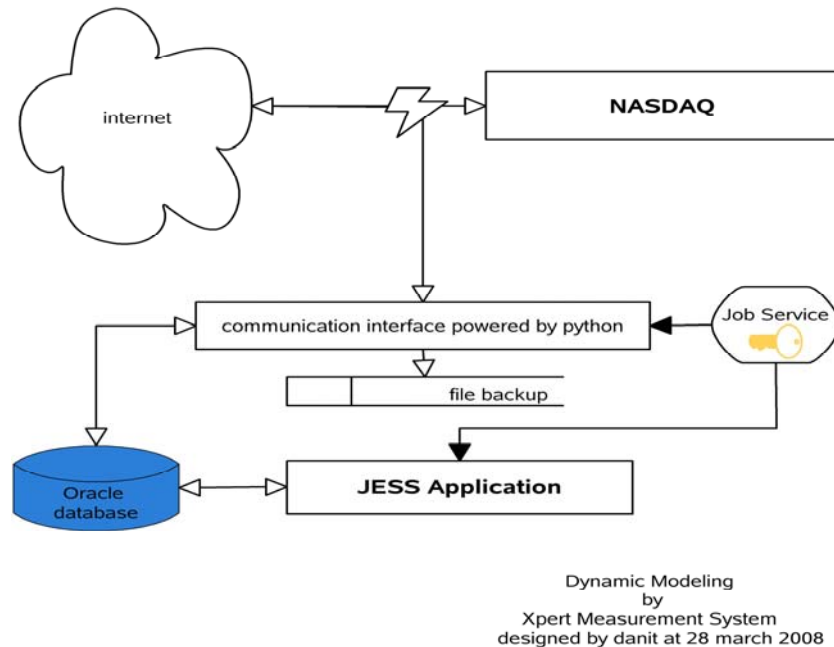
2. The Technology Used for the System Development

To answer to the imposed jobs for the practical system realization I selected the following software tools:

- a) Linux platform, choosed to secure system's protection during production and also for good software compatibility;
- b) Python, for the realization of data acquisition module and others administrative jobs [10];
- c) Oracle database, for records management and data processing and as well as for the result publishing, tool choosed for its «unbreakable» features;
- d) JESS technology, common interface between Oracle database and the Expert Subsystem, tool choosed for the integration role by means of an efficient pattern matching algorithm [11], [12];
- e) JAVA technology.

3. The System Architecture

Fig. 1 The modules and technologies utilized in the system development



- A. The jobs execution are scheduled by the “Job Service” module, which is symbolized by a key Fig. 1, and it is implemented by the crontab;
- B. The communication interface ensures the system's link with “the outside world” for the data acquisition module as well as for the eventual actions to be carried out after the data processing and the answer generation. In the same time the communication interface also provides the automatic backup service for the files imported by the acquisition module;
- C. Oracle database occupies the central place in the system architecture because it is ensuring the greatest number of roles during the running of the system;
- D. The JESS Application is specialized in the expert role of the answers integration and the answer's publishing.

4. One of the Possibilities to Describe the Stock Market Transactions

As any dynamic system the Stock Market Transactions model as well can be described by the help of a multitude of equivalent instruments.

The distinctions between different information instruments are chiefly:

1. the specificity of the tackled aspects;

2. the domain activity;
3. the selected system politics, as the priorities in the system production, the administration and optimization of the utilized resources and others;
4. the efficiency in the system developing.

That efficiency is measured by the value of the found answer, at the finale of the system developing cycle, related to the quantity of the utilized resources.

A good decision is to assimilate the time series [13] of the stock market transactions to fractal geometry, which from my point of view directly and naturally answers to the most real aspects of the stock market. Michael Frame and Benoit B. Mandelbrot, from the Mathematics Department, Yale University, have before myself been reached the same conclusion [14]. One of the first published articles with this subject is “A Multifractal Walk Down Wall Street” by Benoit Mandelbrot [15]. That is the start in the actual research and where the model_of_time and Markov chains are used for further development, as next it will be depicted.

