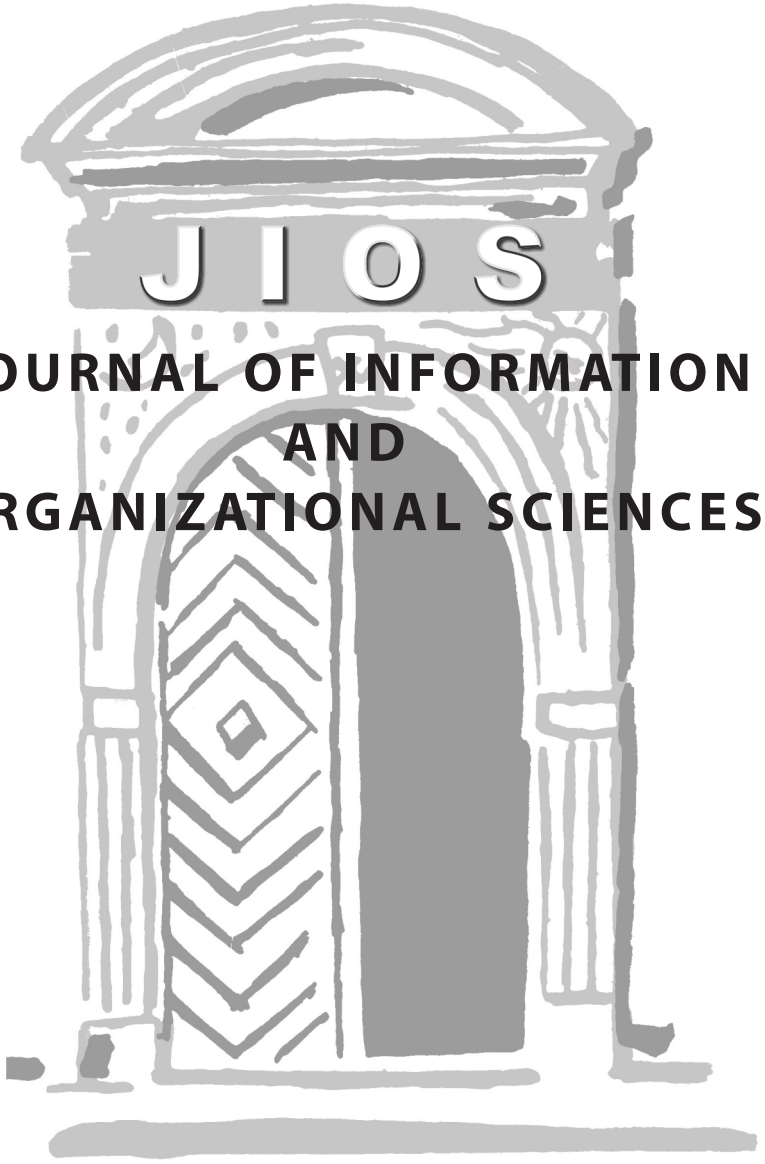
UNIVERSITY OF ZAGREB
FACULTY OF ORGANIZATION AND INFORMATICS

UDC: 007
65.01

ISSN: 0351-804



JOURNAL OF INFORMATION AND ORGANIZATIONAL SCIENCES

2

JIOS VOL. 29 NO. 2 PP. 1 - 95 VARAŽDIN 2005

JIOS

Journal of Information and Organizational Sciences

UDC: 007
65.01
ISSN: 0351-1804

Publisher
Faculty of Organization and Informatics
Varaždin
42 000 Varaždin, Pavlinska 2, Croatia

Subscription per volume: 100 kunas
To subscribe, please contact the publisher

JIOS is published twice a year

JIOS is abstracted (indexed) in:
INSPEC, Referativnyj žurnal (Informatika), Zentralblatt für Mathematik/Mathematics Abstracts, and Mathematical Reviews

A Word from the Editors

All the papers in this issue of the JIOS Journal may fit into the *Science as Usual* category and are therefore not going to be dealt with in detail in the introduction. However, for one of the papers included an exception has been provided, namely the paper *Sigma-notation and the equivalence of P and NP classes* by Miron Ivanovich Telpiz. The circumstances leading to its publication in this issue of JIOS are presented by the editors in this preface.

It is generally acknowledged that, ever since it was defined by S. Cook over thirty years ago, P = NP has presented one of the major open problems in computing science and information science in general. Whereas, on one hand, this problem presents a formidable challenge to any scientist dealing with it, its solution, especially in case it should turn out to be a positive one − considering that current technology tends to evolve in a practical direction and its overall functioning is conducted implicitly under the assumption of a negative solution − would have a major impact on the development of information technology as well as the wider context of human society as a whole. Apart from being put forward in conferences, both positive and negative "solutions" to P = NP problem can be occasionally found in press, scientific and professional journals. The web page titled "P-versus-NP Page" (see http://www.win.tue.nl/~gwoegi/P-versus-NP.htm) provides an excellent source for this issue. It contains links to papers stating that P = NP, as well as those contradicting the opposite, i.e., stating that P ≠ NP. Furthermore, references are given to Oded Goldreich's rationale behind his decision not to review papers proposing solutions to P = NP problem and related *difficult* problems because, in his own words, they "*also attract the attention of non-experts, and one annoying consequence is a flood of false claims of resolutions of these problems*" (see http://www.wisdom.weizmann.ac.il/~oded/faq.html). Dr. Goldreich − a recognised figure among mathematicians of our time – recommends other scientists to follow his example when this issue is concerned. Nevertheless, his arguments (at least those regarding non-experts) are definitely not to be acknowledged in the case of the work of the Russian mathematician and logician Miron Ivanovich Telpiz, (see http://www.tarusa.ru/~mit/ENG/eng.php for details), who has been developing his positionality theory for functions, with a special emphasis on Boolean functions, for over two decades. While doing so, he has been drawing its (primarily conceptual) analogy to positionality theory for number systems. Building on his extensive theory, which is combinatory very complex one indeed, Dr. Telpiz states the proof of P = NP. The underlying theory itself was proposed in the first volume of his book *Positionality principle for notation and calculation the functions,* originally published in Russian by the *Institute of Space Researches of the Russian Academy of Science.* It was due to this book that contacts between a member of the Editorial Board and Dr. Telpiz, as well as Dr. Andrej Fomin, his assistant and associate, were established. During those contacts the idea evolved of Dr. Tepiz delivering an invited lecture at the IIS2004 Conference. Annual IIS Conferences have been organized by the Faculty of Organization and Informatics for years. A lecture on proving the statement P = NP was arranged. Unfortunately, Dr. Telpiz was eventually not capable of giving the lecture. Nevertheless, he proposed for his lecture to be published in form of an article in the JIOS Journal. Upon receiving the Russian version of his article named *Sigma-notation and the equivalence of P and NP classes*, the Editorial Board had it translated into English and proceeded to having it reviewed. The paper was sent to several renowned world experts in the field of Boolean functions, mathematical logic and algorithm theory for review. Yet, no response was received from reviewers contacted although the Editorial Board and Dr. Telpiz made efforts

for access to all relevant papers, as well as the book itself, to be possible. Bearing in mind not only the great relevance of P = NP problem but also the constant need for scientific ideas and findings to be disseminated, the JIOS Editorial Board has decided to have Dr. Telpiz's paper published although it had not been officially reviewed, which the author himself has consented to in writing. Owing to its inclusion in this issue, the paper is being presented to a wider scientific and professional audience who are thus given an opportunity to both consider and assess it from their own perspective.

Editors

CONTENTS

UDC: 510.6

SIGMA-NOTATION AND THE EQUIVALENCE OF P AND NP CLASSES

Miron.I.Telpiz, Tarusa, Russia
mit@tarusa.ru

Abstract: *The aim of this paper is to show that σ–notation, based on the positionality principle, doesn't just answer the question of equivalence of P and NP problem classes, but also represents the key to the solution of (some) more general problems from the domain of multivalued logics. However, the achievement of such an aim in its full scope is hardly possible within the limits of this paper. Therefore, the following plan shall be realized:*

Firstly, the logic algebra fragment necessary for the solution of the P and NP classes equivalence problem shall be expounded.

Secondly, the necessary symbols and definitions shall be introduced to show that the calculations in the domain of the multivalued logic can be directly executed within the framework of σ–notation.

Keywords: *σ-notation, P=NP problem, SAT problem, multivalued logics.*

1. INTRODUCTION

As it is known from [1], to prove the equivalence of the classes of problems mentioned above, it would suffice that some (any) of the NP–complete problems can be solved in time bounded by the polynomial of the dimension of the problem. Here we shall consider the satisfiability problem in its classical formulation [1], which requires the answer to the following question: is there a sequence of variable values, which makes a given statement in CNF true? If the answer is positive, the statement in CNF is called satisfiable, if it is negative — unsatisfiable (or contradictory); if, apart from the recognition of satisfiability, the showing of the associated sequence of the propositional variable values is also required, the problem is called a SAT problem. This particular problem is going to be considered in the course of further explanation.

In 1971 S. Cook proved in [2] the SAT problem being NP–complete. In that way, proving that some (any) of the SAT class problems can be solved in time bounded by a polynomial of the dimension of the problem became sufficient for the proof of the equivalence of P and NP problem classes.

The SAT problem shall be solved by continous realization of conjuctions over disjuncts with the help of some special notation ([3]), whose necessary elements are described below.

2. SOME SYMBOLS AND DEFINITIONS

Beside the binary number system Zh we shall use quaternary Vh, octal Ah and hexadecimal Sh number system. In order to avoid repeated stating of the number system the digits are written in, the symbol θ shall be used to display a binary zero, a $\bar{\theta}$ to display a binary one. We are also introducing the following symbols

$$\begin{aligned}
Zh &\rightleftharpoons \{\theta, \bar{\theta}\},\\
Vh &\rightleftharpoons \{\nu, \tau, \bar{\tau}, \bar{\nu}\} \rightleftharpoons \{\theta\,\theta, \bar{\theta}\,\theta, \theta\,\bar{\theta}, \bar{\theta}\,\bar{\theta}\},\\
Ah &\rightleftharpoons \{\omega, \downarrow, \oplus, /, \overline{/}, \bar{\oplus}, \overline{\downarrow}, \bar{\omega}\} \rightleftharpoons \{\theta\,\nu, \bar{\theta}\,\nu, \theta\,\tau, \bar{\theta}\,\tau, \theta\,\bar{\tau}, \bar{\theta}\,\bar{\tau}, \theta\,\bar{\nu}, \bar{\theta}\,\bar{\nu}\},\\
Sh &\rightleftharpoons \{0, 1, 2, 3, 4, 5, 6, 7, \bar{7}, \bar{6}, \bar{5}, \bar{4}, \bar{3}, \bar{2}, \bar{1}, \bar{0}\} \rightleftharpoons\\
&\rightleftharpoons \{\theta\omega, \bar{\theta}\omega, \theta\downarrow, \bar{\theta}\downarrow, \theta\oplus, \bar{\theta}\oplus, \theta/, \bar{\theta}/, \theta\overline{/}, \bar{\theta}\overline{/}, \theta\bar{\oplus}, \bar{\theta}\bar{\oplus}, \theta\overline{\downarrow}, \bar{\theta}\overline{\downarrow}, \theta\bar{\omega}, \bar{\theta}\bar{\omega}\},
\end{aligned}$$

respecting the convention that a pair of literals–digits in the second pair of curly brackets defines the corresponding literal-digit in the first pair of curly brackets.

Literals-digits from Zh, Vh and Sh shall be used for writing binary vectors, which shall be written in double angle brackets, a notation implicitly including the concatenation of literals-digits. For example, according to these principles, the following holds: $\langle\langle\bar{\theta}\,\theta\,\theta\,\theta\,\theta\,\bar{\theta}\,\bar{\theta}\,\theta\rangle\rangle = \langle\langle\tau\,\nu\,\bar{\tau}\,\tau\rangle\rangle = \langle\langle 16\rangle\rangle$.

Double angle brackets are used for writing the vectors of the length 2^n (that is, of rank n), which are called finite σ–operators. At this the following convention shall be respected: if $\langle\langle\alpha\rangle\rangle$— is some vector of rank k (length 2^k), then the notation $\langle\langle\alpha_j\rangle\rangle$ denotes that each coordinate of the vector

$\langle\langle\alpha\rangle\rangle$ appears j times, and therefore vector $\langle\langle\alpha_j\rangle\rangle$ has the rank $j+k$. The notation $\langle\langle\alpha^i\rangle\rangle$ denotes that the vector $\langle\langle\alpha\rangle\rangle$ appears 2^i times in concatenation. In that way, the notation $\langle\langle\alpha^i_j\rangle\rangle$ — denotes a vector of the rank $k+j+i$, under the condition that $\langle\langle\alpha\rangle\rangle$ — is a vector of rank k.

We would like to draw attention to the fact that the double angle brackets point to the coordinates appearing in the exponential ammount, and also to the fact that the upper, as well as the lower indices of a vector (or/and its coordinates) point to the exponential character of the appearance, with the indices implied (if not given, they equal zero).

Literals-digits from Ah are used for denoting the s–operator (see [3], pp. 25 and 26), but shall here be used only as $\overline{/}$ and $\overline{\downarrow}$ for denoting the bitwise conjuntion and disjunction respectively.

In order to give the SAT problem a table form, we shall write disjuncts in the rows of a table, so that the number of columns would equal the number of variables, and the number of rows the number of disjuncts. The table shall be filled in the following way: The literal with the smallest index i in the given disjunct we shall associate with $\bar{\tau}_{i-1}$ if it equals x_i, and τ_{i-1} if it equals $\bar{x}_i$. This we shall enter in the $(i-1)$–th column. In the other columns, which appear on the right side of the $(i-1)$–th column we shall enter T in the j–th column, if x_{j+1} (T^* for the $\bar{x}_{j+1}$) appears in the disjunct.

The explanation of this form of the SAT problem shall be given later, at the begin of paragraph 3.

In connection with the tables we shall use the following terms:*the length of the row* and *rank of the row.* The length of the row in the table with n columns $0, 1, 2, \ldots, n-1$ and the beginning τ_{i-1} or $\bar{\tau}_{i-1}$ equals $n-i+1$, while the rank equals $n-i-k+1$, where k — is the sum of M productions (which are implied) in the row, i.e. the rank of the row equals the number of the literals appearing in the disjunct.

By permuting rows and columns of the table, a standard shape of the table is acheived, meaning that the columns are aligned from the left to the right, in order of nondecreasing values of the expression $E = E_T * E_{T^*}$, where E_T i E_{T^*} — represents the corresponding number of appearances of T (including τ) in each column. The rows in the table are aligned according to nondecreasing values of their length, the same condition applying to the blocks (and their subblocks) of the subtable with the rows of the same length, which means that the rows in the block (subblock) with the beginning in the k–th row are aligned according to nondecreasing of their length, with the exception of the k–th column.

3. NOTATIONS FOR THE SAT PROBLEM

The left part of the table 3 can serve as an example of stating the SAT problem by using generating rules from the set

$$Q_3 = \{T, T^*, M\},$$

because the effect of every generating rule from the set Q_3 on the given vector $\langle\alpha_j\rangle$, by definition transforms the same vector in the concatenation of two vectors, the result being the double value of the dimension of the initial vector.

$$\langle\alpha_j\, T\rangle = \langle\alpha_j\, \bar{\theta}_j\rangle, \quad \langle\alpha_j\, T^*\rangle = \langle\bar{\theta}_j\, \alpha_j\rangle, \quad \langle\alpha_j\, M\rangle = \langle\alpha_j\, \alpha_j\rangle. \tag{1}$$

Definitions (1) correspond to known definitions (see [3], p. 87).

If the values of logic algebra functions $f(X_j)$ on the variable row $X_j = (x_1, x_2, \ldots, x_j)$, can be represented by the vector $\langle\alpha_j\rangle$, that shall be represented in the following way: $f(X_j) \backsimeq \langle\alpha_j\rangle$.

Theorem 1 If the logic algebra function $f(X_j)$ is represented by the vector $\langle\alpha_j\rangle$, then the following functions, with the domain X_{j+1}, can be represented as following:

$$f(X_{j+1}) = f(X_j) \backsimeq \langle\alpha_j M\rangle, \quad f(X_j)\bar{\downarrow}x_{j+1} \backsimeq \langle\alpha_j T\rangle, \quad f(X_j)\bar{\downarrow}\bar{x}_{j+1} \backsimeq \langle\alpha_j T^*\rangle.$$

This theorem is a fragment of the theorem 14 (see [3], p. 90).

The creation of the SAT problem table in the production system Q_3 can now be considered explained and justified.

The application of the theorem 1 and equivalence (1) on each row of the SAT problem in the system Q_3 makes it possible to notate every disjunction with an σ–operator in the system Vh, which is shown in the subtable of the same table 3.

The translation of the σ–operator table from the Vh system in the Sh system is realized in a relatively simple manner. It should suffice to keep in mind that for the integers $i \geq 0$ and $j \geq 0$ the following equivalences apply:

$$\begin{array}{llll} \langle\langle\tau^i_{j+1}\rangle\rangle = \langle\langle 3^i_j\rangle\rangle, & \langle\langle\tau^{i+1}\rangle\rangle = \langle\langle 5^i\rangle\rangle, & \langle\langle\nu_{j+1}\rangle\rangle = \langle\langle 0_j\rangle\rangle, \\ \langle\langle\bar{\tau}^i_{j+1}\rangle\rangle = \langle\langle\bar{3}^i_j\rangle\rangle, & \langle\langle\bar{\tau}^{i+1}\rangle\rangle = \langle\langle\bar{5}^i\rangle\rangle, & \langle\langle\bar{\nu}_{j+1}\rangle\rangle = \langle\langle\bar{0}_j\rangle\rangle. \end{array} \tag{2}$$

If next to the equivalence (2) the following equivalences are considered

$$\begin{array}{ll} \langle\bar{\tau}_j\, T\rangle = \langle\langle\bar{1}_j\rangle\rangle, & \langle\tau_j\, T\rangle = \langle\langle\bar{2}_j\rangle\rangle, \\ \langle\bar{\tau}_j\, T^*\rangle = \langle\langle\bar{4}_j\rangle\rangle, & \langle\tau_j\, T^*\rangle = \langle\langle 7_j\rangle\rangle, \end{array}$$

then it is possible to immediately move from the table in the Q_3 production system to the table in the Sh production system. In whichever way this is carried out, the SAT problem table in the Sh system shall be represented by the subtable of the table 3.

4. CALCULUS FOR THE SAT PROBLEM

In [3] there is a detailed consideration of the logic calculus in the system Vh. To avoid repeating ourselves, here we shall concentrate exclusively on the calculus in the Sh system for the SAT problem. That means that we have to concentrate on the σ–operators in the Sh system.

At first it should be noticed that the table 1, which realizes the $\bar{/}$ conjuction over the pair of vectors (literals-digits) α and β from Sh (in the table 1 the angle brackets around literals-digits are left out) holds.

The table 1 is used while applying the following obvious lemma

Lemma 1 If $\alpha,\ \beta \in Sh$ and $\langle\langle\alpha\rangle\rangle\,\bar{/}\,\langle\langle\beta\rangle\rangle = \langle\langle\gamma\rangle\rangle$, then
$\langle\langle\alpha^j\rangle\rangle\,\bar{/}\,\langle\langle\beta^j\rangle\rangle = \ = \langle\langle\gamma^j\rangle\rangle, \quad \langle\langle\alpha_j\rangle\rangle\,\bar{/}\,\langle\langle\beta_j\rangle\rangle = \langle\langle\gamma_j\rangle\rangle \quad$ for $j = 1, 2, 3, \ldots$.

Table 1

$\bar{/}$	$\bar{1}$	$\bar{2}$	$\bar{3}$	$\bar{4}$	$\bar{5}$	$\bar{6}$	$\bar{7}$	7	6	5	4	3	2	1			
1	0	1	0	1	0	1	0	1	0	1	0	1	0	1		$\bar{7}$	$\bar{7}$
2	2	0	0	2	2	0	0	2	2	0	0	2	2		$\bar{6}$	$\bar{7}$	$\bar{6}$
3	2	1	0	3	2	1	0	3	2	1	0	3		$\bar{5}$	$\bar{7}$	$\bar{7}$	$\bar{5}$
4	4	4	4	0	0	0	0	4	4	4	4		$\bar{4}$	$\bar{5}$	$\bar{6}$	$\bar{7}$	$\bar{4}$
5	4	5	4	1	0	1	0	5	4	5		$\bar{3}$	$\bar{7}$	$\bar{7}$	$\bar{7}$	$\bar{7}$	$\bar{3}$
6	6	4	4	2	2	0	0	6	6		$\bar{2}$	$\bar{3}$	$\bar{6}$	$\bar{7}$	$\bar{6}$	$\bar{7}$	$\bar{2}$
7	6	5	4	3	2	1	0	7		$\bar{1}$	$\bar{3}$	$\bar{3}$	$\bar{5}$	$\bar{5}$	$\bar{7}$	$\bar{7}$	$\bar{1}$
										$\bar{1}$	$\bar{2}$	$\bar{3}$	$\bar{4}$	$\bar{5}$	$\bar{6}$	$\bar{7}$	$\bar{/}$

In the lemmas which are going to be presented in further exposition, it is important to keep in mind (due to the use of the convention stated in paragraph 2) that the following equivalences (where $j \geq 0$ is an integer) always hold:

$$\begin{array}{llll} \langle\langle 0_{j+1}\rangle\rangle = \langle\langle 0_j 0_j\rangle\rangle, & \langle\langle 1_{j+1}\rangle\rangle = \langle\langle 3_j 0_j\rangle\rangle, & \langle\langle 2_{j+1}\rangle\rangle = \langle\langle\bar{3}_j 0_j\rangle\rangle, & \langle\langle 3_{j+1}\rangle\rangle = \langle\langle\bar{0}_j 0_j\rangle\rangle, \\ \langle\langle 4_{j+1}\rangle\rangle = \langle\langle 0_j 3_j\rangle\rangle, & \langle\langle 5_{j+1}\rangle\rangle = \langle\langle 3^1_j\rangle\rangle, & \langle\langle 6_{j+1}\rangle\rangle = \langle\langle\bar{3}_j 3_j\rangle\rangle, & \langle\langle 7_{j+1}\rangle\rangle = \langle\langle\bar{0}_j 3_j\rangle\rangle, \\ \langle\langle\bar{7}_{j+1}\rangle\rangle = \langle\langle 0_j\bar{3}_j\rangle\rangle, & \langle\langle\bar{6}_{j+1}\rangle\rangle = \langle\langle 3_j\bar{3}_j\rangle\rangle, & \langle\langle\bar{5}_{j+1}\rangle\rangle = \langle\langle\bar{3}^1_j\rangle\rangle, & \langle\langle\bar{4}_{j+1}\rangle\rangle = \langle\langle\bar{0}_j\bar{3}_j\rangle\rangle, \\ \langle\langle\bar{3}_{j+1}\rangle\rangle = \langle\langle 0_j\bar{0}_j\rangle\rangle, & \langle\langle\bar{2}_{j+1}\rangle\rangle = \langle\langle 3_j\bar{0}_j\rangle\rangle, & \langle\langle\bar{1}_{j+1}\rangle\rangle = \langle\langle\bar{3}_j\bar{0}_j\rangle\rangle, & \langle\langle\bar{0}_{j+1}\rangle\rangle = \langle\langle\bar{0}_j\bar{0}_j\rangle\rangle. \end{array}$$

Lemma 2 If $\varepsilon \in Sh$, then

$$\langle\langle 3_{j+1}\rangle\rangle\,\bar{/}\,\langle\langle\varepsilon^{j+1}\rangle\rangle = \langle\langle\varepsilon^j\, 0_j\rangle\rangle, \quad \langle\langle\bar{3}_{j+1}\rangle\rangle\,\bar{/}\,\langle\langle\varepsilon^{j+1}\rangle\rangle = \langle\langle 0_j\,\varepsilon^j\rangle\rangle.$$