It is nothing new as far as the fractal geometry had been described since 1975 by Benoît Mandelbrot when he published his ideas in the article “Les objets fractals, forme, hasard et dimension” [16].

The naming of “Fractal Geometry” concept had been derived from the latin word “fractus” with the meaning "broken" or "fractured" [17]. It is easy to understand the description naturalness of the stock market transactions by means of the fractal geometry and why this tool is preferred.

Starting from Benoît Mandelbrot's explanation of the time series of the stock market transactions and how it is possible to be assimilated with the fractal geometry, that is the approximation of those series by means of a generatrix fractal function, from his article published for the first time in the February 1999 issue of Scientific American (as in [15]), the researcher should put himself the question: “How can I find the actual parameters of this fractal function?”.

The answer to that challenge relies on a model of time defined by the structure built with the time unities: second, hour, trade day, month, year. The structure of this model_of_time is of fractal type where any time sub-sample is included into a hierarchical tree structure. The details of this practical answer are shown immediately below together with all the features of the generic sample used on all the model_of_time's levels, Fig. 2.

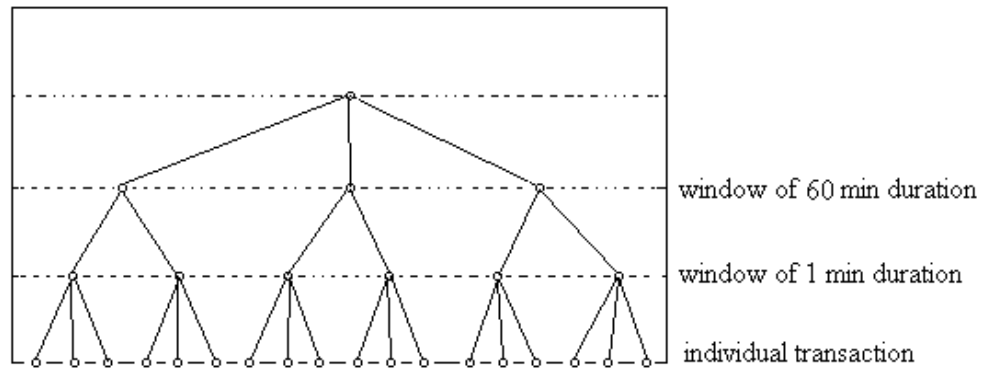


Fig.2 The transactions clustering in windows, of constant duration, and in a tree structure

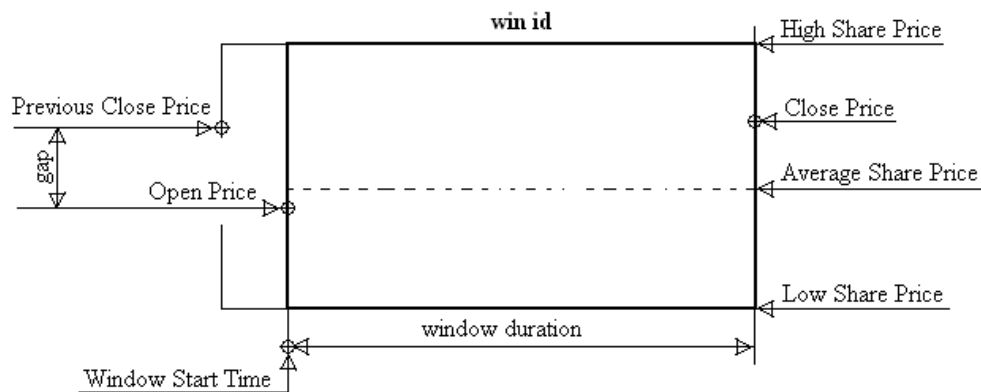


Fig.3 The parameters of a window element identified by a unique win id

As it can be seen from the Fig.2 the individual transactions are clustered [18] into windows of 1 minute duration, for the first step, and into windows of 60 min duration, for the second step.