Table 2

$\bar{/}$	1^1	2^1	3^1	4^1	5^1	6^1	7^1	$\bar{7}^1$	$\bar{6}^1$	$\bar{5}^1$	$\bar{4}^1$	$\bar{3}^1$	$\bar{2}^1$	$\bar{1}^1$
1_1	10	20	1_1	0_1	10	20	1_1	0_1	10	20	1_1	0_1	10	20
2_1	0_1	0_1	0_1	40	40	40	40	70	70	70	70	2_1	2_1	2_1
3_1	10	20	1_1	40	50	60	70	70	60	50	40	2_1	20	10
4_1	01	02	4_1	0_1	01	02	4_1	0_1	01	02	4_1	0_1	01	02
5_1	1^1	2^1	3^1	0_1	1^1	2^1	3^1	0_1	1^1	2^1	3^1	0_1	1^1	2^1
6_1	01	02	4_1	40	41	42	43	70	71	72	73	2_1	31	32
7_1	1^1	2^1	3^1	40	51	62	73	70	61	52	43	2_1	21	12
$\bar{7}_1$	0_1	0_1	0_1	04	04	04	04	$0\bar{7}$	$0\bar{7}$	$0\bar{7}$	$0\bar{7}$	$\bar{7}_1$	$\bar{7}_1$	$\bar{7}_1$
$\bar{6}_1$	10	20	1_1	04	14	24	34	$0\bar{7}$	$1\bar{7}$	$2\bar{7}$	$3\bar{7}$	$\bar{7}_1$	$1\bar{3}$	$2\bar{3}$
$\bar{5}_1$	0_1	0_1	0_1	4^1	4^1	4^1	4^1	$\bar{7}^1$	$\bar{7}^1$	$\bar{7}^1$	$\bar{7}^1$	$\bar{3}^1$	$\bar{3}^1$	$\bar{3}^1$
$\bar{4}_1$	10	20	1_1	4^1	54	64	74	$\bar{7}^1$	$\bar{6}\bar{7}$	$\bar{5}\bar{7}$	$\bar{4}\bar{7}$	$\bar{3}^1$	$\bar{2}\bar{3}$	$\bar{1}\bar{3}$
$\bar{3}_1$	01	02	4_1	04	05	06	07	$0\bar{7}$	$0\bar{6}$	$0\bar{5}$	$0\bar{4}$	$\bar{7}_1$	$0\bar{2}$	$0\bar{1}$
$\bar{2}_1$	1^1	2^1	3^1	04	15	26	37	$0\bar{7}$	$1\bar{6}$	$2\bar{5}$	$3\bar{4}$	$\bar{7}_1$	$1\bar{2}$	$2\bar{1}$
$\bar{1}_1$	01	02	4_1	4^1	45	46	47	$\bar{7}^1$	$\bar{7}\bar{6}$	$\bar{7}\bar{5}$	$\bar{7}\bar{4}$	$\bar{3}^1$	$\bar{3}\bar{2}$	$\bar{3}\bar{1}$

Lemma 3 If $\varepsilon \in Sh$ and $\langle\langle 3\rangle\rangle \,\bar{/}\, \langle\langle\varepsilon\rangle\rangle = \langle\langle\eta\rangle\rangle$, $\langle\langle\bar{3}\rangle\rangle \,\bar{/}\, \langle\langle\varepsilon\rangle\rangle = \langle\langle\mu\rangle\rangle$, then

$$\begin{array}{l}
\langle\langle 1_1\rangle\rangle \,\bar{/}\, \langle\langle\varepsilon^1\rangle\rangle = \langle\langle\eta\, 0\rangle\rangle,\ \langle\langle\bar{1}_1\rangle\rangle \,\bar{/}\, \langle\langle\varepsilon^1\rangle\rangle = \langle\langle\mu\,\varepsilon\rangle\rangle,\\
\langle\langle 2_1\rangle\rangle \,\bar{/}\, \langle\langle\varepsilon^1\rangle\rangle = \langle\langle\mu\, 0\rangle\rangle,\ \langle\langle\bar{2}_1\rangle\rangle \,\bar{/}\, \langle\langle\varepsilon^1\rangle\rangle = \langle\langle\eta\,\varepsilon\rangle\rangle,\\
\langle\langle 4_1\rangle\rangle \,\bar{/}\, \langle\langle\varepsilon^1\rangle\rangle = \langle\langle 0\,\eta\rangle\rangle,\ \langle\langle\bar{4}_1\rangle\rangle \,\bar{/}\, \langle\langle\varepsilon^1\rangle\rangle = \langle\langle\varepsilon\,\mu\rangle\rangle,\\
\langle\langle 5_1\rangle\rangle \,\bar{/}\, \langle\langle\varepsilon^1\rangle\rangle = \langle\langle\eta^1\rangle\rangle,\ \langle\langle\bar{5}_1\rangle\rangle \,\bar{/}\, \langle\langle\varepsilon^1\rangle\rangle = \langle\langle\mu^1\rangle\rangle,\\
\langle\langle 6_1\rangle\rangle \,\bar{/}\, \langle\langle\varepsilon^1\rangle\rangle = \langle\langle\mu\,\eta\rangle\rangle,\ \langle\langle\bar{6}_1\rangle\rangle \,\bar{/}\, \langle\langle\varepsilon^1\rangle\rangle = \langle\langle\eta\,\mu\rangle\rangle,\\
\langle\langle 7_1\rangle\rangle \,\bar{/}\, \langle\langle\varepsilon^1\rangle\rangle = \langle\langle\varepsilon\,\eta\rangle\rangle,\ \langle\langle\bar{7}_1\rangle\rangle \,\bar{/}\, \langle\langle\varepsilon^1\rangle\rangle = \langle\langle 0\,\mu\rangle\rangle.
\end{array}$$

Table 3

$\bar{/}$	0	1	2	3	4	5	σ–oper. in Vh	σ–oper. in Sh	Resulting conjunction
1				$\bar{\tau}$	T^*	T	$\langle\langle\bar{\nu}_3\,\bar{\tau}_3\,\bar{\nu}_4\rangle\rangle$	$\langle\langle\bar{4}_3\,\bar{0}_3\rangle\rangle$	
2				$\bar{\tau}$	T	T^*	$\langle\langle\bar{\nu}_4\,\bar{\tau}_3\,\bar{\nu}_3\rangle\rangle$	$\langle\langle\bar{0}_3\,\bar{1}_3\rangle\rangle$	$\langle\langle\bar{4}_3\,\bar{1}_3\rangle\rangle$
3			$\bar{\tau}$		T	T	$\langle\langle\bar{\tau}_2^1\,\bar{\nu}_3\,\bar{\nu}_4\rangle\rangle$	$\langle\langle\bar{3}_1^1\,\bar{0}_2\,\bar{0}_3\rangle\rangle$	$\langle\langle\bar{3}_1^1\,\bar{3}_2\,\bar{1}_3\rangle\rangle$
4			τ	T^*		T	$\langle\langle(\bar{\nu}_2\,\tau_2)^1\,\bar{\nu}_4\rangle\rangle$	$\langle\langle 7_2^1\,\bar{0}_3\rangle\rangle$	$\langle\langle 2_2\,4_2\,\bar{1}_3\rangle\rangle$
5			$\bar{\tau}$	T		T^*	$\langle\langle\bar{\nu}_4\,(\bar{\tau}_2\,\bar{\nu}_2)^1\rangle\rangle$	$\langle\langle\bar{0}_3\,\bar{1}_2^1\rangle\rangle$	$\langle\langle 2_2\,4_2\,\bar{3}_2\,\bar{1}_2\rangle\rangle$
6			τ	T	T^*		$\langle\langle(\bar{\nu}_3\,\tau_2\,\bar{\nu}_2)^1\rangle\rangle$	$\langle\langle(\bar{0}_2\,\bar{2}_2)^1\rangle\rangle$	$\langle\langle 2_2\,4_2\,\bar{3}_2^1\rangle\rangle$
7			$\bar{\tau}$	T	T	T	$\langle\langle\bar{\tau}_2\,\bar{\nu}_2\,\bar{\nu}_3\,\bar{\nu}_4\rangle\rangle$	$\langle\langle\bar{1}_2\,\bar{0}_2\,\bar{0}_3\rangle\rangle$	$\langle\langle 2_2\,4_2\,\bar{3}_2^1\rangle\rangle$
8		$\bar{\tau}$			T^*	T^*	$\langle\langle\bar{\nu}_4\,\bar{\nu}_3\,\bar{\tau}_1^2\rangle\rangle$	$\langle\langle\bar{0}_3\,\bar{0}_2\,\bar{3}^2\rangle\rangle$	$\langle\langle 2_2\,4_2\,\bar{3}_2\,0_1\,\bar{3}^1\rangle\rangle$
9		$\bar{\tau}$			T^*	T	$\langle\langle\bar{\nu}_3\,\bar{\tau}_1^2\,\bar{\nu}_4\rangle\rangle$	$\langle\langle\bar{0}_2\,\bar{3}^2\,\bar{0}_3\rangle\rangle$	$\langle\langle 2_2\,0_1\,2_1\,\bar{3}_2\,0_1\,\bar{3}^1\rangle\rangle$
10		τ			T^*	T^*	$\langle\langle\bar{\nu}_4\,\bar{\nu}_3\,\bar{\tau}_1^2\rangle\rangle$	$\langle\langle\bar{0}_3\,\bar{0}_2\,3^2\rangle\rangle$	$\langle\langle 2_2\,0_1\,2_1\,\bar{3}_2\,0_2\rangle\rangle$
11		τ		T^*		T	$\langle\langle(\bar{\nu}_2\,\tau_1^1)^1\,\bar{\nu}_4\rangle\rangle$	$\langle\langle(\bar{0}_1\,3^1)^1\,\bar{0}_3\rangle\rangle$	$\langle\langle 2_2\,0_2\,\bar{3}_2\,0_2\rangle\rangle$
12		τ		T	T		$\langle\langle(\tau_1^1\,\bar{\nu}_2\,\bar{\nu}_3)^1\rangle\rangle$	$\langle\langle(3^1\,\bar{0}_1\,\bar{0}_2)^1\rangle\rangle$	$\langle\langle 4_1\,0_1\,0_2\,\bar{3}_2\,0_2\rangle\rangle$
13		τ	T^*			T^*	$\langle\langle\bar{\nu}_4\,(\bar{\nu}_1\,\bar{\tau}_1)^2\rangle\rangle$	$\langle\langle\bar{0}_3\,7_1^2\rangle\rangle$	$\langle\langle 4_1\,0_3\,7_1\,0_2\rangle\rangle$
14		$\bar{\tau}$	T		T		$\langle\langle((\bar{\tau}_1\,\bar{\nu}_1)^1\,\bar{\nu}_3)^1\rangle\rangle$	$\langle\langle(\bar{1}_1^1\,\bar{0}_2)^1\rangle\rangle$	$\langle\langle 4_1\,0_3\,6_1\,0_2\rangle\rangle$
15		$\bar{\tau}$	T^*	T^*			$\langle\langle(\bar{\nu}_2\,\bar{\nu}_1\,\bar{\tau}_1)^2\rangle\rangle$	$\langle\langle(\bar{0}_1\,\bar{4}_1)^2\rangle\rangle$	$\langle\langle 4_1\,0_3\,2_1\,0_2\rangle\rangle$
16	τ			T	T	T	$\langle\langle\tau^2\,\bar{\nu}_2\,\bar{\nu}_3\,\bar{\nu}_4\rangle\rangle$	$\langle\langle 5^1\,\bar{0}_1\,\bar{0}_2\,\bar{0}_3\rangle\rangle$	$\langle\langle 0\,1\,0_3\,2_1\,0_2\rangle\rangle$
17	τ		T^*		T^*	T^*	$\langle\langle\bar{\nu}_4\,\bar{\nu}_3\,(\bar{\nu}_1\,\tau^1)^1\rangle\rangle$	$\langle\langle\bar{0}_3\,\bar{0}_2\,(\bar{0}\,5)^1\rangle\rangle$	$\langle\langle\mathbf{0\ 1\ 0_3\ 2_1\ 0_2}\rangle\rangle$
18		τ	T			T^*	$\langle\langle\bar{\nu}_4\,(\tau_1\,\bar{\nu}_1)^2\rangle\rangle$	$\langle\langle\bar{0}_3\,\bar{2}_1^2\rangle\rangle$	$\langle\langle 0\,1\,0_3\,0_1\,0_2\rangle\rangle$
19	$\bar{\tau}$	T^*			T		$\langle\langle((\bar{\tau}\,\bar{\nu})^2\,\bar{\nu}_3)^1\rangle\rangle$	$\langle\langle(\bar{1}^2\,\bar{0}_2)^1\rangle\rangle$	$\langle\langle\ \mathbf{0}_4\rangle\rangle$

Lemma 4 If $\varepsilon \in Sh$, then

$$\begin{array}{ll}
\langle\langle 1_{j+2}\rangle\rangle \,\bar{/}\, \langle\langle \varepsilon^{j+2}\rangle\rangle = \langle\langle \varepsilon^{j}\, 0_j\, 0_{j+1}\rangle\rangle, & \langle\langle \bar{1}_{j+2}\rangle\rangle \,\bar{/}\, \langle\langle \varepsilon^{j+2}\rangle\rangle = \langle\langle 0_j\, \varepsilon^{j}\, \varepsilon^{j+1}\rangle\rangle, \\
\langle\langle 2_{j+2}\rangle\rangle \,\bar{/}\, \langle\langle \varepsilon^{j+2}\rangle\rangle = \langle\langle 0_j\, \varepsilon^{j}\, 0_{j+1}\rangle\rangle, & \langle\langle \bar{2}_{j+2}\rangle\rangle \,\bar{/}\, \langle\langle \varepsilon^{j+2}\rangle\rangle = \langle\langle \varepsilon^{j}\, 0_j\, \varepsilon^{j+1}\rangle\rangle, \\
\langle\langle 4_{j+2}\rangle\rangle \,\bar{/}\, \langle\langle \varepsilon^{j+2}\rangle\rangle = \langle\langle 0_{j+1}\, \varepsilon^{j}\, 0_j\rangle\rangle, & \langle\langle \bar{4}_{j+2}\rangle\rangle \,\bar{/}\, \langle\langle \varepsilon^{j+2}\rangle\rangle = \langle\langle \varepsilon^{j+1}\, 0_j\, \varepsilon^{j}\rangle\rangle, \\
\langle\langle 5_{j+2}\rangle\rangle \,\bar{/}\, \langle\langle \varepsilon^{j+2}\rangle\rangle = \langle\langle (\varepsilon^{j}\, 0_j)^1\rangle\rangle, & \langle\langle \bar{5}_{j+2}\rangle\rangle \,\bar{/}\, \langle\langle \varepsilon^{j+2}\rangle\rangle = \langle\langle (0_j\, \varepsilon^{j})^1\rangle\rangle, \\
\langle\langle 6_{j+2}\rangle\rangle \,\bar{/}\, \langle\langle \varepsilon^{j+2}\rangle\rangle = \langle\langle 0_j\, \varepsilon^{j+1}\, 0_j\rangle\rangle, & \langle\langle \bar{6}_{j+2}\rangle\rangle \,\bar{/}\, \langle\langle \varepsilon^{j+2}\rangle\rangle = \langle\langle \varepsilon^{j} 0_{j+1}\, \varepsilon^{j}\rangle\rangle, \\
\langle\langle 7_{j+2}\rangle\rangle \,\bar{/}\, \langle\langle \varepsilon^{j+2}\rangle\rangle = \langle\langle \varepsilon^{j+1}\, \varepsilon^{j}\, 0_j\rangle\rangle, & \langle\langle \bar{7}_{j+2}\rangle\rangle \,\bar{/}\, \langle\langle \varepsilon^{j+2}\rangle\rangle = \langle\langle 0_{j+1}\, 0_j\, \varepsilon^{j}\rangle\rangle.
\end{array}$$

Table 4

	σ–definitions in pseudosymbols		
1	$\lambda_4^1 \rightleftharpoons \langle\langle \bar{4}_3\, \bar{0}_3\rangle\rangle$		
2	$\lambda_4^2 \rightleftharpoons \langle\langle \bar{0}_3\, \bar{1}_3\rangle\rangle$		
3	$\lambda_3^3 \rightleftharpoons \langle\langle \bar{3}_1^1\, \bar{0}_2\rangle\rangle,$	$\lambda_4^3 \rightleftharpoons \langle\langle \lambda_3^3\, \bar{0}_3\rangle\rangle$	
4	$\lambda_4^4 \rightleftharpoons \langle\langle 7_2^1\, \bar{0}_3\rangle\rangle$		
5	$\lambda_4^5 \rightleftharpoons \langle\langle \bar{0}_3\, \bar{1}_2^1\rangle\rangle$		
6	$\lambda_3^6 \rightleftharpoons \langle\langle \bar{0}_2\, \bar{2}_2\rangle\rangle,$	$\lambda_4^6 \rightleftharpoons \langle\langle\, (\lambda_3^6)^1\rangle\rangle$	
7	$\lambda_3^7 \rightleftharpoons \langle\langle \bar{1}_2\, \bar{0}_2\rangle\rangle,$	$\lambda_4^7 \rightleftharpoons \langle\langle\, \lambda_3^7\, \bar{0}_3\rangle\rangle$	
8	$\lambda_3^8 \rightleftharpoons \langle\langle \bar{0}_2\, \bar{3}^2\rangle\rangle,$	$\lambda_4^8 \rightleftharpoons \langle\langle\, \bar{0}_3\, \lambda_3^8\rangle\rangle$	
9	$\lambda_3^9 \rightleftharpoons \langle\langle \bar{0}_2\, \bar{3}^2\rangle\rangle,$	$\lambda_4^9 \rightleftharpoons \langle\langle\, \lambda_3^9\, \bar{0}_3\rangle\rangle$	
10	$\lambda_3^{10} \rightleftharpoons \langle\langle \bar{0}_2\, 3^2\rangle\rangle,$	$\lambda_4^{10} \rightleftharpoons \langle\langle\, \bar{0}_3\, \lambda_3^{10}\rangle\rangle$	
11	$\lambda_3^{11} \rightleftharpoons \langle\langle (\bar{0}_1\, 3^1)^1\rangle\rangle,$	$\lambda_4^{11} \rightleftharpoons \langle\langle\, \lambda_3^{11}\, \bar{0}_3\rangle\rangle$	
12	$\lambda_2^{12} \rightleftharpoons \langle\langle 3^1\, \bar{0}_1\rangle\rangle,$	$\lambda_3^{12} \rightleftharpoons \langle\langle\, \lambda_2^{12}\, \bar{0}_2\rangle\rangle,$	$\lambda_4^{12} \rightleftharpoons \langle\langle (\lambda_3^{12})^1\rangle\rangle$
13	$\lambda_4^{13} \rightleftharpoons \langle\langle \bar{0}_3\, 7_1^2\rangle\rangle$		
14	$\lambda_3^{14} \rightleftharpoons \langle\langle \bar{1}_1^1\, \bar{0}_2\rangle\rangle,$	$\lambda_4^{14} \rightleftharpoons \langle\langle (\lambda_3^{14})^1\rangle\rangle$	
15	$\lambda_2^{15} \rightleftharpoons \langle\langle \bar{0}_1\, \bar{4}_1\rangle\rangle,$	$\lambda_4^{15} \rightleftharpoons \langle\langle (\lambda_2^{15})^2\rangle\rangle$	
16	$\lambda_2^{16} \rightleftharpoons \langle\langle 5^1\, \bar{0}_1\rangle\rangle,$	$\lambda_3^{16} \rightleftharpoons \langle\langle \lambda_2^{16}\bar{0}_2\rangle\rangle,$	$\lambda_4^{16} \rightleftharpoons \langle\langle \lambda_3^{16}\bar{0}_3\rangle\rangle$
17	$\lambda_2^{17} \rightleftharpoons \langle\langle (^{05})^1\rangle\rangle,$	$\lambda_3^{17} \rightleftharpoons \langle\langle \bar{0}_2\, \lambda_2^{17}\rangle\rangle,$	$\lambda_4^{17} \rightleftharpoons \langle\langle \bar{0}_3\, \lambda_3^{17}\rangle\rangle$

Strictly speaking, the equivalences stated above suffice for the realization of all conjuctions over disjunctions, notated by σ–operators in Sh system, in the SAT problem. However, for the sake of convenience, we shall write out lemma 2 (for $j = 0$) and lemma 3 in the table 2 (the brackets are again implied).

Now we shall further expound the illustrated example from the table 3.

The results of conjuction, showed in the last column of the table 3, are acquired relatively simply, and gain a rather *"mechanical"* character, under the condition of introducing recursive definitions with the use of pseudosymbols λ_j^i, μ_j^i, η_j^i, ... for σ–operators in double angle brackets. We should pay attention to the fact that the meaning of the symbols is somewhat different from the one stated at the beginning of the paper: the lower index in the pseudosymbols points to the rank of the vector, the upper— only to the index of the vector, i.e. λ^i, μ^i, η^i are in a fact i–th symbols, but are all of the row j. The next to the last column of the table 3 is shown in the table 4, the first computations stating the following:

$$
\begin{aligned}
1)\quad & \mu_4^1 = \lambda_4^1 \,\overline{/}\, \lambda_4^2 = \langle\langle \bar{4}_3\, \bar{0}_3\rangle\rangle \,\overline{/}\, \langle\langle \bar{0}_3\, \bar{1}_3\rangle\rangle = \langle\langle \bar{4}_3\, \bar{1}_3\rangle\rangle;\\
2)\quad & \mu_4^2 = \mu_4^1 \,\overline{/}\, \lambda_4^3 = \langle\langle \bar{4}_3\, \bar{1}_3\rangle\rangle \,\overline{/}\, \langle\langle \lambda_3^3\, \bar{0}_3\rangle\rangle = \langle\langle \mu_3^2\, \bar{1}_3\rangle\rangle,\\
& \mu_3^2 = \langle\langle \bar{4}_3\rangle\rangle \,\overline{/}\, \langle\langle \lambda_3^3\rangle\rangle = \langle\langle \bar{0}_2\, \bar{3}_2\rangle\rangle \,\overline{/}\, \langle\langle \bar{3}_1^1\, \bar{0}_2\rangle\rangle = \langle\langle \bar{3}_1^1\, \bar{3}_2\rangle\rangle;\\
3)\quad & \mu_4^3 = \mu_4^2 \,\overline{/}\, \lambda_4^4 = \langle\langle \mu_3^2\, \bar{1}_3\rangle\rangle \,\overline{/}\, \langle\langle 7_2^1\, \bar{0}_3\rangle\rangle = \langle\langle \mu_3^3\, \bar{1}_3\rangle\rangle,\\
& \mu_3^3 = \langle\langle \mu_3^2\rangle\rangle \,\overline{/}\, \langle\langle 7_2^1\rangle\rangle = \langle\langle \bar{3}_1^1\, \bar{3}_2\rangle\rangle \,\overline{/}\, \langle\langle 7_2\, 7_2\rangle\rangle = \langle\langle 2_2\, 4_2\rangle\rangle;\\
4)\quad & \mu_4^4 = \mu_4^3 \,\overline{/}\, \lambda_4^5 = \langle\langle \mu_3^3\, \bar{1}_3\rangle\rangle \,\overline{/}\, \langle\langle \bar{0}_3\, \bar{1}_2^1\rangle\rangle = \langle\langle \mu_3^3\, \eta_3^4\rangle\rangle,\\
& \eta_3^4 = \langle\langle \bar{1}_3\rangle\rangle \,\overline{/}\, \langle\langle \bar{1}_2^1\rangle\rangle = \langle\langle \bar{3}_2\, \bar{0}_2\rangle\rangle \,\overline{/}\, \langle\langle \bar{1}_2\, \bar{1}_2\rangle\rangle = \langle\langle \bar{3}_2\, \bar{1}_2\rangle\rangle;\\
5)\quad & \mu_4^5 = \mu_4^4 \,\overline{/}\, \lambda_4^6 = \langle\langle \mu_3^3\, \eta_3^4\rangle\rangle \,\overline{/}\, \langle\langle (\lambda_3^6)^1\rangle\rangle = \langle\langle \mu_3^5\, \eta_3^5\rangle\rangle,\\
& \mu_3^5 = \mu_3^3 \,\overline{/}\, \lambda_3^6 = \langle\langle 2_2\, 4_2\rangle\rangle \,\overline{/}\, \langle\langle \bar{0}_2\, \bar{2}_2\rangle\rangle = \langle\langle 2_2\, 4_2\rangle\rangle,\\
& \eta_3^5 = \langle\langle \eta_3^4\rangle\rangle \,\overline{/}\, \langle\langle \lambda_3^6\rangle\rangle = \langle\langle \bar{3}_2\, \bar{1}_2\rangle\rangle \,\overline{/}\, \langle\langle \bar{0}_2\, \bar{2}_2\rangle\rangle = \langle\langle \bar{3}_2\, \bar{3}_2\rangle\rangle = \langle\langle \bar{3}_2^1\rangle\rangle; \text{ and so on.}
\end{aligned}
$$

In the subtable 3 (the last column of the table) the results of conjuctions are shown (as the result of the application of the calculus explained above): in the i–th row the result of conjuctions over σ–operators (from the next to the last row) starting with the first and ending with the i–th. The σ–operator from the last column of the 17–th row (separated from the rest)) therefore presents the result of conjuction of all 17 operators of the second-last subtable. The separated operator implies the subtable, whose result it is, satisfiable, with the satisfiability achieved on the value rows of the variables with the base ten indices 4; 42; 42, i.e. the rows:

$$(001000), \quad (010101), \quad (110101). \tag{3}$$

We should notice that the table 3 (with the 19 rows shown in it) is contradictory, because the resulting σ–operator equals $\langle\langle 0_4\rangle\rangle$.