Others parameters for the window element, Fig.3, are Transaction Cluster Volume and its Cardinality.

5. The Transformation of Time Series for the Stock Market Transactions

The proposed transformation for the time series of the stock market transactions, of a fractal type and for a given trade symbol, for example “**ORCL**”, which conducts to an equivalent and limited family of Markov chains. This geometric transformation connects the sub-elements of transactions realizations into a relational model, for all sub-elements which are included into two adjacent windows of a constant duration.

	w_k						w_l					
Δn												
Δj												
$\Delta 3$												
$\Delta 2$												
$\Delta 1$												
	a_1	b_1	c_1	...	i_1	...	n_1	a_2	b_2	c_2	...	i_2 ... n_2

Fig. 4 Two adjacent windows w_k si w_l of a constant duration

The diagram of the contextual domain defined by two adjacent windows which support the transformation is shown below.

The first step of the transformation for the time series is the quantification [19] of the contextual domain for transactions realizations Fig.4.

The time quantification of the contextual domain $w_k + w_l$ is done by the pair of series $a_1, b_1, c_1, \dots, i_1, \dots, n_1$ for window w_k and $a_2, b_2, c_2, \dots, i_2, \dots, n_2$ for window w_l . These micro-samples, the elements of the hierarchical tree structure as in Fig.2, are of equal duration and observe the real sequential publishing order of the transactions.

This much, the duration size for the pair of windows and for the micro-samples are constants during the complete processing period. Different values for these constants can be used by distinct versions of independent processing sessions, in parallel processing.

The published price is unfolding between the two local limits for the union of each two windows $w_k + w_l$, low share price and high share price Fig.3.

The direct consequence of this decision is that the dynamic model has a variable scale for the equi-levels of prices, Δj Fig.4.

The calculation of the price level is done after the formula of the ponderate average for the transactions prices which depends on each individual transaction volume.

$$\text{The ponderate average transaction price } a = \frac{\sum(p_i * v_i)}{\sum(v_i)}, \quad (1)$$

where p_i is the trading price and v_i the transaction volume for the generic transaction t_i with $t_i \in \Delta j$.

It is choosed, for simplification, equal number of quantification steps for the two orthogonal dimensions. In other words the number of elements for the direction of time series, the horizontal component of the window, is equal with the number of price levels for the transactions after the quantification, the common vertical component for the two adjacent windows.

There is now a practical possibility to compare the transactions price levels of equivalent coordinates, after the quantification vertical result, for each pair of elements which belong to those two adjacent windows. The solution is equivalent with the comparison of two equal square matrices, hence it is building the dynamical relational model of the analyzed transaction. Those comparing conditions, used by JESS application, are:

$$c_{ij}^{w,w} = \begin{cases} 1, & \text{if the elements } e_{ij}^{w_1} \text{ and } e_{ij}^{w_2} \text{ have in accordance realizations for each of the sub-windows } u_{ij}^{w_1} \text{ and } u_{ij}^{w_2}; \\ 0, & \text{for all the others situations, namely when the realizations of the elements } e_{ij}^{w_1} \text{ and } e_{ij}^{w_2} \text{ are} \\ & \text{missing at least for one of the sub-windows } u_{ij}^{w_1} \text{ or } u_{ij}^{w_2}. \end{cases} \quad (2)$$

where e_{ij} is the generic element of equivalent coordinates and belonging to the adjacent windows; and u_{ij} is the generic sub-window.

The equivalent and limited family of Markov chains, of n^2 cardinality and with the states “0” and “1”, represents the formalization of the same transaction process and in this way it is still the continuation of the original idea about the transformation of time series.

The family of Markov chains is thus building the dynamic model for the stock market transactions and together with the expert module is generating the searched predictive answer.