In order to illustrate another important idea on this simple example, let us assume that in the each row of the resulting table a maximum of R (in our example $R = 4$) literals–digits can be entered (with or without indices), and that the problems in 3 should be solved with this limitation borne in mind. In that case the last subtable from the table 3 would look as the first subtable of the table 5 (let us notice that the rows would stay unchanged up to the eigth row). By looking at the 17–th row, we reach the conclusion (based on the beginning of the separated operator) of satisfiability confirmed by the row of the corresponding variable values (the first row in 3).

If table 3 contains all 19 rows, then in the 19–th row of the table 5 the result is gained, with which we are returning to the 8–th row to acquire the σ–operators shown in the second subtable of the table 4. The result of the 18–th row tells us about the problem being contradictory.

Table 5

N oper.	1	N oper.	2
8	$\langle\langle 2_2\, 4_2\, \ldots\rangle\rangle$	8, 9	$\langle\langle 0_3\, \bar{3}_2\, 0_1\, \bar{3}^1\rangle\rangle$
9, 10	$\langle\langle 2_2\, 0_1\, 2_1\, \ldots\rangle\rangle$	10, 11, 12	$\langle\langle 0_3\, \bar{3}_2\, 0_2\rangle\rangle$
11	$\langle\langle 2_2\, 0_2\, \ldots\rangle\rangle$	13	$\langle\langle 0_3\, 0_1\, 7_1\, 0_2\rangle\rangle$
12, 13, 14, 15	$\langle\langle 4_1\, 0_1\, 0_2\, \ldots\rangle\rangle$	14	$\langle\langle 0_3\, 0_1\, 6_1\, 0_2\rangle\rangle$
16, 17	$\langle\langle \mathbf{0\ 1\ 0}_1\ \mathbf{0}_2\, \ldots\rangle\rangle$	15, 16, 17	$\langle\langle 0_3\, 0_1\, 2_1\, 0_2\rangle\rangle$
18	$\langle\langle 0\, 1\, 0_1\, 0_2\, \ldots\rangle\rangle$	18	$\langle\langle \mathbf{0}_4\rangle\rangle$
19	$\langle\langle \mathbf{0}_3\rangle\rangle$		

Examinations and calculations illustrated on individual examples have a universal character, which makes us conclude that in the general case the text of the resulting σ–operator for the standardized table of the SAT problem cannot have exponential growth. Let us name every σ–operator literal–digit with or without indices, different from $\langle\langle 0_s\rangle\rangle$, for $s \geq 0$, significant component.

Then we shold look at the lemmas 1, 2 and 4 and the tables 1 and 2 (which are sufficient to calculate conjuctions) and notice the following: a conjuction over two significant components cannot generate more then two significant components. The components $\langle\langle 0_s\rangle\rangle$ are absorbtive, because of which their appearance cannot facilitate a strong text growth. The volume of the text doesn't even have to be estimated: it should be noticed that for the solution of the SAT problem the acceptance of the important idea stated below would be sufficient, with putting $R = 2n$ in the general case. In our example it would be necessary to put $R = 12$, but in that case there would be no return (surplus), which was something we wanted to illustrate. All this being said, the statement of the following conclusion becomes possible.

Theorem 2 The class of NP–complete problems is equivalent to the class P.

With this theorem we are bringing our fragment of the notation and logic calculus, oriented on the SAT problem, to an end. Let us notice that, in case of necessity of solving more general classes of problems from the logic algebra domain (such as direct calculations demanded by logic problems, without reducing them to standard representatives, then the Zegalkin systems, criptography problems, etc.) it would, without major effort, be possible to widen the calculations to encompass the whole operator system.

$$Op = \{\downarrow, \oplus, /, \overline{/}, \bar{\oplus}, \bar{\downarrow}\},$$

That shall, however, not be demonstrated in this paper. Instead we shall move on to multivalued logic and show how the positionality principle can also be applied on such cases.

5. SYMBOLS AND DEFINITIONS FOR MULTIVALUED LOGICS

Let $PL_{k\cdot m}$ — be a symbol for positional multivalued logic, for integers k and $m \geq 2$. If $k = m$, we write PL_k.

In $PL_{k\cdot m}$ let us assume that the arguments of the function $f(x_1, x_2, \ldots, x_n)$ belong to the set $E_k = \{0, 1, 2, \ldots, k-1\}$, and values to the set $G_m = \{0, 1, 2, \ldots, m-1\}$.

The function $f(x_1, x_2, \ldots, x_n)$ is completely stated if its table is stated, in which the rows of arguments are notated as representations of numbers $0, 1, 2, \ldots, k^n - 1$ in the k–ary number system, with increase of the digit position from left to the right (attention has to be payed to the latter, since it is not an usual notation). The symbol f is interpreted as mapping determined by the table, and the symbols $x_1, x_2, \ldots, x_n$ — as the names of the columns. Over the columns, as over the one variable functions, unary operations are carried out, given by the permutations from the symetric group S_k.

All that has been said presents a complete repetition of classic conventions, and the mentioned positionality begins with the representations of f symbols with the positional s– operators and column notations via σ–operators.

For $PL_{k\cdot m}$ q–ary s–operators are vectors of the dimension $q \cdot k - 1$ with the coordinates from the set G_m. The numeration of the vector coordinates from left to the right begins with index 0, whereby in the case of $m \leq 10$ it is not necessary to separate the coordinates with a comma. In all other cases comma is obligatory.

Example I. In the case of PL_2 such binary, nontrivial s–operators are operators from Op, the first three of them having the following form:

$$\downarrow = \langle 100\rangle, \quad \oplus = \langle 010\rangle, \quad / = \langle 110\rangle.$$

As far as the next three s–operators are concerned, they emerge as a result of the application of the operation of unary inversion:

— $\rightleftharpoons$ (01):

$$\bar{/} = \langle 001\rangle, \quad \bar{\oplus} = \langle 101\rangle, \quad \bar{\downarrow} = \langle 011\rangle,$$

which represents the transposition from the group S_2.

Example II. In the case of PL_3, binary nontrivial s–operators are operators from the next two sets:

$$\begin{aligned} O_p' = \{&\langle 00001\rangle, \langle 00010\rangle, \langle 00011\rangle, \langle 00101\rangle, \langle 00110\rangle,\\ &\langle 01001\rangle, \langle 00100\rangle, \langle 01010\rangle, \langle 01110\rangle\},\end{aligned}$$

$$\begin{aligned} O_p'' = \{&\langle 00012\rangle, \langle 00102\rangle, \langle 00112\rangle, \langle 00120\rangle, \langle 00121\rangle,\\ &\langle 00122\rangle, \langle 01002\rangle, \langle 01012\rangle, \langle 01020\rangle, \langle 01021\rangle\\ &\langle 01102\rangle, \langle 01112\rangle, \langle 01120\rangle, \langle 01201\rangle, \langle 01210\rangle\}.\end{aligned}$$

All the other nontrivial binary s–operators from PL_3 emerge from the ones counted in $O_p^{'}$ and $O_p^{''}$ as the result of the application of the following unary operations (permutations from the group S_3):

$$- \rightleftharpoons (01),\ \neg \rightleftharpoons (02),\ \bullet \rightleftharpoons (12),\ \rightarrow \rightleftharpoons (012),\ \leftarrow \rightleftharpoons (021) \tag{4}$$

and the operation of conjugation (see [3] p. 29): (For the given vector $\langle R_n \rangle = \langle r_1, r_2, \ldots, r_n \rangle$ conjugated with them is the vector $\langle R^* \rangle = \langle r_n, \ldots, r_2, r_1 \rangle$.

If α is the symbol for the first vector from $O_p^{''}$, we get:

$\alpha = \langle 00012 \rangle$, $\bar{\alpha} = \langle 11102 \rangle$, $\overset{\neg}{\alpha} = \langle 22210 \rangle$, $\overset{\bullet}{\alpha} = \langle 00021 \rangle$, $\overset{\rightarrow}{\alpha} = \langle 11120 \rangle$, $\overset{\leftarrow}{\alpha} = \langle 22201 \rangle$,

$\alpha^* = \langle 21000 \rangle$, $\bar{\alpha}^* = \langle 20111 \rangle$, $\overset{\neg}{\alpha}^* = \langle 01222 \rangle$, $\overset{\bullet}{\alpha}^* = \langle 12000 \rangle$, $\overset{\rightarrow}{\alpha}^* = \langle 02111 \rangle$, $\overset{\leftarrow}{\alpha}^* = \langle 10222 \rangle$.

Example III. In $PL_{2\cdot 3}$ binary nontrivial s–operators are vectors:

$$\pi = (012),\ \bar{\pi} = (102),\ \overset{\neg}{\pi} = (210),\ \overset{\bullet}{\pi} = (021),\ \overset{\rightarrow}{\pi} = (120)\ \overset{\leftarrow}{\pi} = (201) \tag{5}$$

for which the following equivalences apply:

$$\pi = \overset{\neg}{\pi}^*,\ \bar{\pi} = \overset{\leftarrow}{\pi}^*,\ \overset{\bullet}{\pi} = \overset{\rightarrow}{\pi}^*.$$

Beside the vector (5), the application of unary operations (4) on Op also results in vectors:

$$\downarrow = \bar{/}^* = \langle 100 \rangle,\quad \bar{\downarrow} = /^* = \langle 011 \rangle,\quad \overset{\neg}{\downarrow} = \overset{\rightarrow}{/}^* = \langle 122 \rangle,$$
$$\overset{\bullet}{\downarrow} = \overset{\leftarrow}{/}^* = \langle 200 \rangle,\quad \overset{\rightarrow}{\downarrow} = \overset{\neg}{/} = \langle 211 \rangle,\quad \overset{\leftarrow}{\downarrow} = \overset{\bullet}{/}^* = \langle 022 \rangle$$

and selfconjugated vectors:

$$\overset{\neg}{\oplus} = \langle 212 \rangle,\ \overset{\bullet}{\oplus} = \langle 020 \rangle,\ \overset{\rightarrow}{\oplus} = \langle 121 \rangle,\ \overset{\leftarrow}{\oplus} = \langle 202 \rangle.$$

The application of simple, as well as the complex s–operator on the vector argument is well known (see [3] pp. 25 – 26) and shall therefore not be repeated here.

Now we shall say a few words about the column notation via σ–operator. In a general case, when we are dealing with $PL_{k\cdot m}$, double angle brackets are used to notate a vector of the dimension k^n (i.e. of the rank n, where the base k is defined through PL), which shall be further referred to as σ–operators.

The following convention shall be upheld: if $\langle\langle \alpha \rangle\rangle$ — is a vector of the rank h (dimension k^h), then the notation $\langle\langle \alpha_j \rangle\rangle$ denotes that every coordinate of the vector $\langle\langle \alpha \rangle\rangle$ appears with the arity j (of the dimension k^j), because of which the vector α has the rank $h + j$. The notation

$\langle\langle\alpha^i\rangle\rangle$ denotes that the vector $\langle\langle\alpha\rangle\rangle$ itself appears in the concatenation h^i times. In that way, the notation $\langle\langle\alpha^i_j\rangle\rangle$ — is a vector of the rank $h+j+i$, under the condition of $\langle\langle\alpha\rangle\rangle$ — being a vector of the rank h.

Once again we shall emphasize that the double angle brackets point to the coordinates appearing in the exponential frequency, as well as the fact that the upper (as well as lower) vector indices (or/and its coordinates) point to the exponential character of the appeareance, and that the indices are always implied (if not stated, then equal to *zero*).

In already discussed $PL_{k\cdot m}$ the colums are:

$$x_j \;\simeq\!\rightarrow\; \langle\langle\rho^{n-j}_{j-1}\rangle\rangle, \quad \overset{\triangle}{x}_j \simeq\!\rightarrow \langle\langle\overset{\triangle}{\rho}{}^{\,n-j}_{j-1}\rangle\rangle, \tag{6}$$

where the vector is $\langle\langle\rho\rangle\rangle = \langle\langle 0,\, 1,\, 2,\, \ldots\,,\, k-1\rangle\rangle$, and the unary operation $\triangle$ — one of the permutations from the symmetric group S_k.

A special case (see. [3] p. 100) for PL_2 is:

$$x_j \;\simeq\!\rightarrow\; \langle\langle\bar{\tau}^{n-j}_{j-1}\rangle\rangle, \quad \bar{x}_j \;\simeq\!\rightarrow\; \langle\langle\tau^{n-j}_{j-1}\rangle\rangle,$$

where $\langle\langle\tau\rangle\rangle = \langle\langle 10\rangle\rangle$.

In the case of $PL_{3\cdot m}$, in correspondance with (5) and (6), the following holds:

$$x_j \;\simeq\!\rightarrow\; \langle\langle\pi^{n-j}_{j-1}\rangle\rangle, \;\; \bar{x}_j \;\simeq\!\rightarrow\; \langle\langle\bar{\pi}^{n-j}_{j-1}\rangle\rangle, \;\; \vec{x}_j \simeq\!\rightarrow\; \langle\langle\vec{\pi}^{\,n-j}_{j-1}\rangle\rangle,$$
$$\overset{\bullet}{x}_j \simeq\!\rightarrow\; \langle\langle\overset{\bullet}{\pi}{}^{n-j}_{j-1}\rangle\rangle, \;\; \overrightarrow{x}_j \simeq\!\rightarrow\; \langle\langle\overrightarrow{\pi}^{\,n-j}_{j-1}\rangle\rangle, \;\; \overleftarrow{x}_j \simeq\!\rightarrow\; \langle\langle\overleftarrow{\pi}^{\,n-j}_{j-1}\rangle\rangle.$$

Further, the calculations in $PL_{k\cdot m}$ along the scheme, analog to the scheme in paragraph 4 of this paper, should be explained in detail, but such a task would be difficult to realize within the limits of this paper, because of which this exposition shall be brought to an end by some simple illustrating examples, shown in the table 6, where:

$R = \langle 01120\rangle$, $Q = \langle 01201\rangle$, $W = \langle 01210\rangle$, whereby
$F = W(R,\, Q) = \langle 02121\rangle$ holds.

It can be read from the table columns that

$$\begin{aligned}
W(R(x_1,x_2),\, Q(x_1,x_2)) &= F(x_1,x_2),\\
W(R(x_1,\vec{x}_2),\, Q(x_1,\vec{x}_2)) &= F(x_1,\vec{x}_2),\\
W(R(\overset{\bullet}{x}_1,x_2),\, Q(\overset{\bullet}{x}_1,x_2)) &= F(\overset{\bullet}{x}_1,x_2),\\
W(R(\overset{\bullet}{x}_1,\vec{x}_2),\, Q(\overset{\bullet}{x}_1,\vec{x}_2)) &= F(\overset{\bullet}{x}_1,\vec{x}_2).
\end{aligned}$$

These examples represent a confirmation of the results for PL_3, analog to the result of the theorem 3 (see [3] p. 77) for PL_2.

Table 6

x_1	x_2	R	Q	W	F		x_1	$\bar{x}_2$	R	Q	W	F		$\dot{x}_1$	x_2	R	Q	W	F		$\dot{x}_1$	$\bar{x}_2$	R	Q	W	F
0	0	0	0	0	0		0	2	1	2	1	1		0	0	0	0	0	0		0	2	1	2	1	1
1	0	1	1	2	2		1	2	2	0	2	2		2	0	1	2	1	1		2	2	0	1	1	1
2	0	1	2	1	1		2	2	0	1	1	1		1	0	1	1	2	2		1	2	2	0	2	2
0	1	1	1	2	2		0	1	1	1	2	2		0	1	1	1	2	2		0	1	1	1	2	2
1	1	1	2	1	1		1	1	1	2	1	1		2	1	2	0	2	2		2	1	2	0	2	2
2	1	2	0	2	2		2	1	2	0	2	2		1	1	1	2	1	1		1	1	1	2	1	1
0	2	1	2	1	1		0	0	0	0	0	0		0	2	1	2	1	1		0	0	0	0	0	0
1	2	2	0	2	2		1	0	1	1	2	2		2	2	0	1	1	1		2	0	1	2	1	1
2	2	0	1	1	1		2	0	1	2	1	1		1	2	2	0	2	2		1	0	1	1	2	2

6. CONCLUDING REMARKS

The proof of equivalence of P and NP class problems (the second one), expounded in this paper, is basically already present in [3] (see. 8.1), even if not in an explicit form. The main issue not explicated in [3], is the important idea of calculating with the bound R applied to the memory volume for literals-digits with or without indices. I have assumed that this result was going to be recognized without additional emphasis, which turned out to be unrealistic.

I should also mention, that although the first proof of the equivalence of P and NP class problem was carried out by the end of the last century, the second was carried out at the very beginnig of the present century. I should also state that it is possible to acquire a third proof, as a common result of the last two results. However, I believe that the more important issue is the application of the sigma–notation on superreduction, and even more important is the computer realization based on σ–notation, for its capability to make possible on existing computers what is expected of quantum computers.

REFERENCES

[1] Michael R.Garey and David S. Johnson. Computers and Intractability: A Guide to the Theory of NP–Completeness. W.H.Freeman, 1979.

[2] Stephen Cook. The complexity of theorem proving procedures. In Proceedings of the Third Annual ACM Symposium on Theory of Computing, pp. 151 – 158, 1971.

[3] Тельпиз М.И. Принцип позиционности для счисления и исчисления функций. М.; ИКИ РАН, 2001, 457 с.

[4] Тельпиз М.И. NP–полнота, суперприведение и проблема четырёх красок. http://www.tarusa.ru/∼mit/ENG/4col_e.php

Recieved: 7 December 2004

UDC:004.415.2
Original scientific paper

A UML PROFILES FOR VIEWPOINT-ORIENTED MODELING

Abdellatif Hair
Faculty of Sciences and Technics, Beni-mellal, Morocco
hair@fstbm.ac.ma

Abstract: *The viewpoint-oriented modeling consists in providing an unique model accessible by users with various access rights according to their needs. The system is represented as sub-systems (called sub-models). This cutting makes the design of system easier by an autonomous design of its sub-systems.*

UML is a standard modeling language and supports extension mechanism to allow tailoring UML to fit the needs of a specific domain. UML Profile is a predefined set of extension mechanisms. UML profile allows the stakeholders of a certain domain to express the semantics of their systems using a well-defined set of extensions. In this paper we propose an initial discussion on UML profile for viewpoint oriented modeling, which extends the standard UML by incorporating new concepts like views, viewpoints, flexible, visibility relationships, etc.

Keywords: *Modeling, viewpoint, view, visibility relationship, profile, UML.*

1. INTRODUCTION

The development of complex software systems involves many users (agents) with different perspectives of the system they are trying to describe. These perspectives, or viewpoints, are usually partial or incomplete. Software engineers have recognized the need to identify and use this multiplicity of viewpoints when creating software systems [1][5][15].

For that, the viewpoint concept has received significant attention and has been used for different purposes such as a filter, subject, role, state, perspective, and view [8][20][2]. We recover these different concepts in object-oriented databases, knowledge representation systems and object-oriented analysis and design methods. In object-oriented analysis and design, one of the results more succeed is the one of VBOOM which is a method integrating the viewpoint approach in the object-oriented modeling [3][11]. And also the work that consists in elaborating an unified oriented-viewpoint method (named U_VBOOM) based on the VBOOM method [6][7].

The Viewpoint-Oriented Modeling (VOM) is concerned with providing mechanisms for explicitly capturing crosscutting user's need, and therefore, enhancing modularity. In VOM, systems are represented as sub-systems (called sub-models). Each one reply a particular user's need. Systems, however, have properties that cut across their functional components increasing their interdependencies. Viewpoint-oriented is a paradigm that complements the

object-oriented concept and helps in avoiding the endless circle of reengineering entire software projects for minor changes.

Since UML is a graphical language for specifying, visualizing, constructing, and documenting the artifacts of software systems [23][24]. UML is a standard modeling language endorsed by Object Management Group (OMG) in 1997 as UML 1.1. UML is the graphical notation that defines the semantics of the object meta-model (model of UML itself), and defines a notation for capturing and communicating object structure and behavior. UML supports extension mechanisms [19][21][23][24] (stereotypes, tagged values, and constrains) to allow tailoring UML to fit the needs of a specific domain. UML Profile [16][10][12] is a predefined set of extension mechanisms. UML profile allows the stakeholders of a certain domain to express the semantics of their systems using a well-defined set of extensions. The Profile does not extend UML by adding a new concept. Instead, it provides connections for applying and specializing UML to a specific domain.

This paper addresses VOM at the analysis and design phases. The purpose of the paper is to present a UML profiles for VOM. The viewpoint concept will be represented as standard object-oriented artifacts, which may be integrated in object-oriented modeling. We will use an UML-like notation [19] to illustrate our subjects.

The paper is organized as follows: section 2 describes and defines the terms used in the VOM. Section 3 discusses briefly the UML profiles. The proposed UML profiles for VOM is described in section 4 and this section gives an example that illustrates the practicality of the VOM. Finally, section 5 presents our conclusions.

In this paper, we are going to illustrate our subjects with the **Media Library** Management example. The specifications of this example are more explained in [13]. The **Media Library** system must allow its members to consult and to borrow various types of support: books, video and audio disks, audio CD, etc. Only one member of the library can borrow books, reviews, etc. The borrow is limited in time. The potential users of the **Media Library** system are: the **librarian** who manages the loans, the **adherents responsible** who will add and withdraw members, the **exemplaries responsible** who will seize the new exemplaries and to withdraw those damaged, and finally the **system engineer** who ensures the good exploitation of system for the users.

2. VIEWPOINT-ORIENTED MODELING

The development of complex software systems requires the building of several partials models, each one reply a particular user's need. The use of multiple viewpoints in complex system modeling provides a big profit. But each author deals differently with those notions. Our approach in VOM consists in providing an unique model accessible by users with various access rights according to their needs and supporting multiple and evolutive viewpoint. This approach guarantees knowledge centralization, consistency checking and non-redundancy information and supports multiple and evolutive viewpoint. We will first define briefly the visibility relationship and its derived concepts.

By visibility, we mean that an entity can be seen upon several angles. It's the fact of a **Media Library** system, which can be studied under the Exemplaries responsible angle, Adherents responsible or Librarian one. The syntax and the semantic of this concept is detailed in [14], we summarize here the principal characteristic.

A **view** is an abstraction of the model. It constitutes the unity of visibility; it is the result of factorizing user's needs.

A viewpoint is the sight that has a user to the model. It 's a combination of views.

A **flexible class** is a class that declares more than two views; its instances, which must specify a particular viewpoint, take various appearances. The Media_Library class is a flexible class, which owns 3 views (Loans, Exemplaries, Counts_Adherents).

A sub-**model** is a partial model or sub-system, operating different knowledge of system. Each one concerns a specific viewpoint. In the **Media Library** system, we distinguish three sub-models***:*** *Loans management*, *Exemplaries management*, and *Adherents management*. Those sub-models are associated respectively to viewpoints descended of three actors: *Exemplaries responsible*, *Adherents responsible* and *Librarian*.

3. UML PROFILES

UML [23] is the graphical notation that defines the semantics of the object meta-model (model of UML itself), and defines a notation for capturing and communicating object structure and behavior. In its meta-model architecture, UML supports extension mechanisms that allow tailoring it to fit the needs of a specific domain. These extension mechanisms are the means by which UML can adapt to new technologies, domains and methodologies. The Object Management Group (OMG) defines UML modeling language in a four-layer architecture as shown in Figure 1 [23][24]. The meta-meta-model level defined by the MOF (meta object facility), the UML meta-model is defined and standardized on top of the MOF, the meta-model layer specifies the modeling language, the model layer defines the model, and the user object layer defines instances of the model. The standard meta-models represent areas under which specific domains exist that require a specialization of their domain. In order to provide mechanisms for defining domain standards specializing the standard meta-models, UML introduces the notion of Profile, which is specified in white paper [16]. UML provides the means by which one can extend it's meta-model (model of UML itself) to fit the needs of a modeling concern. It provides extension mechanisms that allow customizing and extending UML model elements (i.e. <<class>>). The UML extension mechanisms are stereotype, tagged values, and constrains. Stereotypes are classification of an existing UML model element, which has its own properties that are expressed as tagged values, constrains are restrictions placed on the stereotypes. UML profiles are predefined set of stereotype, tagged values, constrains, and graphical icons to allow modeling a specific domain.

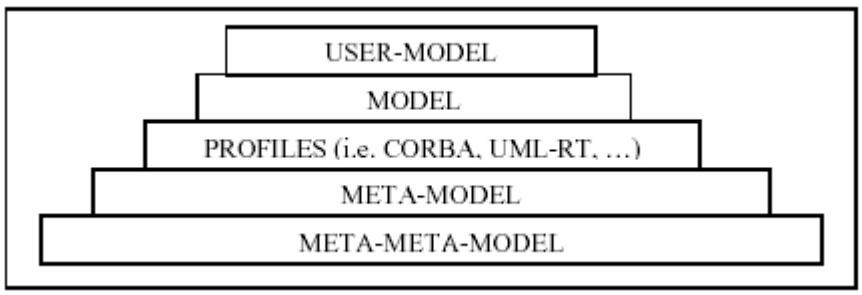


Figure 1: UML architecture

There have been many UML profiles proposed to OMG, UML profile for real time systems UML-RT [23] is a UML extension to support Real time intensive application, the profile itself is based on concepts from [22] the profile introduces the notions Capsules, ports, connectors, protocols, and protocol role to UML. CORBA Profile [17] CORBA allows applications to communicate with each other regardless of the location and design, the UML Profile for CORBA provides a standard means for expressing the semantics of CORBA IDL using UML artifacts, which enable expressing these artifacts with UML tools. UML Profile for CORBA has been adopted by OMG in October of 2000.

In the following section we show our proposed UML Profiles to support viewpoint-oriented software development. Our proposed Profiles introduce basic VOM terminology to UML to enable the visual representations of viewpoint-oriented systems in the design models.

4. UML PROFILES FOR VIEWPOINT-ORIENTED MODELING

UML profiles for VOM will provide conventions and guidelines for applying and specializing standard UML to the graphical notation for VOM (Specialize UML meta-model for VOM). It would help in capturing every desired terminology of VOM and provide "native" support for VOM in UML. The profiles will set a common means of expressing VOM artifacts visually for all VOM stakeholders. We need to create a set of extensions (stereotypes, tagged values, and constraints) to support VOM and not a particular implementation of VOM. We believe that the extension mechanisms are capable of adapting UML semantics to express VOM terminology without changing the UML meta-model itself.

4.1. STEREOTYPES AND RELATIONSHIPS FOR VIEWPOINT-ORIENTED MODELING

Table 1 starts the list of stereotypes that we suggest for VOM. The 1st and 2nd row define <<flexible>> and <<view>> as stereotypes. These are stereotypes of the UML "Class" model element. The <<flexible>> stereotype is used to indicate a multi-views class. The <<view>> stereotype is used to indicate that a class is view of flexible class. Viewpoints or use types of an agent (user) through the system are concerns that cut across a system's main components. These components are called sub-models. These sub-models are sub-systems of system's main. The VOM attempts to modularize sub-models and uses an efficient process to melt those sub-models in order to get a coherent global model. For that the 3rd row of Table 1 defines <<sub-model>> as a stereotype, this stereotype is used to indicate that the use type of an agent through the system classifies packages (sub-systems).

Table 1: Candidates stereotypes for VOM

Base UML meta-model element	Tagged Value	Explanation
class	<<flexible>>	The <<flexible>> stereotype is used to indicate a flexible class (Figure 2).
class	<<view>>	The <<view>> stereotype is used to indicate that a class is view of a flexible class (Figure 2).
package	<<sub-model>>	The <<sub-model>> stereotype is used to indicate that the packages are classified by the use types of agents.

Table 2 starts the list of relationships that we suggest for VOM. The UML generalization relationship and the <<seen_as>> stereotype represent the relation between a flexible class and its views. This relation is named visibility relationship. Each flexible class has at least two-visibility relationship because the flexible class is a class that declares more than two views.

A View might have dependency relation over another view, that is to say, when the information of view source depends of other information of view target. If the design does not specify dependency relation, this means that the views are not dependent. To specify

that a view has dependency relation than another, this extension extends the UML "association" meta-model element with the stereotype <<view_dependency>> linking two views. The association's arrow comes from the view source to the view target. Table 2 shows the <<view_dependency>> stereotype. This mechanism of dependence between the views is called repercussion of modification.

Table 2: Candidates' relationships for VOM

Base UML meta-model element	Tagged Value	Explanation
generalization	<<seen_as>>	It's visibility relationship. This relation can be used between flexible class and its views. The relation arrow points from the flexible class to its views. For example, Media_Library is flexible class and it is linked to its 3 views Loans, Exemplaries, and Counts_Adherents by the visibility relationship (Figure 2).
association	<<view_dependency>>	This relation can be used between two views. The relation arrow points from the view source to the view target. For example, when an adherent does a loan, numbers of Exemplaries available to the library must be modified. Then, the association with <<view_dependency>> stereotype must be used between two views Loans and Exemplaries (Figure 2).

The extension mechanisms are based mainly on the stereotype concept, which provides a way of classifying elements so that they behave as if they were instances of the new meta-model. For example when defining <<flexible>> as a stereotype of the classifier <<class>> it is guaranteed that <<flexible>> behaves the same way if it was created as an instance of <<class>>. The newly constructed stereotype becomes a model element and can participate in all relationships a model element can (i.e. Dependency relationship, generalization, etc.).

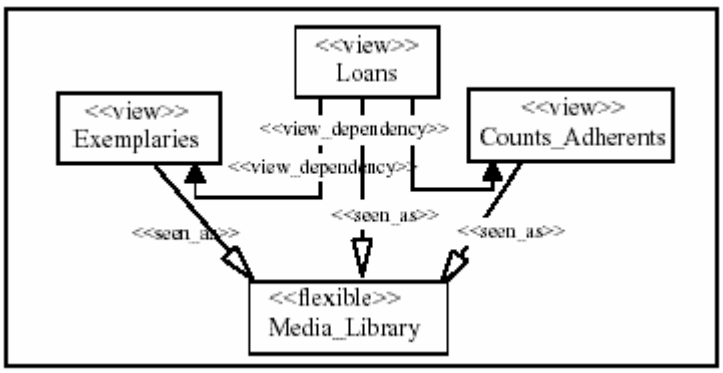


Figure 2: Flexible class and its views representation

3.2. ANALYSIS AND DESIGN PROFILES FOR VIEWPOINT-ORIENTED MODELING

In this section, we propose the practicality of VOM systems, we present it on the analysis and design profiles and we apply it on the **Media Library** example. The elaboration of these two profiles is based on the U_VBOOM method. U_VBOOM is an analysis/design method based on UML, which integrates the viewpoint approach in a coherent and deductive process [6][7]. U_VBOOM proposes two kinds of models: static and dynamic one to describe the system and its behaviors. U_VBOOM provides users the possibility to:

- deal strategically with problem space by identifying user's needs,
- improve the interaction expert/user and designer,
- establish sub-models (partials models),
- generate an unique model integrating different sub-models.

3.2.1. Analysis profile

The object of Analysis Profile is to define model components. In this profile, the analyst begins by elaborating the use cases diagram for system (Figure 3). Then, he identifies the different viewpoints (viewpoint per agent) and their combination of views. Three viewpoints are identified for **Media Library** system: *Librarian_viewpoint, Exemplaries_responsible_viewpoint,* and *adherents_responsible_viewpoint.*

The views elaboration of system is supported by the use cases description technique [9][7]. This technique consists in describing the use case as an action sequence that will make an agent to achieve his goal (Table 3) [4]. An action is an effect produced by an agent according the given way on the system or by the system itself. The use cases identified for agents and the intersection of their actions are going to cut the viewpoints in views. A view corresponds to set actions of one viewpoint or the intersection actions result of viewpoints group (Figure 4).

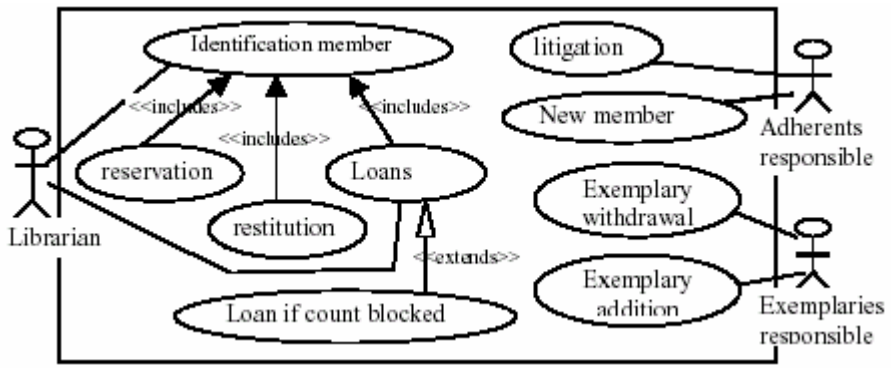


Figure 3: Use case diagram of MEDIA LIBRARY system

Table 3: Use cases decomposition in actions

Agents	Use case	Decomposition in actions
Librarian	Loan	a1: To identify an adherent a2: Count adherent (To check right loan for member) a3: To seek for an exemplary a4: To treat an exemplary (to validate the output of an exemplary) a5: To treat adherent (to indicate the loan by adherent)
	Reservation	a1: To identify an adherent a2: Count adherent (To check right loan for member) a3: To seek for an exemplary a6: To reserve an exemplary
	Restitution	a1: To identify adherent a3: To seek for an exemplary a4: To treat an exemplary (to validate the input of exemplary) a5: To treat adherent (to indicate the return of an exemplary)
	Loan if counts blocked	a1: To identify adherent a2: Count adherent (To check right loan for member)
	Identification member	a1: To identify adherent
Adherents responsible	New adherent	a2: Adherent account (to Add adherent)
	Litigation	a1: To identify adherent a2: Count adherent (Blocked count adherent) a7: To inform adherent
Exemplaries responsible	Exemplary addition	a3: To seek for an exemplary a8: To add copy
	Exemplary withdrawal	a3: To seek exemplary a9: To withdraw the damage exemplary

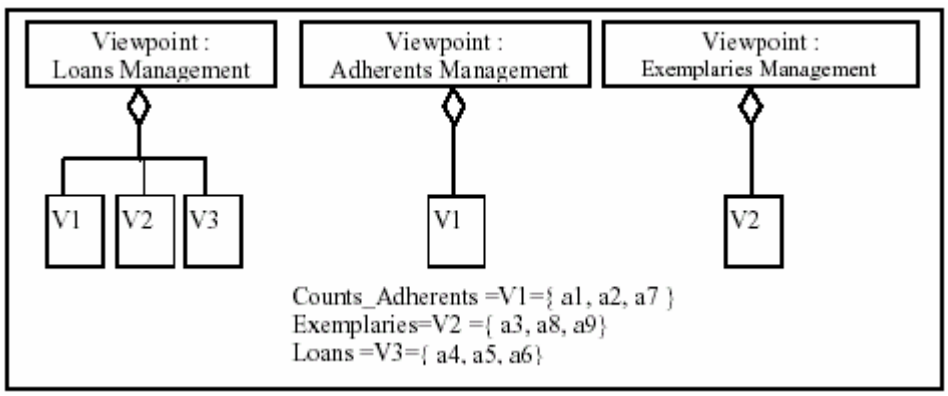


Figure 4: Viewpoint diagrams of MEDIA LIBRARY system

After, the Analysis Profile must enable the analyst to acquire scenarios as collaboration diagrams for each use case. The Collaboration diagrams concentrates on the structure of interaction between objects and their inter-relationships rather than focuses the temporal dimensions of a scenario. Thereafter, the potential classes of our system are identified and

the flexible classes are selected. At the end, the initial dictionary is elaborated and contains simple and flexible classes and views (Figure 5).

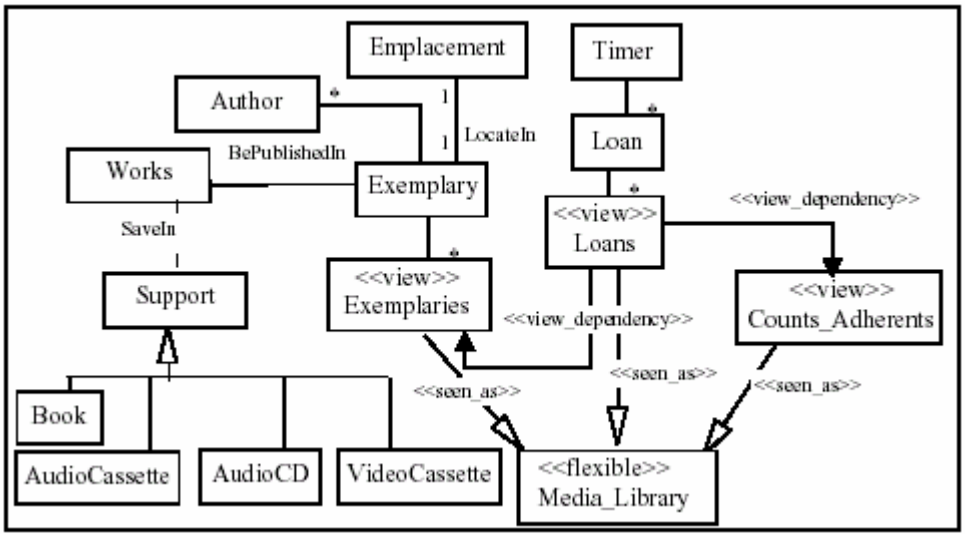


Figure 5: Initial class diagram of MEDIA LIBRARY system

3.2.2. Design profile

The analysis has been done with guidelines of the Analysis Profile. The Analysis Profile is enriched and is refined under the control and guidelines of the Design Profile.

The Design Profile must permit firstly to make the sub-models design, secondly to melt and to validate these sub-models. These sub-models constitute an essential artifact on the Design Profile.

3.2.2.1. Sub-models design

The Design Profile must allow the designers to develop their sub-models commonly or not. Three sub-models are identified for **Media Library** system: *Loans management*, *Exemplaries management*, and *Adherents management*. The designers must develop the static and dynamic representation for each sub-model. The Design Profile proposes the class diagrams to describe the static representation. Each class in the class diagram defines its own static structure (Interface) including attributes and methods. If a class is present in different sub-models, then its static structure corresponds to a partial description. For example, **Media_Library** flexible class of our system possesses a partial description in three sub-models. For the dynamic representation, the Design Profile proposes the sequence diagram and statechart diagram. A sequence diagram shows interactions among a set of objects in temporal order, which is good for understanding timing issues. A statechart diagram shows the sequence of states that an object goes through during its lifecycle in response to stimuli. Generally a statechart diagram may be attached to a class of objects with an interesting dynamic behavior. At the end, for each sub-model a partial dictionary is elaborated. Figure 6 represents the class diagram (partial) of Loans Management sub-model.

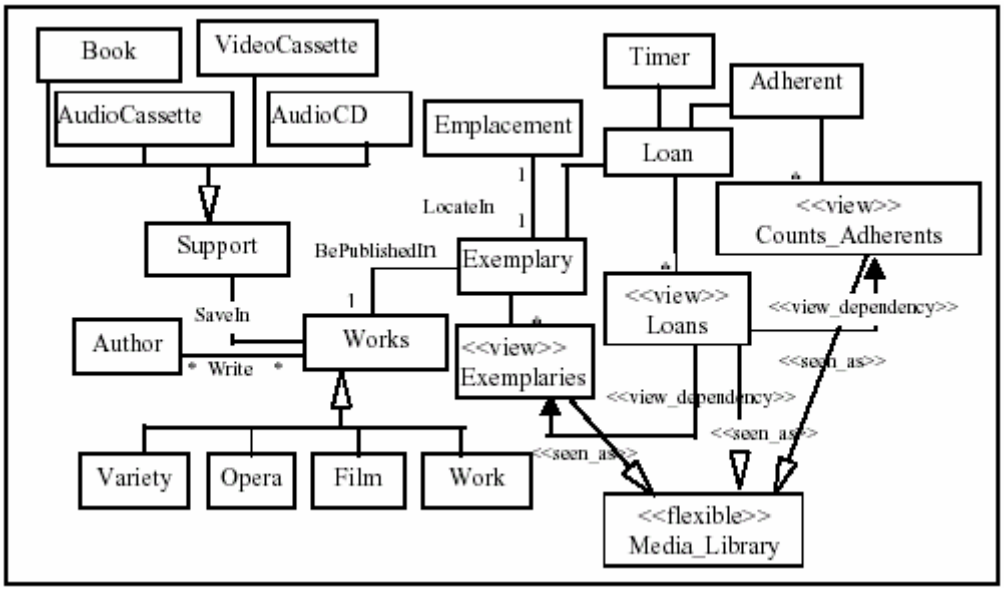


Figure 6: Class diagram of Loans Management sub-model

3.2.2.2. Melting and validation of sub-models

The Design Profile must also enable the designer to generate a coherent document (global model, dictionary of sub-models). That is to say, to allow the designer to melt partial interfaces, resolve conflicts and identify modifications to reverberate between classes of sub-models and provide the global architecture of system. To lead this operation of melting, an efficient process is purposed. This process is based on the following rules (A more detailed description of this process is in [7][11]):

- Same concepts must be represented by the same names: The melting rules will use names to define the semantics of the involved concepts. Thus, if two different sub-models had names Adherent and Member to represent the same concept, a melting procedure would have to be used to rename one of them. The renaming can be applied to classes, attributes, and methods.
- Attribute types must be consistent: attributes representing the same concept should also have the same type. That means that if the same attribute is an integer in one sub-model and a float in other, the melting rules will deal with these variations.
- Cyclic hierarchies must be avoided: the melting of sub-models cannot generate an inheritance cycle. If there is a sub-model-i within a certain class hierarchy such that class-sub is a subclass of class-super and a sub-model-j in which class-super is a subclass of class-sub there is an inconsistency and the melting rules can be applied.

At the end, the Design Profile must enable to validate the document and check coherency between global model and interfaces.

In our example, the global model is relatively identical to the **Loans Management** sub-model because this last is predominant, but this situation is not a generality (Figure 7).

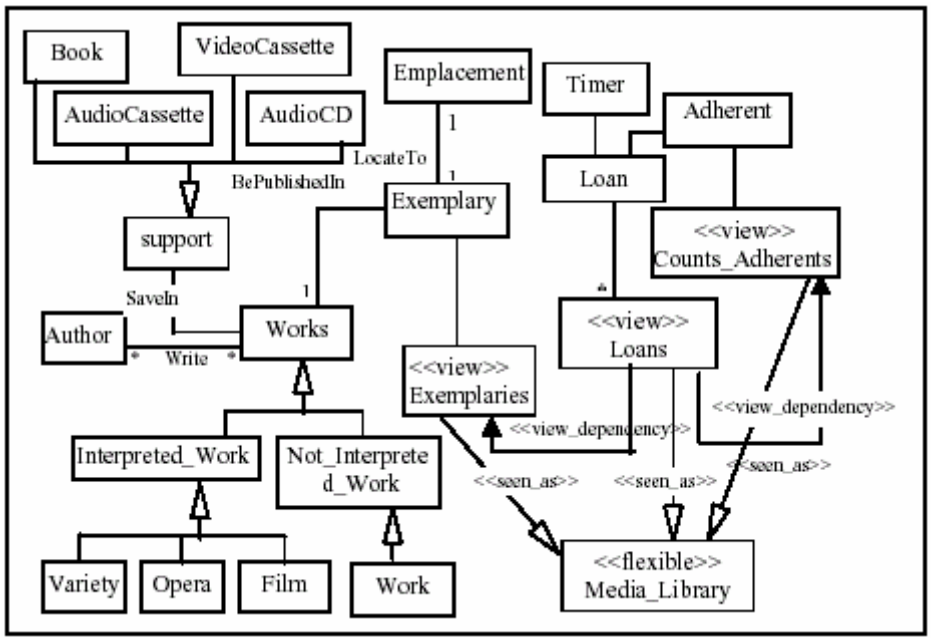


Figure 7: Class diagram of MEDIA LIBRARY system

4. CONCLUSION

In this paper, we have argued for a UML profile for viewpoint-oriented software development. We have proposed the Analysis and Design Profiles, which are based on U_VBOOM method. These profiles extend the standard UML by incorporating new concepts like views, viewpoints, flexible, visibility relationships, etc. to object-oriented technology.

In VOM, establishing sub-models is important as it makes global model design easier. Moreover, the sub-models melting are an efficient and automatic.

The Rational Rose is used to support the proposed extension. Rational Rose provides several ways for extending and customizing its capabilities to meet specific software development needs. A detailed discussion of Rational Rose Extensibility Interface can be found in [18]. We used Rose Script to generate a VBOOL skeleton for the class diagram built using our proposed extension.

The work describes herein is part of a project which define a methodology of development of components multi-views objects. Among the tasks remaining to achieve in this project, we can mention:

- the definition of the composing multi-views notion as regrouping of multi-views classes,
- the development of design pattern supporting the viewpoints approach.