The guarantee of the system quality answer is provided by the confidence factor attached to the result.

6. The Developing of the Expert Module with JESS Technology

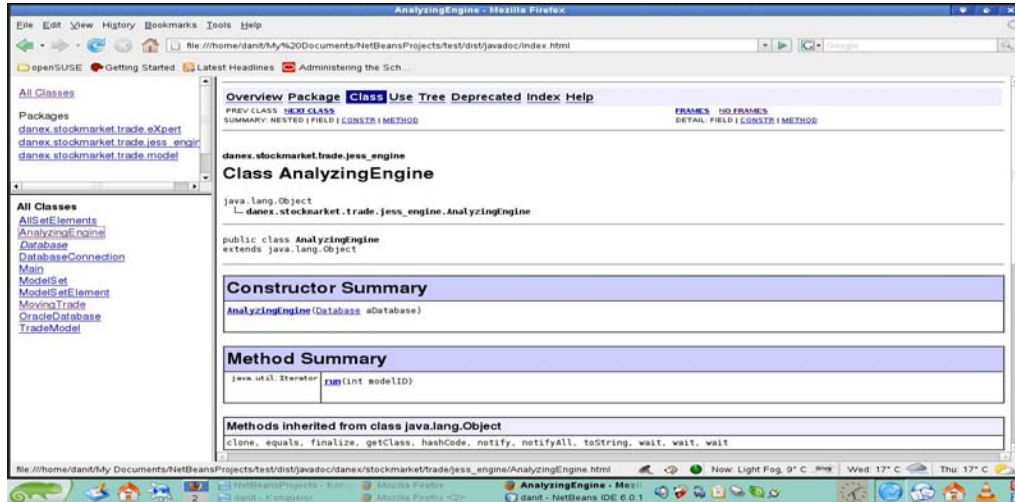


Fig. 5 The Documentation for Analyzing Engine class generated by NetBeans after the source code

The Expert Module first of all is playing the technical role as a double common interface between Oracle database and Rete algorithm.

The source code for the initialization of the Rete algorithm interface is fetched from Oracle database at the run time as it can be seen from the below JAVA source code example:

```
package danex.stockmarket.trade.jess_engine;

import danex.stockmarket.trade.*;
import danex.stockmarket.trade.model.MovingTrade;
import danex.stockmarket.trade.model.ModelSet;
import danex.stockmarket.trade.eXpert.DatabaseConnection;
import jess.*;

import java.util.Iterator;
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public AnalyzingEngine(Database aDatabase) throws JessException,
Exception {
// Create a Jess rule engine
```



```
engine = new Rete();
engine.reset();
String selSQL = "SELECT statement FROM business_rules WHERE
modul_name = 'JESS' order by rule_id";
try {
    // Connect to the database
    DatabaseConnection currConnection = new DatabaseConnection();
    conn = currConnection.openConnection();
    stmt4jess = conn.createStatement();
    rSetJessModel = stmt4jess.executeQuery(selSQL);
} catch (SQLException s) {
    System.out.println(s.getMessage());
    s.printStackTrace();
}
while (rSetJessModel.next()) {
    String stmt4JESS = rSetJessModel.getString(1);
    engine.eval(stmt4JESS);
}
engine.eval("watch compilations");
engine.eval("view");
rSetJessModel.close();
conn.close();
database = aDatabase;
engine.addAll(database.getAllTradeItems());
marker = engine.mark();
}
```

This approach has been selected to make the developing of the Expert Module easier by means of a specialized module, implemented by Oracle database. The “Business Rules” module is in fact administrating the knowledge database for the Expert Engine powered by JESS application (and is equivalent with the *.clp file).

The Expert Module has two separate types of inputs: “The Facts” which are dynamically and similar “predictive answers” with the source in the family of n^2 cardinality Markov chains and “The Knowledge” statements with the source in the ongoing developing of the knowledge acquisition process.

The main role of the Expert Module is thus the completion of the answer building process from a large family of many similar answers, by means of the gained Knowhow. And it has the second role to compute the confidence factor that will be attached to the systems predictive answer.

7. The E-R Model for the Expert Measurement System [20]

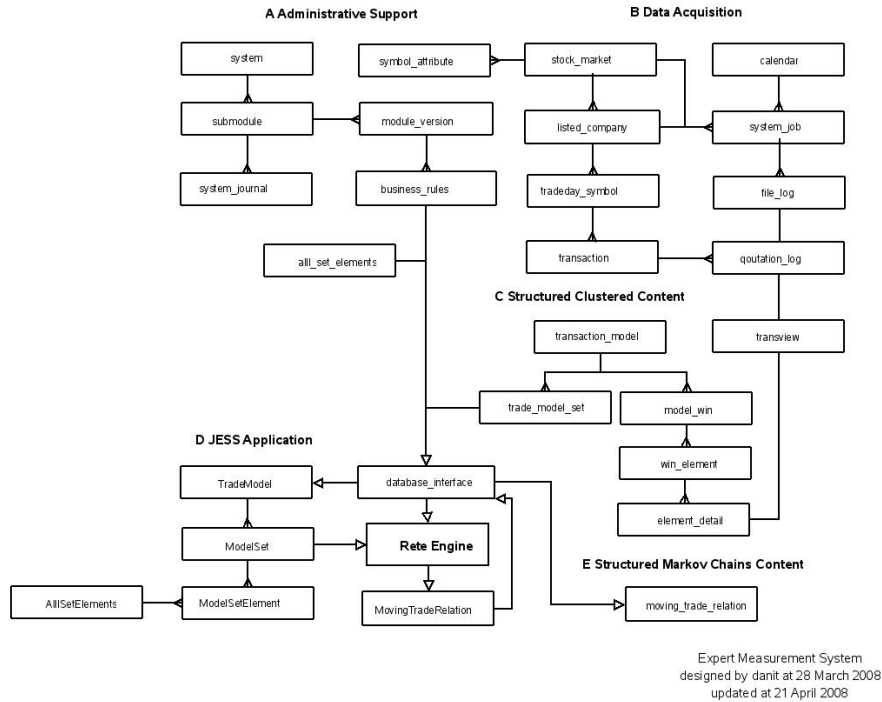


Fig. 6 The E-R Diagram of the Oracle database for the Expert Measurement System

There are five areas for the E-R Diagram, as it can be seen from Fig. 6, enumerated below:

- A) Administrative Support;
- B) Data Acquisition, input data recorded as a native type;
- C) Structured Clustered Content, transformed data of fractal type;
- D) Jess Application;
- E) Structured Markov Chains Content, transformed data as a Markov family.

The information system specification is mapped to the logic data model. Thus, the model in turn is mapped to the physical database infrastructure by means of the source schema script and with the help of developing tools.

Table 1

Database tables which are directly mapped to JESS classes		
no	Table	Class
1	Trade_Model_Set	ModelSet
2	All_Set_Elements	AllSetElements
3	Moving_Trade_Relation	MovingTradeRelation

8. Reporting some Aspects from the Data Processing

M_WIN_ELEMENT

ELEMENT	Wind	Elements	Prev Close	Open	Avg Share Price	Share Volume	Low Share Price	High Share Price	Close	Trade time
10636	7506	1	19.93	19.93	19.9305	58943	19.93	19.94	19.94	08-03-19 12:30
10637	7506	2	19.94	19.94	19.9493	56098	19.94	19.95	19.95	08-03-19 12:31
10638	7506	3	19.95	19.95	19.9577	69198	19.95	19.96	19.96	08-03-19 12:32
10639	7506	4	19.96	19.95	19.9573	14254	19.95	19.96	19.96	08-03-19 12:33
10640	7506	5	19.96	19.95	19.9507	27972	19.95	19.96	19.96	08-03-19 12:34
10641	7506	6	19.95	19.95	19.9401	78064	19.94	19.95	19.94	08-03-19 12:35
10642	7506	7	19.94	19.93	19.9401	83156	19.93	19.95	19.9489	08-03-19 12:36
10643	7506	8	19.9489	19.9482	19.9403	70207	19.93	19.9482	19.93	08-03-19 12:37
10644	7506	9	19.93	19.94	19.9321	66891	19.92	19.94	19.93	08-03-19 12:38
10645	7506	10	19.93	19.94	19.9231	62571	19.92	19.94	19.92	08-03-19 12:39