REFERENCES

[1] Ainsworth M, Cruickshank A H, Groves L J, Wallis P J L. Viewpoint specification and Z. Information and Software Technology 1994; 36(1):43-51.
[2] Carré B, Geib J. The Point of View notion for Multiple Inheritance. In Proceedings of the ECOOP/OOPSLA, 1991.
[3] Coulette B, Kriouile A, Marcaillou S. Viewpoints approach in the object-oriented development of complex systems. Object Review 1996; 2(4): 13-20.
[4] Dano B, Briand, H, Barbier F. An Approach Based on the Concept of Use Cases to Produce Dynamic Object-Oriented Specifications. In Proceedings of the Third IEEE International Symposium on Requirements Engineering 1997.
[5] Filkelstein A, Gabbay D, Hunter A, Kramer J, Nuseibeh B. Viewpoints: A Framework for Integrating Multiple Perspectives in System Development. International Journal of Software Engineering and Knowledge Engineering, 1993; 2(1):31-58.
[6] Hair A: Analysis and design process based on the viewpoint concept. RITA - Revista de Informática Teorica e Aplicada -, vol. X n.2, 2004, pp 63-75.
[7] Hair A, Sbihi B, Ettalbi A: Object-oriented modeling by viewpoint using UML ; AMO - Advanced Modeling and Optimisation - journal , vol. 5 n. 2, 2003, pp 107-115.
[8] Harrison W, Ossher H, Kaplan M, Katz A. Subject-oriented programming: a critique of pure objects. In Proceedings of OOPSLA 1993; 28(10): 411-428.
[9] Jacobson I, Christerson M, Jonhson P, Overgaard G. Object-Oriented Software Engineering, A Use Case Driven Approach, Addison-Wesley, 1992.
[10] Jacobson I, Booch G, Rumbaugh J. The unified software development process. Addison-wesley, 1999.
[11] Kriouile A. VBOOM: Analysis and Design object method based on par objet on the viewpoint concept. Ph.D. thesis faculty of sciences, Rabat 1995.
[12] Krutchen P. The Rational Unified Process - An Introduction. Addison-Wesley 2003.
[13] Lopez N, Migueis J, Pichon E. To integrate UML in your projects. Edition Eyrolles, 1998.
[14] Marcaillou S. , Integration of the viewpoint concept in the object modeling; The VBOOL language. Thesis of Paul Sabatier university, Toulouse 1995.
[15] Mili H, Dargham J, Mili A, Cherkaoui O, Godin R.View programming of OO applications. In Proceedings of TOOLS 1999.
[16] OMG White Paper on the Profile mechanism V1.0. OMG Document ad/99-04-07.
[17] OMG, UML Profile for CORBA Specification V1.0, OMG Document FormaI/02-04-01, 2001
[18] Rational Rose 2003. Rose Extensibility User's Guide. Copyright © 2003 Rational Software Corporation.
[19] Rumbaugh J, Jacobson I, Booch G, The Unified Modeling Language Reference Manual, Second Edition. Addison-Wesley, 2004.
[20] Shilling J, Sweeny P. Three Steps to Views. In Proceedings of OOPSLA 1989; 1:353-361.

[21]Sinan S.A., Extending UML. Distributed Computing (1998); (1):28-32.

[22]Slic B, Gullekson G, War P. Real-Time OO Modeling. John Wiley &sons, 1995.

[23]UML Syntax and Semantics Guide V2.0, OMG Document format, http:// www.rational.com/UML, 2004.

[24]Unified Modeling Language, Version 2.0. Object Management Group, http://www.omg.org, 2004.

Received: 22 November 2004
Accepted: 16 December 2005

UDC:004.738.52
Original scientific paper

LOG FILE ANALYSIS AND CREATION OF MORE INTELLIGENT WEB SITES

Mislav Šimunić
Faculty of Tourism and Hospitality Management, Opatija, Rijeka University, Croatia
mislavs@fthm.hr

Željko Hutinski, Mirko Čubrilo
Faculty of Organization and Informatics, Zagreb University, Varaždin, Croatia
{zeljko.hutinski, mirko.cubrilo}@foi.hr

Abstract: *To enable successful performance of any company or business system, both in the world and in the Republic of Croatia, among many problems relating to its operations and particularly to maximum utilization and efficiency of the Internet as a media for running business (especially in terms of marketing), they should make the best possible use of the present-day global trends and advantages of sophisticated technologies and approaches to running a business. Bearing in mind the fact of daily increasing competition and more demanding market, this paper addresses certain scientific and practical contribution to continuous analysis of demand market and adaptation thereto by analyzing the log files and by retroactive effect on the web site. A log file is a carrier of numerous data and indicators that should be used in the best possible way to improve the entire business operations of a company. However, this is not always simple and easy. The web sites differ in size, purpose, and technology used for designing them. For this very reason, the analytic analysis frameworks should be such that can cover any web site and at the same time leave some space for analyzing and investigating the specific characteristic of each web site and provide for its dynamics by analyzing the log file records. Those considerations were a basis for this paper.*

Keywords: *log file analysis, dinamic web site, intelligent web site*

1. LOG FILE CHARACTERISTICS AND IMPORTANCE

The basic characteristic of a log file is a stochastic and unordered record created by the web server containing the data on the web site visitors, who and when visited the web site, and which pages were visited (see Fig. 1). However, the use and analysis of a log file record could be at a much higher level.

Generally, any serious *web host (web service provider in a broader sense)* gives its server logs to Webmasters of certain web sites to analyze their traffic (*or they give already processed files i.e. complete statistics*). Many programs use the log files to analyze the traffic, but there are also aspects of considering the analysis issue in view of raising it to a higher level. What does that mean? The answer to that question should be given throughout

this paper but, to put it briefly, a log file should be used as much as possible for deepening **its analysis in terms of statistics and semantics and for providing for a program controllable web site, which is not an easy task [2], [3]** . In fact, almost all data file analyses come down to the use of current analytic programs (WebTrends, Webalizer) as illustrated in Figure 2, the result of which is daily detailed graphical and tabulated listing of all the events on the server.

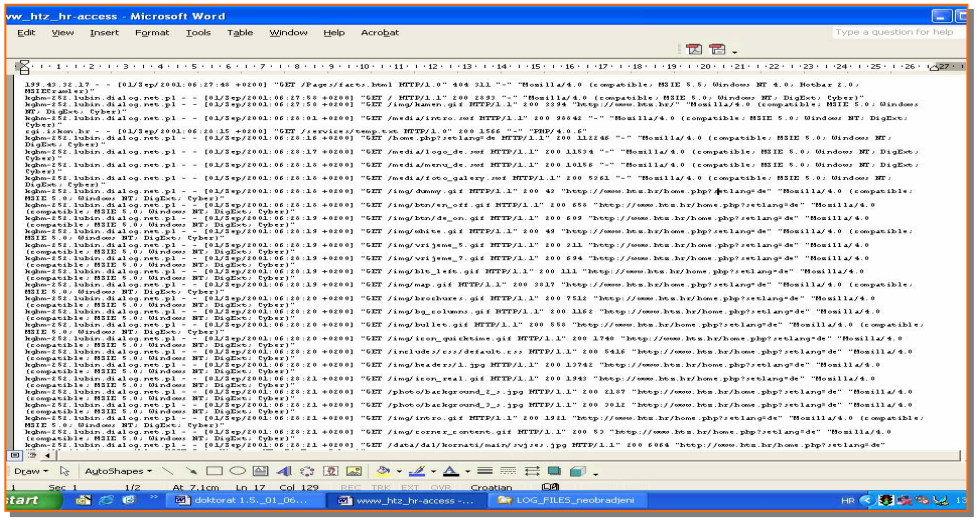


Figure 1: Stochastic and unordered record of a data file

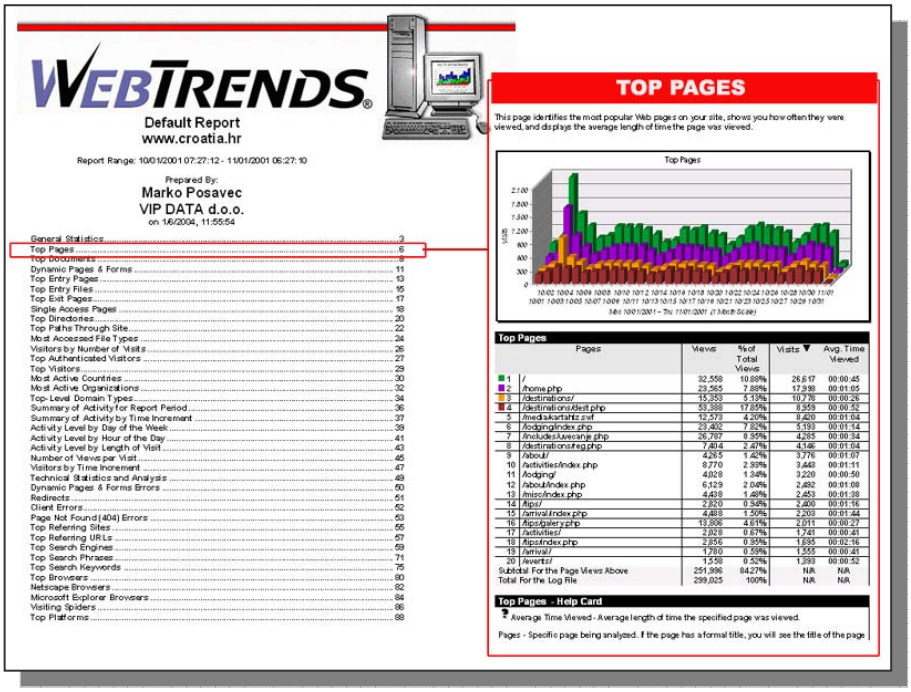


Figure 2: An example of a processed log file (WebTrends Statistics)
Source: "Web trends default report" completed by the authors

However, in most cases this is not the end of the story. Why is it so? We shall try to answer through the following facts:

- Web analyses are very extensive.
- Web analyses are narrow-profiled and adapted to the profession i.e. to the web site type.
- From day to day the world of business is getting more and more dynamic, managers do not have time to spend a few hours every day just for analyzing statistical reports to decide what is next to be done with the web site.
- Initially, the idea of «Web analyses» was good, but with time and development of technology, the analyses have become insufficient.

2. LOG FILE AS A DATA SOURCE AND A BASIS FOR ANALYSIS

Digital environment allows for application of new methods for monitoring and analyzing the usage of digital sources. With adequate software support, any activity or transaction with the server computer system could be automatically analyzed, so-called **log file analysis**. Originally, a log file analysis was intended for server administrators to monitor the system load, but with time, also analysts of different profiles and needs started using it. Basic elements for analyzing the log file contain the answers to the following questions:

- **Who?** – User identification; in most cases IP address of PC from which the user accesses is identified. When the access is controlled by the user's account (username and password), the user and its activities can be personally identified.
- **When?** – Date and time of access;
- **What?** – Identification of web pages used and the method of using the material (*statistical analysis*). There is software to perform standard monitoring but for the needs of this paper, the software is improved and raised to a higher level.
- **How?** – Identification of the way information i.e. web page is used (*semantic analysis*). To perform monitoring, additional and improved software is required.

When collecting the data, the deficiencies and imperfections should not be overlooked but pointed at and this will be done further in this paper:

- The user's activities are recorded by the time the data is received from the database, so we do not know in what way it is further used e.g. do the users read it directly on the screen?
- Data search in the cache memory could be kept for a few days. If within that time the user looks again for the same content, it will be provided from the cache memory and the server will not record that access.
- Do the requests registered come from one or more persons? It is not always possible to see if multiple requests for the same document come from one or more users. Individual user cannot be identified by means of an IP address if there is no fixed connection to the Internet. Even if there is a fixed connection, the access by the same person cannot be systematically monitored if the access is made from more PCs. Similarly, more people can use the same computer

meaning that for different enquiries, the same IP address is recorded and again the data we get on the usage is inaccurate. It should be mentioned that the usage monitoring is possible when a log in is required for using the web site i.e. username and password should be entered.

Consequently, it is suggested that there are realistic theoretic and practical bases for more in-depth improvement of the model design issue aimed at constantly meeting the users' interests to the maximum possible extent [5].

3. STOCHASTIC DATA WITHIN A LOG FILE

Log file as a text (txt) ASCII file is a computer i.e. data representation of a web site use process by a certain number of users within a time interval monitored. The representation changes quickly meaning that we can characterize it as a dynamic change. Dependent on the size and "popularity" of a Web site, the change dynamics can be faster or slower which, of course depends on the number of users surfing on the Web site, its size, user's activities on the Web site, and the time users spend on searching for information about the Web site. Any activity of any user is registered in a log file. No future state of the records in the log file can be predicted upfront because no future interaction of users with the Web site can be predicted either. Therefore, we find the log file almost fully stochastic. Similarly, the log file records are not sorted by any structure already defined. As regards their format, they differ from one server to another. The record in the log file is very much changeable and it is always monitored and stored according to predefined time intervals (hour, day, week, month, etc.), which is generally determined by the web site size and the surfing intensity. The consequence of it is also the very size of the log file and the dynamics of changes taking place in it. To process previously noted records-data (Figure of unordered log file), we need a program that will recognize the log file format, read all necessary data, structure them and store them in an adequate database – the log can be entered directly in the database. All data collected in this way is used for further analyses, which gave rise to making this paper.

4. LOG FILE AS A BASIS FOR PROVIDING FOR WEB SITE DYNAMICS

As previously noted, a log file is a source of stochastic data. Nowadays, there are programs (Webalizer, WebTrends, etc.) intended for analyzing stochastic data collected in a log file. However, the analyses such as these although often intrinsically good are just not efficient enough and in most cases, they are just a set of tables and graphs that remain unused in practice. This was why the author's thoughts were focused on improving the usability of the log file analysis for realization of a program controllable web site. The idea of developing a program controllable web site [7] is illustrated on Figure 3.

As showed on the figure above the novelty is **WLE GUI** (WebLogExplorer Graphical User Interface), an integrated software solution providing both for the data analysis from a log file and the effect on the web site that is the web site dynamics. Basic components of the WLE software solution are a module for data analysis and a module for ensuring the dynamics within the web site. By implementation of the WLE software solution, although currently in an experimental stage, the dynamics within the HTZ (Croatian National Tourist Board) web site has been successfully performed, as showed further in the paper.

Figure 3: Program controllable web site

5. WLE GUI – A SOFTWARE SOLUTION FOR WEB SITE DYNAMICS

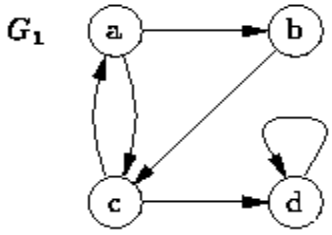
The paper further discusses and describes the role of WLE as a software solution for ensuring the web site dynamics. The WLE software has been developed by a theory of graphs [4]. In fact, the theory of graphs is a mathematical discipline studying the regularity of (mathematical) graphs. The graphs are mathematical structures for which we can say they consist of vertices and edges connecting them. More precisely, among many regularities and definitions the one below can be singled out:

Oriented graph is an ordered pair $G = (V, E)$ of below characteristics:

The first component V is a final non-empty set. The set elements are called the graph vertices.

The second component E is a finite set of ordered pairs of vertices, i.e. $E \subset V \times V$. The elements of that set are called the graph edges.

An example of the graph is given on Figure 5-3. In the graph $G_1 = (V_1, E_1)$ is $V_1 = \{a, b, c, d\}$ and $E_1 = \{(a,b),(a,c),(b,c),(c,a),(c,d),(d,d)\}$.



Source: B. R. Preiss: »Data Structures and Algorithms with Object-Oriented Design Patterns in C++», University of Waterloo, Waterloo, Department of Electrical and Computer Engineering, Canada

Figure 4: Oriented Graph

For presenting the navigation paths, we used an oriented graph like the one showed above: the graph vertices are the web sites, the oriented edges are the links from one web site to another, and the vertices labels resemble the symbol of the navigation path to which a vertice belongs. The oriented graphs we used are somewhat more complex than the usual ones: any vertice can belong to none or to a larger number of navigation paths i.e. every vertice can have no slabels or several labels.

6. WLE CHARACTERISTICS (WEBLOGEXPLORER) – LOG FILE ANALYSIS PROGRAM

Before the characteristics of the WLE software were developed and defined, it was necessary to set the criteria and requirements such as:

- recognition of a log file format to be processed,
- creation of an optional format if none from the program is suitable,
- the fastest possible processing of the log file data and its storage in a database,
- presentation of data to the user in the most intuitive way possible,
- possibility of creation of various analyses within a certain time period,
- relating that data to the dynamic change of the page,
- the simplest possible orders for a dynamic change generally given by a web programmer or design engineer,
- generation of various reports and a printout,
- selection of a software tool that could meet all or the most of above criteria,
- selection of a database

The basic characteristics of the program arise from the criteria and requirements defined before making the program:

- possibility to process the log files according to the formats: IIS server, Apache server or in the case of any other log file format, the format can be free defined according to the tags;
- recording in a centralized database after data processing;
- possibility to get various analyses (statistical and semantic) within a certain time;
- a review and a printout of various reports;
- data filtering according to some special criteria;
- creation of dynamic changes that should happen on the pages selected;
- possibility to create a schedule according to which dynamic changes should take place;
- definition of aliases for optional addresses to facilitate the data representation on the screen or in the report;

However, before making the program, we selected **MS Access** database (because it was sufficient for experimental investigation process; for practical use for more complex models, we recommend MySQL, MSSQL, Oracle, PostreSQL) and the program language **C# from Visual Studio .NET 2003 [6]** that could meet our requirements and criteria.

We have designed DBMS, created the tables, attributes and interrelations. Once we have chosen the software tool and the database into which we shall store the data from the

log file, we made the basic classes for connecting the program and the database. The class for the database consists mostly of functions for storing and reading the data. Also the class for as accurate processing of log as possible has been made taking into account the criteria set.

To create an optional log file format, a specification of tags has been introduced as a support to the WLE software:

- %S server name
- %d day
- %M month
- %Y year
- %h hour
- %m minute
- %s second
- %E method (GET, PUT, POST)
- %e URL path
- %q URL query
- %c server status
- %f host name
- %B browser and OS
- %j any other symbol (space and the like)

Before arranging the tags, one should know the log file format. In addition, a graphical interface has been design as well as other parts of the program to help the user with the program application. After that, it was necessary to create the functions to get necessary statistical analyses within a certain time.

To get a semantic analysis, we developed a class that enables sorting of data processed by servers and defining the navigation paths by which the user visited certain web pages within a web site.

Above was followed by creation of a window and the logics for dynamic change of pages. To define the rules following which the dynamics will take place, we used the data stored and a "substitute code" to be entered to change the page content. With that, we have fulfilled the concept of permanent dynamics within a web site based on the user's activities.

7. ANALYSIS OF DATA STRUCTURED AND EFFECT ON THE WEB SITE

When talking about the analysis of structured data and the retroactive effect on the Web site, we actually start talking about the web site dynamics issue involving the following tasks:

- Processing of the current state of visiting a web site and
- changing the web site (dynamics i.e. retroactive effect)

Processing of the current state is carried out according to the data collected within the period the users' interactions with the web site took place. This data is stored in the log files of web server, whose record can take place in certain time intervals. By its format it differs

from one server to another. The usual intervals are hour, day, week, month, or according to the file size.

The basic log file format according to W3C standard is:

- date and time
- TCP/IP user address
- server name
- method
- web address (link)
- web query (link query)
- server status
- users' host
- (browser and operating system)

Some servers have the possibility to records some other additional parameters such as the beginning and end of user's session, amount of pages received and sent in bytes, etc. Order of the format parameters depends on the web server (Figure 5 - Parameters on ISS Server).

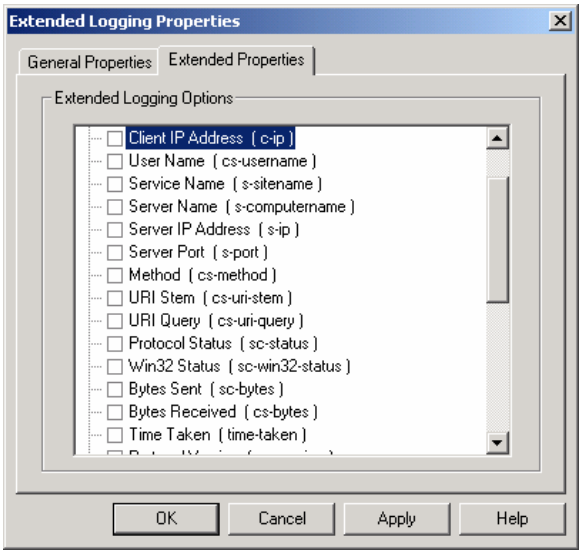


Figure 5: Parameters on IIS Server

To process the data mentioned above, a program is required that will recognize the log file format, read all necessary data, and store it in an appropriate database. All the data collected in this way is used for further analyses on which the web site dynamics is based on.

Change of a web site covers the changes in the content of specific web site pages in line with the current state of users' interactions with the web site. Decisions on modifications are based on the criteria such as web page visiting rates, server's frequency (users'), frequency of navigation parts and the like. This data is obtained from the program mentioned and the database. A qualified person without any specialized programs for dynamic changes could make the changes in the content, or the process could be partially automated by application of such a program. Full automation is restricted by recognition of

different web languages and the logic used to "write" the web pages. Since we wish to perform the dynamics on already existing web sites, the web programmers should properly understand the code and find the places for possible changes. To avoid that, the changes in the page and in the code are performed by specialized **WLE software**.

8. COMPARISON WITH SIMILAR PREVIOUS RESEARCH (FLORID)

The Web provides access to large data sources which are not explicitly organized as databases. Instead, the information is presented as *semistructured* data. In contrast to integrating classical distributed databases, integration of such data raises several new problems such as schema discovery, wrapping and reorganizing the data sources and coping which changes in autonomous sources. FLORID project system has been used for extraction and integration of semistructured data from the Web.

In 1997-1999, the FLORID system has been extended with Web access capabilities (Versions 2.x). A methodology for wrapping and integrating HTML pages by mapping the information in an integrated F-Logic data model representing both the structure of the data sources and containing an application/level model of the information has been developed. HTML pages are wrapped using generic rules for the usual structuring means (i.e., lists, tables, comma-lists, emphasized keywords). In 2000, FLORID has been extended to FloXML with special functionality for handling XML data (XML/DTD parsing, metadata provided by DTDs and XMLSchema, XML export functionality).

FLORID (*F-LO*gic *R*easoning *I*n *D*atabases) is a deductive object-oriented database system employing F-Logic as data definition and query language. The development was supported by the Deutsche Forschungsgemeinschaft (project La 598/3-2). With the increasing interest in *semistructured data*, Florid has been extended for handling semistructured data in the context of Information Integration from the Web. The Experiences with F-Logic and FLORID have been continued with the LoPiX project for XML. Languages supporting deduction and object-orientation seem particularly promising for querying and reasoning about structure and contents of the Web, and for the integration of information from heterogeneous sources. FLORID, an implementation of the deductive object-oriented language F-logic, has been extended to provide a declarative semantics for querying the Web. This extension allows extraction and restructuring of data from the Web and a seamless integration with local data. Since the functionality of wrappers and mediators is integrated into a single declarative language, the development of advanced applications based on the Web as an information source is significantly simplified.

The folloeing goals served as a main motivation for the research in the FLORID:

Querying the Web

- (express declaratively how to query navigate on the Web, extract data from Web pages for populating a database (Web data warehousing)

Management of Semistructured Data

- structure is irregular, partial, unknown, implicit in the data
- example: HTML pages
- querying/navigation using general path expressions
- discover structure

Information Integration

- heterogeneous sources with different structure, wrappers, mediators [3].

Some ideas and knowledge from FLORID project and F-logic has been implemented in our research and integrated in the WLE program algoritms [7].

9. EXAMPLE: WLE IMPLEMENTATION ON HTZ WEB SITE

The role of WLE software is multidimensional in terms of recognition of the log file format, its processing allowing different analyses in certain time intervals, and finally giving the order for dynamic change of the web page within a web site.

The first task of the WLE software is to recognize the log file format and its processing. As noted on several occasions earlier in the paper, the log file content is very unordered and stochastic. Similarly, it is very unlikely that its content is repeated twice or more times implying that each time the WLE software is started different results will be obtained i.e. a review of the status and activities on the market demand side reviewed by the log file.

Consequently, a research analysis performed in the context of such consideration would mean to subject a certain log file to the entire WLE process. For the needs of this paper, we have collected the data of the log file of HTZ Web site from July 2003 on (31 off log files). An average daily "log" is c. 600,000 sets, which is a large amount of data and reflects a representative picture of the activities i.e. demand market interests. Because of the daily log size, among the available data (log file), a random selected log file will be processed and parsed to show the stochastic characteristics of the data within the log files. Having obtained the data in this way, the file selected will be subject to the entire WLE process. When the log file selected is decompressed from *.zip file, and moved to the WLE software module defined directory for off line analyses, a parsing process of the same log file should be run to obtain a statistical and semantic data representation of log file records, which is then used to define and run dynamic changes to the web site. Because of the restricted size of this paper, only one simple example of defining dynamics within a web site based on a log file analysis will be given to justify the essence of the issue discussed in the paper. The figure below illustrates the status of HTZ home web page (http://localhost/home.php). Attention should be given to the order of items in the menu "Dolazak" ("Arrival")

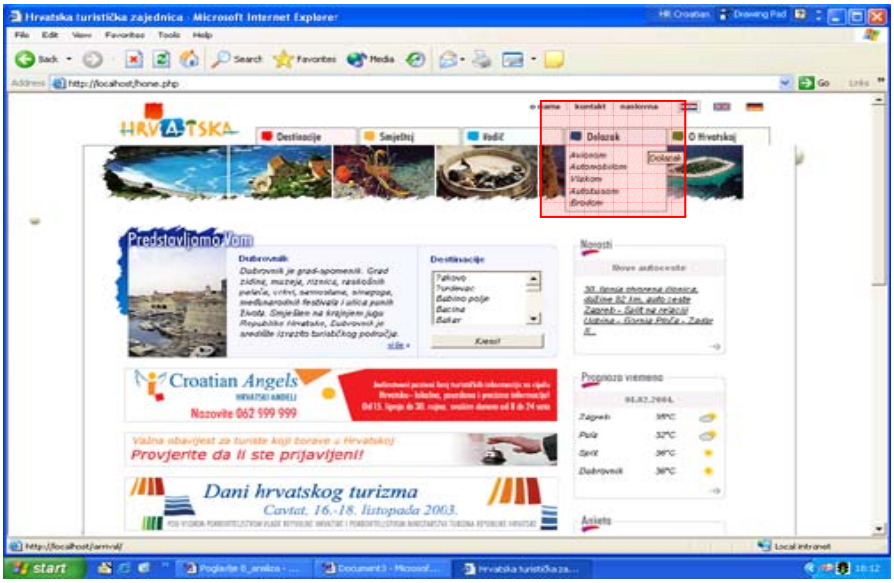


Figure 6: Web page http://localhost/home.php before dynamic changes

We have chosen for the analysis a log file of 4 July 2003 having the characteristics given in the table below:

Table 1: Characteristics of the log file for analysis (4 July 2003) – L4

	Characteristics of parsed log file **(L4)**				
Item	**Analytic process**	**logfile***	**logfiledate***	**logsize***	**logline***
1	**L4**	www_croatia_hr-access_04_07_2003.0	05.07.2003	76 770 373 bytes	364 459

After statistical analysis of the log file selected, below values of the parameters frequency relating to the use of items from the menu "Dolazak" (arrival) have been obtained:

Table 2: L4 –Internal relation of the menu items "DOLAZAK" obtained by the analysis

L4: Internal relation of the menu items DOLAZAK (100 d.n.r)*		
Item	**Web Page Name**	**Quantity**
1	/arrival/index.php?menu=3	51
2	/arrival/index.php?menu=4	155
4	/arrival/index.php?menu=5	0
5	/arrival/index.php?menu=6	0
3	/arrival/index.php?menu=7	66
	TOTAL	**272**

*100 top ranged events

When on the basis of the data obtained the parameters are defined within the WLE software in order to define the dynamic changes to the web site, the layout of the items in the Menu "Dolazak" on the web page http://localhost/home.php looks as follows:

Figure 7: **Web page http://localhost/home.php, after dynamic change**

10. CONCLUSIONS

A log file is an important source of data whose basic characteristics reveal a stochastic and unpredictable record created by the web server and contains the data on the web page use processes within a web site. Worldwide, there are software packages for analyzing log files (WebTrends, Webalizer, etc.) and the methods for monitoring log file records (Clickstream). However, their main feature is generating reports. The authors wanted to make a scientific, theoretic, and practical contribution to development of a new more sophisticated software solution (WLE) providing a program controllable web site i.e. web page dynamics based on the users' activities on the web site thus increasing the log file usability to a much higher level.

Based on the facts presented, it is suggested that there is every reason for defining the controllable dynamics web sites including also further development of the WLE software. The authors are fully aware of many shortcomings of the WLE software but for the purpose of this paper, the intention was to explain the standpoint and define the frameworks within which the WLE exists and functions. For the business systems, the acceptance of the idea of program controllable dynamics of web sites based on the users' activities would mean quicker and better global communication and exchange of information by the Internet thus considerably contributing to raising the company's performance and competition to a higher level. Such and similar segments of business operations are vital for the prosperity of companies, business systems, and the country's economy as a whole.

REFERENCES

[1] E. G. Abels, M.D. White, K.Hahn:"*A user based design process for Web Sites*", Internet Research: Electronic Networking Applications and Policy, vol.8, n.1, 1998

[2] Jane Greenberg, Stuart Sutton, D Grant Campbell*:* "*Metadata: A fundamental component of the Semantic Web*", Bulletin of the American Society for Information Science and Technology. Silver Spring: Apr/May 2003, Volume 29, Issue 4

[3] W. May: "*Information Extraction from the web with Florid*", Dagstuhl Seminar, Declarative Data on the Web, September 1999

[4] B. R. Preiss: "*Data Structures and Algorithms with Object-Oriented Design Patterns in C++*", University of Waterloo, Waterloo, Department of Electrical and Computer Engineering, Canada

[5] M. Randić, M. Šimunić, P. Knežević: "*Modelling Structure and Dynamic Behaviour of Web Pages Augmented with Dynamic HTML*", DAAAM Symposium, Vienna, 1998. pp. 413-414.

[6] M. Stiefel, R.J. Oberg: "*Application Development Using C# and .NET*", Prentice Hall PTR, NJ 07458, 2002

[7] M. Šimunić: "*A Model of Double Dynamic Web Pages*", Faculty of Organization and Informatics, Varaždin, 2004 – Ph Theses

Received: 06 June 2005
Accepted: 05 December 2005

UDC:004.738.52
Original scientific paper

WEB LOG EXPLORER – CONTROL OF MULTIDIMENSIONAL DYNAMICS OF WEB PAGES

Mislav Šimunić
Faculty of Tourism and Hospitality Management, Opatija, Rijeka University, Croatia
mislavs@fthm.hr

Željko Hutinski, Mirko Čubrilo
Faculty of Organization and Informatics,
Pavlinska 2, Varaždin, Croatia
{zeljko.hutinski, mirko.cubrilo}@foi.hr

Abstract: *Demand markets dictate and pose increasingly more requirements to the supply market that are not easily satisfied. The supply market presenting its web pages to the demand market should find the best and quickest ways to respond promptly to the changes dictated by the demand market. The question is how to do that in the most efficient and quickest way. The data on the usage of web pages on a specific web site are recorded in a log file. The data in a log file are stochastic and unordered and require systematic monitoring, categorization, analyses, and weighing. From the data processed in this way, it is necessary to single out and sort the data by their importance that would be a basis for a continuous generation of dynamics/changes to the web site pages in line with the criterion chosen. To perform those tasks successfully, a new software solution is required. For that purpose, the authors have developed the first version of the WLE (WebLogExplorer) software solution, which is actually realization of web page multidimensionality and the web site as a whole. The WebLogExplorer enables statistical and semantic analysis of a log file and on the basis thereof, multidimensional control of the web page dynamics. The experimental part of the work was done within the web site of HTZ (Croatian National Tourist Board) being the main portal of the global tourist supply in the Republic of Croatia (on average, daily "log" consists of c. 600,000* sets, average size of log file is 127 Mb, and c. 7000-8000 daily visitors on the web site).

Keywords: *market, competition, log file, web page, dynamics, WLE-WebLogExplorer.*

1. INTRODUCTION

Global trends are increasingly more dynamic in terms of quicker and better-defined market demands the consequence of which is a demand dictating faster pace of changes on the supply side. Consequently, the supply has to find ways to respond to the demands of the market in the quickest possible way. However, there is still a question, how to do that in the best and quickest way. When a tourist city/town, tourist destination, or even the State publishes some advertising material, it is very difficult and almost impossible to publish the

new ones (their updates) in the same year or change their content because it is a time consuming and demanding process. Similarly, certain time should elapse, end of the season for instance, during which different professional and other organizations (institutes, faculties, etc.) carry out all kind of polls, surveys, and analyses, which again require a certain time to process them and to respond to the market supply in a methodological way. It is important to notice that a process as such requires quite some time from the "establishment of state/facts to the response". However, talking about the Internet and business running by the Internet, the entire issue should be approached in a completely different way. The purpose of this paper is to point out, at a scientific, theoretical, and practical level, that there is no Web site model that would allow permanent and direct retroactive effect on the web site along with a continuous statistical and semantic analysis of the log files. Monitoring and analyzing the amount of people coming to the site by means of the log file analysis make the response time shorter, give better quality and more systematic response to the market demand, the demand market requirements are continuously program monitored, and a continuous web site dynamics is ensured. With the work on the WLE software solution design, which can be implemented in already existing web site, the authors wanted to give a certain scientific and practical contribution to deepening the issue discussed.

2. WEBLOGEXPLORER (WLE) – SOFTWARE CHARACTERISTICS

Before determining and developing the module and algorithms of the WLE software, the criteria and requirements for development of WLE should be established, such as:

- recognition of a log file format to be processed,
- creation of an optional format if none from the program is suitable,
- the fastest possible processing of the log file data and its storage in a database,
- presentation of data to the user in the most intuitive way possible,
- possibility of creation of various analyses within a certain time period,
- relating those data to the dynamic change of the page,
- the simplest possible orders for a dynamic change generally given by a web programmer or design engineer,
- generation of various reports and a printout,
- selection of a software tool that could meet all or the most of above criteria,
- selection of a database

The basic characteristics of the program arise from the criteria and requirements defined before making the program:

- possibility to process the log files according to the formats: IIS server, Apache server or in the case of any other log file format, the format can be free defined according to the tags;
- recording in a centralized database;
- recording in a centralized database after data processing;
- possibility to get various analyses (statistical and semantic) within a certain time;

- a review and a printout of various reports;
- data filtering according to some special criteria;
- creation of dynamic changes that should happen on the pages selected;
- possibility to create a schedule according to which dynamic changes should take place;
- definition of aliases for optional addresses to facilitate the data representation on the screen or in the report;

2.1. ADVANTAGES, DRAWBACKS AND PROSPECTS

The specific purpose of WLE is permanent fulfilment of current requirements of demand market that is of end users. One of the main characteristics of WLE software is the possibility to perform as automatic as possible the dynamic changes in the web page content according to previously described rules. However, since WLE is still in the development stage, its advantages, drawbacks and prospects are discussed further in the paper.

2.1.1. Advantages

The advantages of the WLE software are supported by the facts noted below:

- Program application for log file processing with no changes in the Web site code meaning that we need to install the software, set its basic parameters and start the processing, which is in contrast to FLORID [2] model *(**F-LO**gic **R**easoning **I**n **D**atabases – a deductive object-oriented system of database using F-Logics as a language for data definition and as a Query Language)* whose object model should be installed in every web site page if we wish to get the data we need;
- Simple definition of optional formats if none of the formats suggested by the software is applicable;
- Log file processing and storing into a database every "n" lines thus reducing the possibility of error or the software "breakdown";
- Possibility to analyze the data within a certain period;
- Dynamic change of the content according to the analysis data within a certain time;

2.1.2. Drawbacks

The drawbacks in application of WLE software (which we shall try to remedy during our further work on the program development and its improvement) could be summed up as follows:

- Slightly lower accuracy of the data processed from the log file in contrast to the FLORID model, which has its object model installed in the web pages for which we wish to get certain analyses.
- Data is processed a bit later than in the case of the FLORID model, which immediately stores the data in the database from which it performs further analyses.

- Reduced accuracy of processing and analyzing if most users in a day log surfed on the web site from an institution in which case only one TCP/IP address (proxy) is listed thus making impossible to identify the number of people actually «surfing». This is also the main problem of all other similar software.
- Depending on the web servers it is not possible to include recording of the beginning or end of session for some extensions of pages (this is possible on ISS server with a certain extension or if the pages are *.asp or *.html) thus reducing the accuracy of the users' navigation maps and paths.
- It would be ideal if the dynamic change of pages could be made fully automatic.
- Some servers have the possibility to record the log files directly in the database with some program extension that should be purchased. In this case, the program could be adapted to that database thus making the data processing much faster (there are already some program extensions for IIS server).

2.1.3. Prospects

Briefly, the prospects of WLE software could be described as:

- Improvement in accuracy of data processing;
- Introduction of new analyses oriented to end user's interests;
- Improvement in dynamic changes of the page content according to the specific nature of the Web site for which the software is designed and/or adapted;

3. SOFTWARE DEVELOPMENT PROCESS FOR LOG FILE ANALYSIS

Before making the program, we selected MS Access database (because it was sufficient for experimental investigation process; for practical use for more complex models, we recommend MySQL, MSSQL, Oracle, PostreSQL) and the program language C# from Visual Studio .NET 2003 [4] that could meet our requirements and criteria. We have designed DBMS, created the tables, attributes and interrelations. Once we have chosen the software tool and the database into which we shall store the data from the log file, we made the basic classes for connecting the program and the database. The class for the database consists mostly of functions for storing and reading the data. Also the class for as accurate processing of log as possible has been made taking into account the criteria set. To create an optional log file format, a specification of tags has been introduced as a support to the WLE software:

- %S server name
- %d day
- %M month
- %Y year
- %h hour
- %m minute
- %s second
- %E method (GET, PUT, POST)

- %e URL path
- %q URL query
- %c server status
- %f host name
- %B browser and OS
- %j any other symbol (space and the like)

Before arranging the tags, one should know the log file format. In addition, a graphical interface has been designed as well as other parts of the program to help the user with the program application. After that, it was necessary to create the functions to get necessary statistical analyses within a certain time. To get a semantic analysis, we developed a class that enables sorting the data processed by servers and defining the navigation paths by which a user visited certain web pages within a web site.

Above was followed by creation of a window and the logics for dynamic change of pages. To define the rules following which the dynamics will take place, we used the data stored and a "substitute code" to be entered to change the page content.

4. ANALYSIS OF STRUCTURED DATA AND EFFECT ON THE WEB SITE

Talking about the analysis of structured data and the retroactive effect on the Web site [1], [3] means talking also about the operability of WLE whose tasks are:

- processing of the current state of visit of a web site and
- changing the web site (retroactive effect)

Processing of the current state is carried out according to the data collected within the period the users' interactions with the web site took place. This data is stored in the web server log files, whose record can take place in certain time intervals. By its format it differs from one server to another. The usual intervals are hour, day, week, month, or according to the file size.

The basic log file format according to W3C standard [6] is:

- date and time
- TCP/IP user address
- server name
- method
- web address (link)
- web query (link query)
- server status
- users' host
- (browser and operating system)

Change of a web site covers the changes of the content of specific web site pages in line with the current state of users' interactions with the web site. Decisions on modifications are based on the criteria such as web page visiting rates, server's frequency (users'), frequency of navigation parts and the like. This data are obtained from the program mentioned and the database. A qualified person without any specialized programs for dynamic changes can make the changes in the content, or the process could be partially automated by application of such a program. Full automation is restricted by recognition of different web languages and the logic used to "write" the web pages. Since we wish to perform the dynamics on already existing web sites, the web programmers should properly understand the code and find the places for possible changes. To avoid that, the changes in the page and in the code are performed by specialized **WLE software** [5].

5. WLE – DESCRIPTION OF SOFTWARE COMPONENTS

The software is divided into several wholes:

- Main software (Figure 1)
- Report overview and printing
- Parameter setting
- Dynamic change

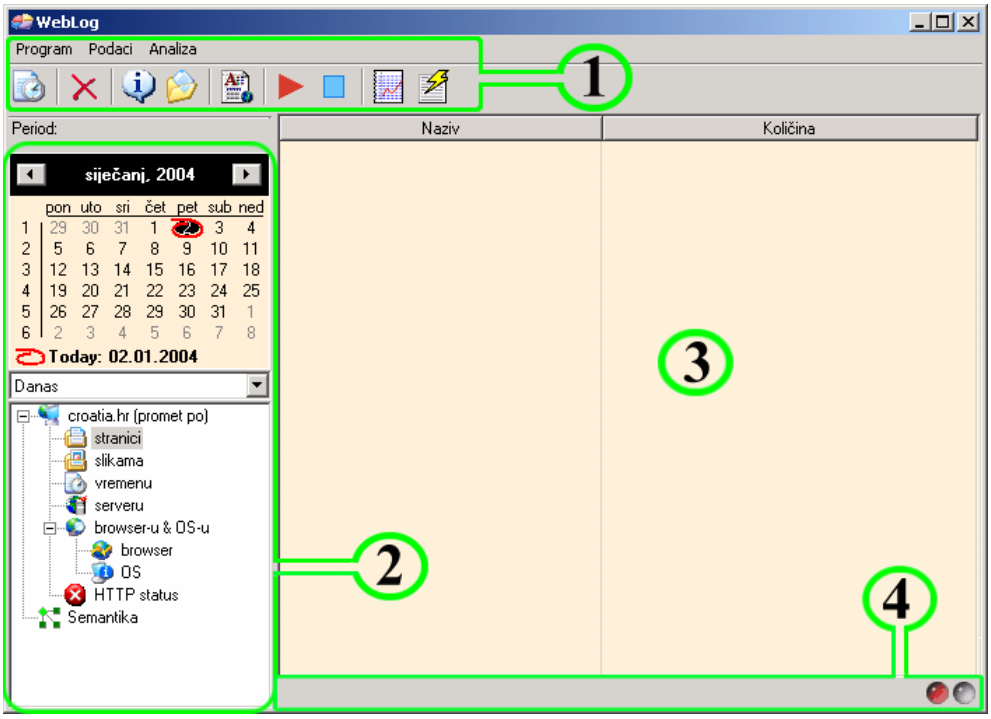


Figure 1: WLE – Main software (interface)

5.1. WLE Software – main part

The main part of the software consists of:

1. Menu and toolbar with icons
2. Left panel (containing a calendar, filter and control with the areas that can be analyzed)
3. Right panel (check list for data presentation)
4. Status bar (showing if data processing is initialized)

Because of the restricted size of this paper we have not given a detailed description of WLE software, which would considerable exceed its framework and the size.

6. WLE ANALYSIS AND GENERATION OF DYNAMICS

Since the log files obtained from HTZ (Croatian National Tourist Board) for the analysis were stored as *.zip files, to be able to use them and to subject them to a complete analysis process by WLE software modules, we have to decompress them and transfer them to a folder defined by WLE software for off line analysis (Figure 2). All decompressed *.zip files have the same default name (www.croatia.hr-access) so it was necessary to change the names according to the time they refer to (in this particular case, we determined a day), and only after that transfer them to the folder intended for processing. When all previously noted conditions are met, it is possible to perform WLE stochastic analysis of the web site usage, *on HTZ web site in this particular case*.

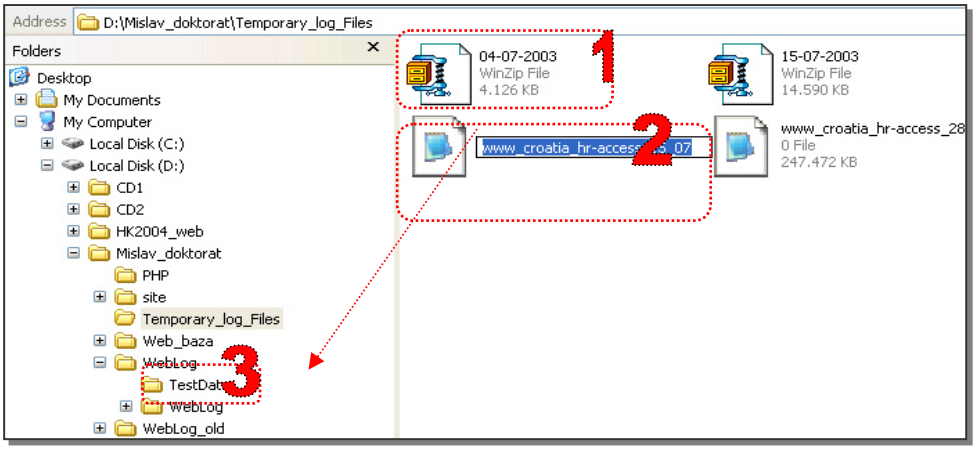


Figure 2: Data preparation (log files) for analysis

At the beginning of the analysis process, the log file should be parsed, providing it is previously decompressed, renamed, and placed in WLE software defined folder for processing. The initialization of the analytic process and the log file processing itself are illustrated on Figures 3 and 4.

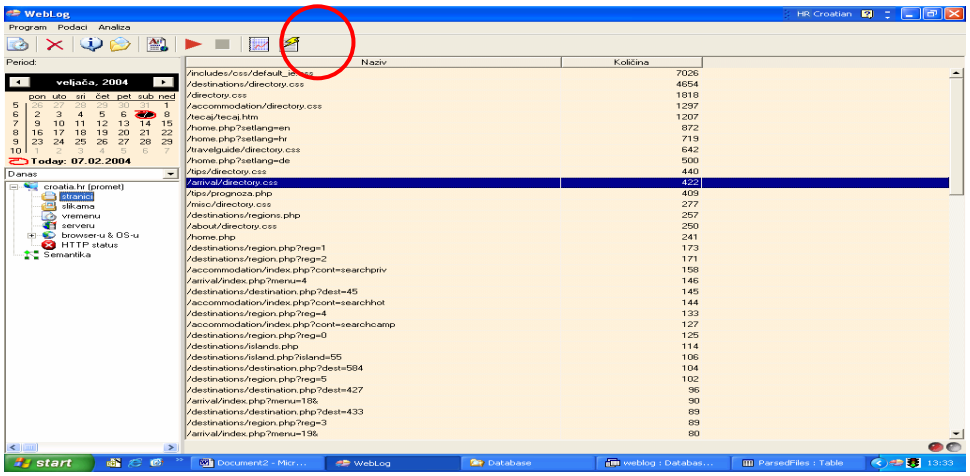


Figure 3: Initialization of analytic process – LOG FILE PARSING

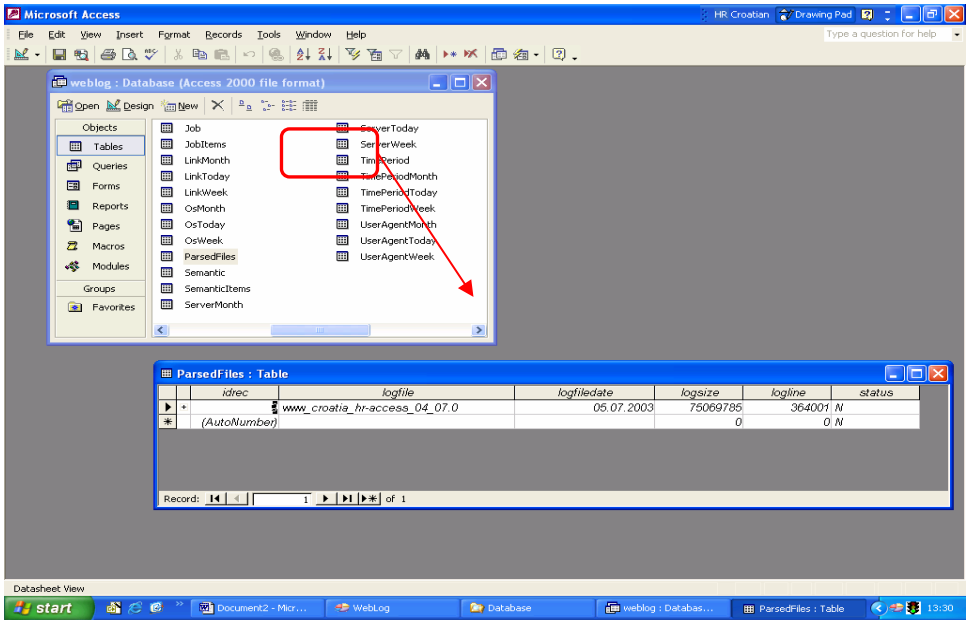


Figure 4: Log file processing by WLE (MS ACCESS Database)

When the log file parsing is completed, we get a statistical and semantic data representation of log file records (Figure 5), which is used for dynamic changes to the web site.