rows 1 - 10 of more than 500

Done

Fig. 7 The Report of the Win Element records from the hierarchical transaction structure

TRADE_MODEL_SET

Instance Id	ModelId	Symbol	RECORD_SEQ_DISPLAY	Elements	Element Val
2	266	ORCL	41	1	A1
2	266	ORCL	42	2	B3
2	266	ORCL	43	3	C5
2	266	ORCL	44	4	D5
2	266	ORCL	45	5	E5
2	266	ORCL	46	6	F6
2	267	ORCL	47	1	A3
2	267	ORCL	48	2	B5
2	267	ORCL	49	3	C4
2	267	ORCL	50	4	D4

rows 41 - 50 of 124

Done

Fig. 8 The Report of the equivalent Markov family chains which build dynamically the model from the stock market transactions values

9. Conclusions

The main original contribution of the research, as it has been shown in this paper, is the geometric transformation of the the stock market transactions, of a fractal type, to a formal and dynamic model, which is using a family of Markov chains to make a guaranteed prediction within a confidence factor.

What I consider truly remarkably in the research of the time series of the Stock Market Transactions are these two aspects:

1. the practical possibility to build a model of “the real thing”;
2. to make a good prediction upon it.

The transformation of the time series for the stock market transactions to the Markov chain formality, as it have been demonstrated here, gives in fact the chance to have the same approach for as many topics as one can think of.

I am intending to tackle the challenge of some others applications from different areas by means of an updated and improved “Expert Measurement System” in the further research work.

REFERENCES

- [1] System Architecture, http://en.wikipedia.org/wiki/System_architecture
- [2] Expert System, http://en.wikipedia.org/wiki/Expert_System
- [3] Measurement System, <http://en.wikipedia.org/wiki/Measurement>
- [4] Dynamic Modeling, http://en.wikipedia.org/wiki/Mathematical_models
- [5] Stock Market, http://en.wikipedia.org/wiki/Stock_Market
- [6] Stock Market Transactions, http://en.wikipedia.org/wiki/Transaction_cost
- [7] Fractal Geometry, <http://en.wikipedia.org/wiki/Fractal>
- [8] Markov chains, http://en.wikipedia.org/wiki/Markov_chain
- [9] NASDAQ, <http://en.wikipedia.org/wiki/NASDAQ>
- [10] Python, <http://www.python.org/>
- [11] JESS Technology, <http://www.jessrules.com/>
- [12] Rete Algorithm, http://en.wikipedia.org/wiki/Rete_algorithm
- [13] Time Series, http://en.wikipedia.org/wiki/Time_Series
- [14] Finance and Risk can be accessed at the address <http://classes.yale.edu/Fractals/Panorama/>
- [15] Benoit B. Mandelbrot, A Multifractal Walk Down Wall Street, Scientific American February 1999
- [16] Benoit B. Mandelbrot, http://en.wikipedia.org/wiki/Benoît_Mandelbrot
- [17] Fractal, <http://en.wikipedia.org/wiki/Fractal>
- [18] Cluster, http://en.wikipedia.org/wiki/Data_clustering
- [19] Quantification, <http://en.wikipedia.org/wiki/Quantification>
- [20] The E-R Diagram, Entity-relationship model, http://en.wikipedia.org/wiki/Entity-relationship_model