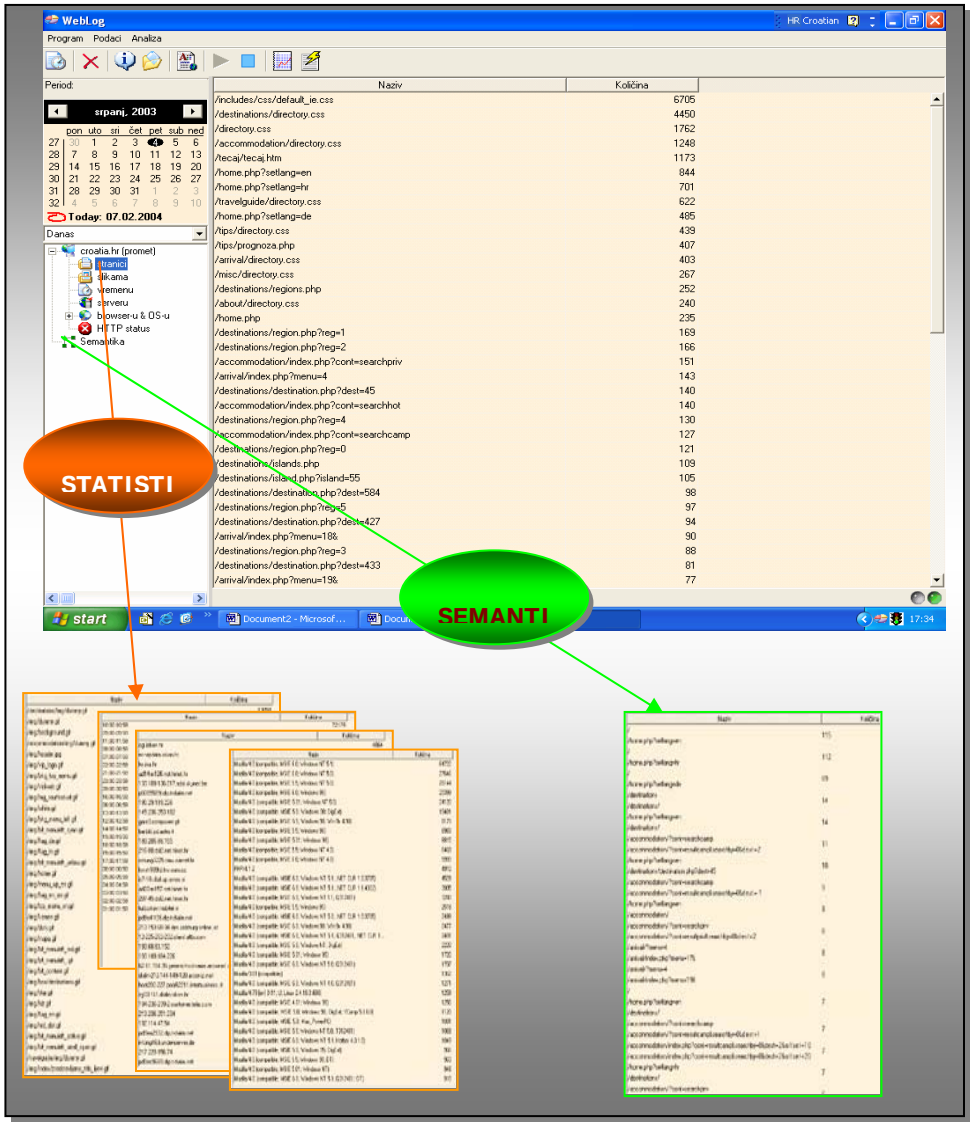


Figure 5: Area of basic events – Statistical and semantic data representation obtained by analyzingthe log file records

For easy understanding, below is a description of a simple example of a dynamic change to the web site based on the data obtained by a statistical and semantic analysis of log files by WLE. Figure 6 illustrates the status of HTZ web site home web (home.php), which will undergo, after the dynamic change, some corrections/changes required by statistical-semantic WLE analysis and generated dynamics.

At this moment, it is important to note the status (order) of the items in Dolazak menu. Below is showed the process of giving commands for generation of changes to the web page selected and the web page look after the commands for dynamic changes.

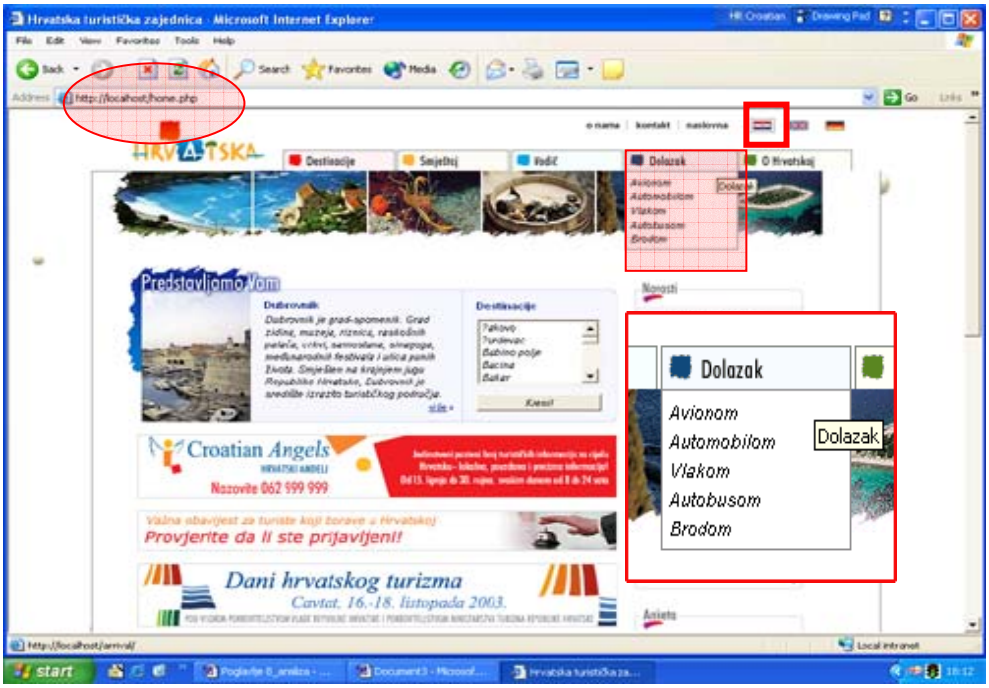


Figure 6: Web page before the dynamic change

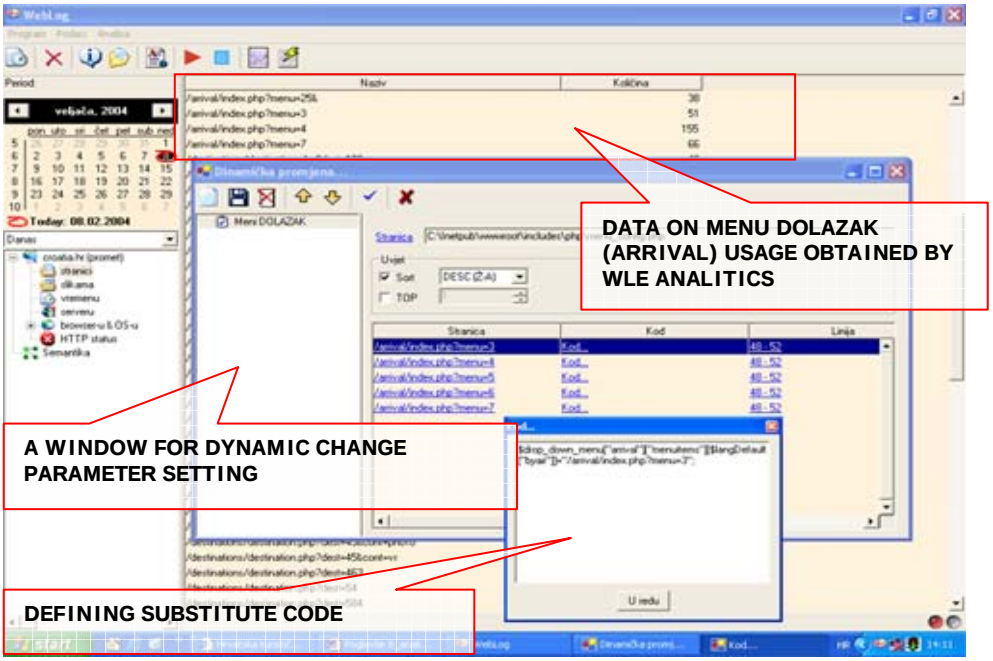


Figure 7: Parameter setting for performing dynamic changes on HTZ web site on the Web page menu_config.php (based on statistical analysis)

After adjustment of parameters for performance of the dynamic changes, the time to perform the change is defined in the main window of WLE software. At the time set, WLE starts the process of dynamic changes and in this particular example the items of the Menu DOLAZAK are rearranged (Figure 8) according to the selected parameter of WLE analytics (Figure 5).

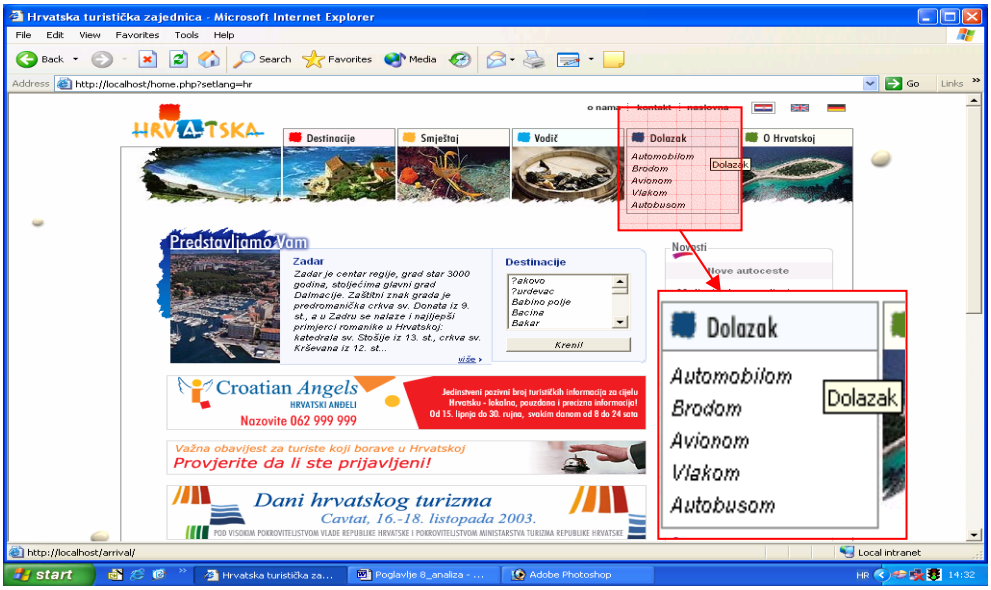


Figure 8: Web site home.php after dynamic change

Above figure (Figure 8) shows a place on the web page subject to WLE dynamic change according to currently relevant parameters. As already noted in the chapter describing in detail the WLE software, the dynamics can be controlled according to the current interest and adapted to the current demand market changes permanently recorded in the log file. In this example, the web page statistics was a basis for defining the changes. A basis for generating the changes can be any sub-analysis described under the WLE Software analysis description.

CONCLUSION

To enable successful performance of any business system and continuous generation of benefits for the supply as a whole and for the supply segments, any company or a business system should use the world trends and advantages of present-day technologies in the best possible way. This paper discusses and explains the application of a new approach to generation of permanent dynamics by implementation of WLE software as an integral part of a web site. Below are basic and concrete conclusions:

- The purpose of the WLE software at this stage was experimental, so for its commercial uses it requires improvements and algorithms that are more sophisticated. This particularly refers to the part involving the mathematical model and analytic algorithm that would enable better connection from the statistical and semantic analysis to the web site.

- WLE is particularly acceptable for transient stages of web sites until a certain business system decides on designing a new web site within which its own object-oriented model is created and incorporated in the web site during its design stage. However, the parer does not address this issue in detail.
- The preliminary analyses and pilot investigation of the WLE software included also *.css files. However, because of their large and dominant frequency of events (which is understandable) which is irrelevant, *.css files can be excluded when implementing the WLE software because they are not relevant indicators in the analysis (they have been excluded from the current version of WLE program WLE 1.0). In most cases, the use of htm, html, asp, and php files is sufficient.
- With regard to the web site size (number of files), technology, the language in which the web site pages were made, it is necessary, after the implementation of WLE program, to define the parameters on which basis the analyses will be made because after data parsing they are a basis for dynamic changes to the web site.
- Special attention should be paid to defining the items that represent the basis of dynamic changes. The dynamic changes themselves, writing and setting of algorithms for their execution, and regularities of their execution should result in cooperation among the experts from the business-problem solving area, informatics, psychology, mathematics, etc. In effect, the cooperation would facilitate development of more sophisticated WLE or similar versions that would base on algorithms of higher sophistication and the results of such an approach would for sure have consequences in better performance of any business system.

REFERENCES

[1] E.G.Abels, M.D.White, K.Hahn: "*A user based design process for Web Sites*", Internet Research: Electronic Networking Applications and Policy, vol. 8, n.1, 1998, p. 39-48

[2] H.Davulcu, J.Freire, M.Kifer, I.V.Ramakrishnan.: "*A layered Architecture for Quering Dynamic Web Content*", Bell Laboratories

[3] H.Davulcu, J.Freire, M.Kifer, I.V.Ramakrishnan:"*A layered Architecture for Querying Dynamic Web Content*", Bell Laboratories

[4] Jane Greenberg, Stuart Sutton, D Grant Campbell*: « Metadata: A fundamental component of the Semantic Web»,* Bulletin of the American Society for Information Science and Technology. Silver Spring: Apr/May 2003. Vol. 29, Iss. 4;

[5] W.May: "*Information Extraction from the web with Florid*", Dagstuhl Seminar, Declarative Data on the Web, September 1999

[6] W. May:"*An Integrated Architecture for Exploring, Wrapping, Mediating, and Restructuring Information from the Web*", ADC 2000, Canberra, Australia, 2000

[7] B. R. Preiss: "*Data Structures and Algorithms with Object-Oriented Design Patterns in C++*", University of Waterloo, Waterloo, Department of Electrical and Computer Engineering, Canada

[8] M.Stiefel, R.J.Oberg: «*Application development using C# and .NET*», Prentice Hall PTR, NJ 07458, 2002

[9] M. Randić, M. Šimunić, P.Knežević: "*Modelling structure and dynamic behaviour of Web pages augmented with dynamic HTML*", DAAAM Symposium, Vienna, 1998. p. 413-414

[10] M. Šimunić: "*A Model of Double Dynamic Web Pages*", Faculty of Organization and Informatics, Varaždin, 2004 – Ph Theses

[11] World Wide Web Consortium (W3C) - http://www.w3.org

Received: 06 June 2005
Accepted: 08 December 2005

UDC:004.657
Original scientific paperr

INDEPENDENT DE-DUPLICATION IN DATA CLEANING [#]

Ajumobi Udechukwu[1], Christie Ezeife[2], Ken Barker[3]
[1,3] Dept. of Computer Science, University of Calgary, Canada
{ajumobiu, barker}@cpsc.ucalgary.ca

[2]School of Computer Science, University of Windsor, Canada
cezeife@cs.uwindsor.ca

Abstract: *Many organizations collect large amounts of data to support their business and decision-making processes. The data originate from a variety of sources that may have inherent data-quality problems. These problems become more pronounced when heterogeneous data sources are integrated (for example, in data warehouses). A major problem that arises from integrating different databases is the existence of duplicates. The challenge of de-duplication is identifying "equivalent" records within the database. Most published research in de-duplication propose techniques that rely heavily on domain knowledge. A few others propose solutions that are partially domain-independent. This paper identifies two levels of domain-independence in de-duplication namely: domain-independence at the attribute level, and domain-independence at the record level. The paper then proposes a positional algorithm that achieves domain-independent de-duplication at the attribute level, and a technique for field weighting by data profiling, which, when used with the positional algorithm, achieves domain-independence at the record level. Experiments show that the proposed techniques achieve more accurate de-duplication than the existing algorithms.*

Keywords: *Data cleaning, De-duplication, data quality, field-matching, record linkage.*

1. INTRODUCTION

Data cleaning deals with detecting and removing errors and inconsistencies from data in order to improve the quality of data [28]. There are data quality problems existing in single data collections, such as files and databases. When these stand-alone sources are integrated (e.g., in data warehouses, federated database systems or global web-based information systems), the data quality problems are inadvertently escalated. Data quality problems (also referred to as "dirty data") could be as a result of different data formats within distinct sources, data entry errors, or the unavailability of common keys among the different data sources to aid integration. Examples of dirty data include the following:

[#] An earlier, abridged version of this work was presented at the 5th International Conference on Enterprise Information Systems (ICEIS) 2003, with the title "Data position and profiling in domain-independent warehouse cleaning".

- Spelling, phonetic and typing errors (e.g., typing "Smit" instead of "Smith", or "Karl" instead of "Carl")
- Word transpositions (e.g., in a free-form text field for address, there could be two representations of the same address as follows: "Sunset Ave., 401" and "401 Sunset Ave.")
- Extra words (e.g., a name field may have the following two representations for the same name: "Miss Alice Smith" and "Alice Smith")
- Missing data (i.e., fields with null entries)
- Inconsistent values (e.g., having a character entry in a number field)
- Multiple values in a single free-form field (e.g., if a field for cities is free-form, there may be entries of the form "Austin Texas" in that field)
- Mis-fielded values (e.g., entering the last-name in the first-name field, and vice versa)
- Illegal values (e.g., having 13 in a month field, or 200 in a student-age field)
- Synonyms and nicknames (e.g., using both "csc" and "compsci" to represent "Computer Science" in different records of the data)
- Abbreviations, truncation and initials (e.g., entering "IBM" for "International Business Machines" or "CalTech" for "California Institute of Technology")
- Data that do not obey functional dependencies and referential integrity

A number of the instances listed above could be solved by using data integrity checks (e.g., having a minimum and a maximum possible data value), and by using more structured schemas (e.g., avoiding free-form fields and null values whenever possible). Others (e.g., spelling, phonetic and typing errors, abbreviations, etc), however, can only be solved by explicit cleaning routines. A major consequence of dirty data is the existence of duplicates (i.e., multiple entries in the database – stand-alone or integrated, referring to the same real world entity). The removal of duplicates constitutes a major cleaning task. Duplicate elimination is also referred to as record linkage or record matching in research literature [35, 36]. The word *"record"* is used to refer to a syntactic representation of some real-world object, such as a tuple in a relational database. The duplicate elimination problem arises whenever records that are not identical, in a bit-by-bit sense, or in a primary key value sense, may refer to the same real life object [21]. Duplicate records may exist in single-source data tables, or may result from the integration of multiple "clean" data tables. Several approaches to de-duplication have been proposed in the research literature. Most of these approaches are tailored to specific application domains (e.g., mailing lists in the USA). Domain-specific techniques rely of in-depth domain knowledge. Such knowledge may not always be available or sufficient for quality data cleaning, thus the need for reliable domain-independent approaches. Domain-independent approaches also result in data cleaning tools that can be easily reused.

1.1 RELATED WORK

Almost all published work on duplicate elimination emphasize specific application domains, and propose algorithms that directly handle particular domains [1, 11, 12, 13, 14, 19, 22, 24, 25, 35, 36,]. For example, some papers discuss duplicate elimination for customer addresses, census records, or variant entries in a lexicon. Other works on

duplicate elimination are not strictly domain specific, but assume that domain-specific knowledge will be supplied by a human for each application domain [17]. Some other techniques aim to improve the domain-specific approaches by using sample labeled sets of duplicates, based on domain knowledge, for training the de-duplication function [2, 16, 32, 33]. Thus, duplicate elimination algorithms vary by the amount of domain specific knowledge that they use.

The work by Monge and Elkan [23, 26] and Monge [20, 21] is quite related to the research presented in this paper. They explore the possibility of cleaning datasets without having a complete knowledge of the source domains. Their work addresses domain-independence at the attribute level, and uses equal weights for all fields at the record level. Such an approach however does not effectively address de-duplication at the record level. The techniques proposed in our work achieve domain-independent de-duplication at both the attribute and record levels. Our technique also achieves better accuracy in de-duplication than earlier approaches.

De-duplication is usually supported by strategies and frameworks aimed at bringing likely duplicates together to ensure that they are compared with each other. A naïve strategy may be to compare each record with every other record in the dataset. The records may also be sorted in some way, then neighboring records compared, as in [3, 27]. Some authors have also proposed alternative strategies to improve the comparison process. For example, the sorted neighborhood method [13], the priority queue [26], and the adaptive filtering technique [10]. These strategies are applicable for both domain-specific and domain-independent approaches to de-duplication, and are not the subject of this paper. Some other researchers on data cleaning have focused on how best to integrate the user into the data cleaning process [5, 6, 7, 8, 9, 29, 30, 34]. The emphasis of the research reported in these works is improving the usability of data cleaning systems. User interaction with data cleaning processes is however not the focus of the work presented in this paper.

1.2 RESEARCH CONTRIBUTIONS

This paper proposes effective techniques to achieve domain-independence in de-duplication at both the attribute and record levels. Earlier approaches to domain-independent de-duplication only address attribute level matching (and require user input for record level matching). Secondly, this paper proposes a field-matching algorithm that incorporates the positional ordering of characters and words. Experiments show that the proposed positional technique improves on the accuracy of match results when compared to existing techniques. The rest of this paper is organized as follows. Section 2 presents the proposed de-duplication scheme. In section 3, the accuracy of the proposed scheme is experimentally evaluated. Conclusions are drawn in section 4.

2. THE DE-DUPLICATION SCHEME

Two levels of domain independence are identified in this work, namely: domain-independence at the attribute level, and domain-independence at the record level. Individual attributes of a record constitute data domains that could be cleaned differently. For example, a domain-specific algorithm could be developed to determine when two entries in the age field are equivalent or differ slightly. This could also be done for the other fields in the record. The work by Hernandez and Stolfo [12, 13] uses this approach. Monge and Elkan [23] and Monge [20] propose algorithms that could be used to determine field matches in two records irrespective of the domain of the field (i.e., the same algorithm can be used for the age field, the address field, the name field, etc). Domain independent approaches also are not affected by the inherent characteristics of the data sources (e.g.,

USA and European addresses can be cleaned with the same techniques without changing parameters). Experiments done by Monge in [20] show that the domain-independent techniques perform as well as the domain-specific techniques in identifying duplicates in the database. The author also experimented the use of classical string matching functions (e.g., edit distance and Smith-Waterman algorithm [31]) for field matching. Classical edit distance functions are however not well suited for field matching in data cleaning because they do not handle acronyms and abbreviations well. For example, the edit distance between "dept" and "department" is 6 (i.e., the number of insertions, deletions, substitutions, etc. required to transform one string into the other). The edit distance between "dept" and "teacher" is also 6, which means that a threshold distance that returns "dept" and "department" as a match will also match "dept" and "teacher". This scenario generally leads to a high level of false matches when classical string matching functions are employed for de-duplication purposes in databases. The experiments by Monge [20] confirm the shortcomings of the classical string matching functions, and show that the recursive algorithm returns more accurate results than earlier techniques. In this paper, we propose a positional algorithm that improves on the results obtained with the recursive algorithm, and also reduces the number of false positives that could have resulted if classical textual similarity functions (such as edit distance) are used.

The second level of domain independence is at the record level. The challenge here is identifying the fields in the database schema that should be given higher preference in determining matching records. For example, the last name attribute could easily be seen as being more important than the gender (sex) attribute in determining if two records in a given database table match. The approach used in commercial data cleaning packages and in most published research, is to have the user or a domain expert specify the levels of importance of the fields in the database. In cases where the packages or algorithms were developed for specific application domains (e.g., names and addresses within the United States), the field importance is hard-coded in the algorithms. In [20], the scheme assigns equal weights to all the fields in the records being compared (except where any of the fields has a null entry, in which case that field is not counted as part of the fields in the records being compared). Such assignment of importance has its obvious shortcomings. Our work proposes a novel, intuitive approach to assign levels of importance (or weights) to the individual fields of the data record for the purpose of de-duplication, thus achieving domain independence at the record level (i.e., the technique will work irrespective of the database schema or the data stored in the data tables). To the best of the authors' knowledge, no previous research had proposed an approach for assigning weights to the fields of the record for the purpose of duplicate elimination.

2.1. POSITIONAL ALGORITHM

The positional algorithm is an enhancement to the recursive algorithm [20, 23, 25]. The basic idea used in the recursive algorithm (with character base) is to locate the characters that make up one entry in a field in the second entry. Thus, if there are two strings "dept" and "department" in two entries within a field, the algorithm checks for the existence of the characters that make up "dept" in "department". For this example, the result is an exact match, which is correct. However, there are instances where this approach fails. For example, given two strings which are entries in the name field of a database:

A = " John Johnson"

B = " Johns Smith"

Using the recursive algorithm with character base, the following steps would be taken to determine if the two strings are duplicates in the database.

1. Tokenize the strings. The resulting tokens are:
   A = {“John” “Johnson”}
   B = {“Johns” “Smith”}
2. Break up each token into its individual atomic characters, as follows:
   A = {“ ‘j’, ‘o’, ‘h’, ‘n’ ” “ ‘j’, ‘o’, ‘h’, ‘n’, ‘s’, ‘o’, ‘n’ ”}
   B = {“ ‘j’, ‘o’, ‘h’, ‘n’, ‘s’ ” “ ‘s’, ‘m’, ‘i’, ‘t’, ‘h’ ”}
3. Take each token in string ‘A’ and compare with each token in string ‘B’. In each case, check for the constituent atomic characters in one token that are contained in the second token. The score is assigned as the ratio of the discovered characters to the length of the word. The highest score achieved for each word in string ‘A’ is taken to be the match score for that word.

   In the example being used, “john” from string ‘A’ is compared to “johns” in string ‘B’ with a score of 1. Also, “john” from string ‘A’ is compared to “smith” in string ‘B’ with a score of 0.25. The higher score, which is 1, is taken as the match score for “john” in string ‘A’. Going by the recursive algorithm, the second token in string ‘A’, which is “johnson”, is compared with all the tokens in string ‘B’. First, “johnson” is compared with “johns”, resulting in a score of 1, then “johnson” is compared with “smith” with a score of 0.29. The higher of the scores, which is 1, is taken as the match score for this token. Taking the average of the match scores for string ‘A’, a match of 1 (exact match) is arrived at in error.

   Such errors can also occur at the word-match level, for example, given two strings:

   A = “Tims” and B = “Smith”

   Going by the recursive algorithm with character base, a match is found if all the characters that constitute a token in one string are found in the second string. In this example, all the individual characters that constitute string ‘B’ can be found in string ‘A’, thus an exact match is declared in error.

The recursive algorithm may also be implemented with a word base. For the word-base case, the first shortcoming discussed above is still present (i.e., when A = {“John” “Johnson”} and B = {“Johns” “Smith”}). The second shortcoming is not encountered because the words are not broken into characters. However, the recursive algorithm with word base is very rigid and does not tolerate errors in words (i.e., permits only exact word-matches and prefixes). The rigidity in the word-base recursive algorithm may be addressed with classical string similarity functions (e.g., edit distance or Smith-Waterman algorithm). However, as discussed earlier in section 2, these classical functions result in high levels of false-positive matches. The positional algorithm overcomes the shortcomings of the recursive algorithm. The positional algorithm tolerates errors in words while maintaining high accuracy levels, and also supports the matching of acronyms and abbreviations. The steps in the positional algorithm are detailed below.

The steps in the positional algorithm (for field and word matching) are outlined below:

Step 1: Tokenize the two strings (or fields) being compared into words. Denote the string with fewer word-tokens as the first string, and the other string as the second.

Step 2: If one of the strings has only one word-token and the other string has more than one word-token (e.g., A = "IBM" and B = {"International", "Business", "Machines"}), then, break up the word-token in the string with one token into its constituent atomic characters (i.e., A = {"I", "B", "M"}). Compare each atomic character in the string with one word-token, with the first character of the word-tokens in the second string in order. Declare a match if all the atomic characters in the first string are matched with the first characters of the tokens in the second string. End if there is a match (i.e., skip steps 3 and 4).

Step 3: Starting with the first word-token in the first string (the first string is assigned as the string with less word-tokens), compare each word-token in the first string to the word-tokens in the second string. (The technique for comparing two words is discussed in step 4 below). If a match is found (i.e., the threshold is reached) between any two word-tokens being compared (i.e., one word-token from each of the two strings), a match score of 1 is assigned to the word-token, the comparison is stopped and the token position in the second string is marked as the last match position and is un-matchable for subsequent matches, and subsequent word-tokens from the first string are compared first with word-tokens in the second string that are positioned to the right of the matched position, and subsequently the rest of the word-tokens if a match is not found right of the last matched position.

Step 4: How to match two words: At the word token level, the match problem is reduced to a search of the constituent characters of the shorter word token in the longer word token. A match score of the word is returned as the total match score for all characters in the shorter word divided by the number of characters in the shorter word. The characters in the shorter string are searched for in the longer string one after the other. If the length of the shorter word is less than or equal to half the length of the longer string, then the first characters of both words must match for the process to continue. Once a match of a character in the first word is found in the second word, the position of the last character that got a match is noted in the second word. For example, to match the words "John" and "Johns", once the 'J' in the first word, "John" matches the 'J' in the second word, "Johns", position 1 in the second word is noted so that the next search for the following character 'o' will begin at position 2 and not start again from the beginning. Thus, the last matched position is used to indicate the start position (the first character right of the matched position) and the end position (the last character left of the matched position) of the next search. The position tracking is also used to charge penalties for characters that appear out of order or with a gap because other non-target characters are separating them from previously matched characters. If a character matches, a score of 1 is added to the match score of the word being searched for. However, a disorder penalty of –1 is charged for each character disorder encountered in the match. For example, in matching the word "tims" with the word "smith", character 't' matches the 4^{th} character of the second word, but the next character 'i' in the first word is found not right of the character 't' in the second word, but, left of it, causing a position disorder penalty of –1 to be charged. The effective score when such a disorder occurs is 0. Finally, if there is a gap between two matches of the constituent characters, a gap penalty of –0.2 is charged in addition, for each gap in the first set of gaps. The penalty charge for each gap doubles every time a new set of gaps is encountered. For example, in matching the two words "dean" and "department", the first two characters 'd' and 'e' in "dean" are matched with a match score of 2 (i.e., 1 + 1). However, the third character 'a' in "dean" is not the

third character in "department" as there is a character gap (the first set for this word) between the last matched character 'e', and 'a' leading to a gap penalty charge of –0.2 for the one gap. Also, the last character 'n' is four characters away from the last matched character 'a' and this is the second set of gaps. Thus, the penalty charged to each of these four gaps is double the previous penalty (in this case –0.2). The total penalty for the four gaps is –0.2 * 2 * 4 = -1.6. This means that the total match score for the matched word "dean" is (4 (from the 4 characters found) + 0 (no disorder penalties charged) + (-0.2 – 1.6) (gap penalties))/4 = 2.2/4 = 0.55. The decision as to whether any match score is considered a match is dependent on a word match threshold assigned by the user. A score lower than match threshold returns a "no match" result for the word with a word match score of 0, the word match score is 1 otherwise. The penalty is set at –0.2 for each character space in the first occurrence of a gap, and is doubled each time a new set of gaps is encountered.

2.1.1 Example Matching of Fields and Words in Database Records

Example 1: Are the string field values A and B given below, from the *Name* field of a database table, duplicates, assuming word and field match thresholds of 0.75?

A = "John Johnson" and B = "Johns Smith"

The positional algorithm would first break each *Name* field value into its constituent word tokens. The field with fewer word tokens is designated the shorter string and each word token of the shorter string is matched with the word tokens in the second string to obtain a match score for the word token. The total match score of the entire field value is finally obtained as the average of the word match scores of its word tokens. The two field values are declared a match only if this field-match score is greater than or equal to a field match threshold. Thus, the positional field match algorithm will call the positional word match function to compare "John" in string A with "Johns" in string B. At the word comparison level, the constituent characters in the shorter word are compared with those of the longer word with match scores assigned, last match position remembered, and disorder as well as gap penalties assigned where necessary. Now, comparing the word "John" with the word "Johns" yields a character match score of 1 for 'J', 1 for 'o', 1 for 'h', and 1 for 'n'. The average word match score for this word then is 4/4 = 1and since this is higher than the word match threshold of 0.75, this is declared a match and a match score of 1 is recorded for the word "John". Since there is a match, all subsequent comparisons would not involve "Johns" in string B. Thus, "Johnson" in string A is never compared with "Johns" in string B. Next, "Johnson" in string A is compared with "Smith" in string B because the last match position is at the first word, which is now declared unmatchable for subsequent comparisons. A match score of 0 is returned for the word "Johnson" because it does not achieve the word match threshold with the word "Smith". This now means that the overall field match score for string A is (1+0)/2 = 0.5. Since this is lower than the field match threshold of 0.75, the decision returned by the positional field matching function for whether the strings A and B are duplicates is false. The differences between the positional algorithm and the recursive algorithm are that (1) position of character and word tokens are remembered and used to disallow re-matching characters/words that had participated in previous matches, (2) positions are used to charge disorder and gap penalties to matched words. These differences improve the accuracy of the positional algorithm.

Example 2: Are the string word values A and B given below, from the *Name* field of a database table, duplicates, assuming a word match threshold of 0.75?

A = "tims" and B = "smith"

The search problem would be to locate the characters of the shorter string in the longer string; thus, the constituent characters in "tims" would be searched for in "smith". The search locates 't' in the 4th position in "smith", a score of 1 is given, 't' is marked as already matched, and the match position recorded. The next character is 'i', and it is searched for in character positions right of 't' in "smith". In this case, there is no 'i' right of 't', so the search wraps around to the first character of "smith". The character 'i' is then located at the 3rd position in "smith", a score of 1 is assigned for matching, but a disorder penalty of –1 is charged because the character is found left of the previous match position of 4. The current match position recorded for further searches (i.e., the match position becomes the last match position referenced to obtain the start position for the next search). The characters 'm' and 's' are also searched for, each yielding a net score of 0 due to positional disorder relative to the most recent match position. Finally, although all the characters in "tims" were located in "smith", only 't' got a score of 1. Thus, total score for the token is 1 divided by the number of characters (i.e., 4) in the token (in this case, the token score is 0.25) indicating a "no match" answer for the two words.

The examples above demonstrate how duplicates are detected at the word and field levels. At the record level, the match-score for any given pair of records is computed as the sum of the products of the field-match scores and the field weights for the selected fields. The sum of the assigned field weights must be equal to 1, and the field-match scores range between 0 and 1, thus the resultant sum for the record-match score would always range between 0 and 1 (where 0 depicts a no-match and 1 depicts a perfect match). The positional algorithm (just like the recursive algorithm) has quadratic time complexities $O(nm)$ at both the word-match level and the field-match level, where n and m are the number of tokens (characters or words respectively) in the strings. At the record-match level, the recursive algorithm also has a quadratic time complexity $O(n^2)$, where n is the number of fields in each record. This is because each field in one record is matched with every field in the second record. The positional algorithm however compares only corresponding entries in the selected fields (with assigned weights); thus, the number of fields in the records being compared does not affect its time complexity. The result is an $O(k)$ time complexity at the record level, where k is the number of fields with assigned weights. Data profiling is used to select the fields that participate in record matching, as well as assign weights (or importance) to the selected fields. Details of the data profiling technique proposed in this work are discussed in section 2.2.

2.2. ALGORITHM FOR ASSIGNING FIELD IMPORTANCE USING DATA PROFILING

Data profiling is simply the characterization of data through the collection and analysis of data content statistics [18]. The scheme used in this work is based on the discriminating power of the attributes of the records in a database to uniquely identify the individual records of the database. The technique gathers data content statistics from a subset of the data table to be cleaned, and uses the information to assign weights to the attributes of the data table. Thus, the technique adapts well to changing data domains. The scheme is outlined in the following steps:

1. Given a database table of N records, where N is a large number, select a subset of the table (n records) and collect the following statistics from the data contained in the

fields of the *n* records: uniqueness, presence of null values, and field length. The major characteristic of interest here is uniqueness. Uniqueness measures the percentage of the data contained in each of the attributes of the n records that are without repetition. In determining the uniqueness factor, each field in the *n* subsets of the database table is grouped into clusters. Two entries are merged into a cluster if they are exact matches or one is a prefix of the other. Also, fields with lengths greater than 100 are grouped into one cluster. The ratio of the number of resulting clusters to the number of records in the subset is the percentage uniqueness (note that one resulting cluster is equivalent to zero uniqueness).

2. After processing the *n*-records subset of the data table, a scheme is used to determine the score of each attribute. This scheme assigns to each field, (1) a uniqueness score between 100 (highly unique) and 0 (not unique), (2) a null value score between -0 (lowest null value presence, no penalty) and -30 (highest null value presence), and (3) a field length score between -0 (short field length) and –30 (long field length with more than 50 characters). The aim is to select fields with a high percentage of unique data entries and low presence of null entries. The scheme also avoids selecting fields with excessive entries (e.g., a memo field). Table 1 shows the scoring scheme used in our work.

3. Given a table with *K* attributes, if all the *K* attributes performed perfectly well in step 2 above, then the total score of all the attributes would be 100**K*. The fields are then ranked in descending order of their scores. The first *l* fields that achieve a total score of (100**K*)/2 (minimum of 2 fields if $K \geq 2$), up to a maximum of 5 fields, are given weights relative to their scores and used in the field-matching algorithm. If all the fields do not achieve a total of at least (100**K*)/2 (i.e., the attributes have a low discriminating score), then the top (*K*/2) + 1 attributes from the ranked list of attributes will be assigned weights relative to their scores and used in the match.

4. If *l* fields are selected from step 3 above, each with a score of *t*, the total score for the selected fields, *T* is then the sum of the t_i's. The weight *w* for each field is then computed as: $w = t/T$

Table 1: Scheme for Scoring Field Weights

Characteristic	Percentage Achieved	Score
Uniqueness	95% - 100%	100
	90% - 94%	90
	80% - 89%	80
	70% - 79%	70
	60% - 69%	60
	50% - 59%	50
	40% - 49%	40
	30% - 39%	30
	20% - 29%	20
	10% - 19%	10
	0% - 9%	0
Null Values	< 5%	- 0
	5% - 10%	- 5
	11% - 20%	- 10
	21% - 30%	- 15
	31% - 40%	- 20
	41% - 50%	- 25

	> 50%	- 30
Field Length (> 50 Characters)	< 5%	- 0
	5% - 10%	- 5
	11% - 20%	- 10
	21% - 30%	- 15
	31% - 40%	- 20
	41% - 50%	- 25
	> 50%	- 30

The major activity in the data-profiling algorithm is the clustering of the entries in each field to determine uniqueness. To determine clusters, each entry in a particular field of the database would be compared to all the previous clusters in that field. In the worst case, all the entries would be unique clusters, resulting in $\frac{n}{2}(n-1)$ comparisons, where n is the number of records in each subset of the data-table selected for profiling. This results in an $O(n^2)$ time complexity for scoring each field in the subset of the database. Given that the data table has k attributes, then since each attribute needs to be scored, the time complexity for scoring each subset of the database would be $O(kn^2)$. Once all the attributes are scored, the other major activity would be the ranking/sorting of the field scores, which has an $O(k \log k)$ time complexity. The dominant cost is thus $O(kn^2)$, and this is the time complexity of profiling each selected subset of the data table. If the algorithm is set up such that the profiling is done m times, then the time complexity for the entire data profiling process would be $O(kmn^2)$. In practical scenarios, n is set as a very small fraction of the entire dataset, such that the profiling phase does not add any significant cost to the entire de-duplication process.

The four steps outlined above must be performed in the first instance before the duplicate elimination process begins. The user can then set an interval after which the data may be profiled again and the scores averaged with the existing scores. The algorithm adapts to the new scores and assigns weights accordingly. For example, given a data table with 100,000 records, we could choose to profile the first 100 records in the first instance. The scores obtained from this first profiling are then used to assign weights to the fields of the database. We can also choose to profile the data after every 20,000 records; thus, in the first pass of the multi-pass merge/purge technique [13] we could profile the data 5 times. Each of the profiling operations updates the scores of the data fields and consequently, the weights and selection of fields that should participate in the matching decision.

3. EXPERIMENTAL EVALUATION

This section presents the results of experimental evaluation of the positional algorithm, recursive algorithm with word-base, and recursive algorithm with character-base. Two data sets were used to test the algorithms for accuracy. The datasets were set up from real-world subscription lists using techniques similar to those used by Hernandez and Stolfo [13] and Monge [20, 21]. The base subscription list consists of 500 unique records, which can generate 124,750 record pairs (i.e., pairing each record with every other record). Subsets of the base list are randomly selected to create the two datasets used in our experiments. Testing for accuracy of de-duplication techniques requires controlled introduction of duplicate records. The errors introduced in the duplicate records vary from small typographical errors to word (and field) transpositions. We draw from popular studies on errors and spelling correction algorithms [4, 15]. The set-up of the experiments and the results achieved for accuracy are discussed in section 3.1.

3.1. EXPERIMENTS FOR ACCURACY

Two datasets were used to test the algorithms for accuracy. The experiments use two measures of accuracy to evaluate the effectiveness of the algorithms in de-duplicating the datasets. The first measure of accuracy used is recall. Recall is the ratio of true matches returned by a matching algorithm to the number of duplicates in the original database. The second measure of accuracy used is precision. Precision measures the ratio of true matches returned by a matching algorithm to the total number of matches returned by the algorithm. Thus, precision gives a measure of the level of false positives in the returned results. The three algorithms were run using threshold values ranging from 0.1 to 1.0. The data tables were set up such that the first column is a unique identifier. This is useful in evaluating the results of the various runs. The results of the experimental runs on the two data sets are discussed in the paragraphs below.

Test Data 1

Test data 1 consists of 7021 pairs of records, with fourteen pairs of duplicated records. The duplicates were set up with acronyms, character substitutions, deletions, insertions, and transpositions, with multiple errors in each duplicated record. There were no exact duplicates in this data set, and each duplicated record was allowed only one duplicate. There were twelve attributes in each record (record id, name, address, city, state/province, postal code, telephone, membership year, level of education, discipline, occupation, and gender). Table 2 presents the results achieved when the three algorithms were run on test data 1. Figures 1 and 2 depict the percentage precision and percentage recall achieved for varying thresholds respectively. Results for test data 1 (Table 2) show that the positional algorithm achieves a higher precision than the other two algorithms for all levels of recall. The highest recall achieved by the recursive algorithm with word base for test data 1 is 87.51%, and its best precision for this threshold is 0.2%. The recursive algorithm with character base achieved a 100% recall, but its best precision at that recall level is 0.22%. Test data 1 was clearly challenging for all three algorithms; however, the positional algorithm outperforms the other two algorithms with respect to accuracy on this dataset.

Table 2: Experimental Results on Test data 1

Threshold	Positional Algorithm		Recursive Algorithm with word-base		Recursive Algorithm with character-base	
	%Recall	%Precision	%Recall	%Precision	%Recall	%Precision
0.1	100	0.20	85.71	0.20	100	0.20
0.2	100	0.20	50	0.14	100	0.20
0.3	100	0.26	50	0.25	100	0.20
0.4	100	2.89	42.86	0.47	100	0.20
0.5	100	15.73	28.57	0.87	100	0.20
0.6	100	50	14.29	2.15	100	0.22
0.7	42.86	37.5	14.29	9.52	92.86	0.29
0.8	14.29	50	14.29	33.33	57.14	0.86
0.9	0	N/A	0	N/A	21.43	25
1.0	0	N/A	0	N/A	0	N/A

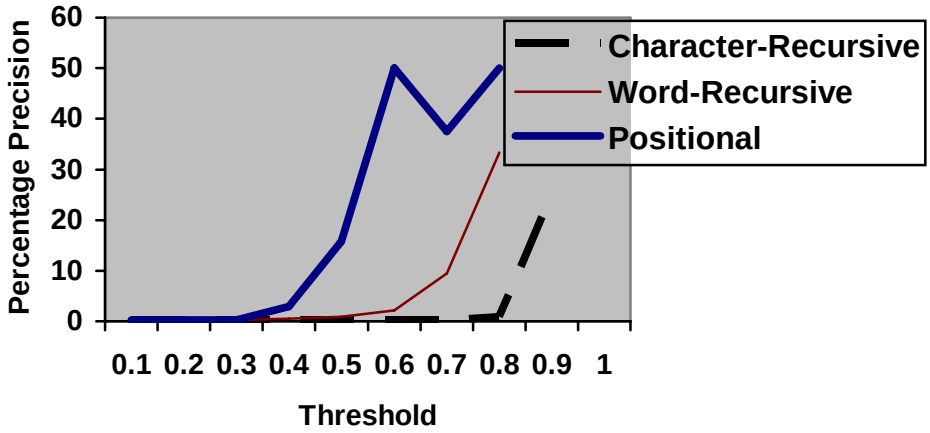


Figure 1: Percentage Precision versus Threshold on Test data 1

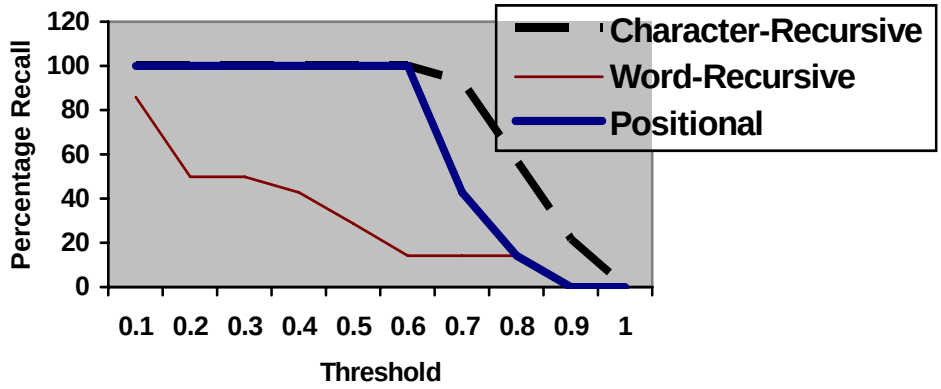


Figure 2: Percentage Recall versus Threshold on Test data 1

Test Data 2

Test data 2 consists of 9045 pairs of records, with fourteen pairs of duplicated records. The duplicates in this case were developed with word transpositions, deletions, substitutions, insertions, as well as exact duplicates, thus it is expected that a hundred percent precision can be achieved at a recall that is higher than zero. Each record in the dataset consists of two fields (record id and name). The record-id field was not used in any of the matching algorithms. Its function was in the evaluation of match results. This data set was set up especially to evaluate the field-matching effectiveness of the various algorithms irrespective of other constraints such as field weights. The results from running the three algorithms on this data set are presented in Table 3. Figures 3 and 4 depict the percentage precision and percentage recall achieved for varying thresholds respectively. The three algorithms performed better with test data 2 than with test data 1. The positional algorithm however, still outperforms the other two algorithms on accuracy, achieving a higher precision for every level of recall. As with test data 1, the recursive algorithm with word

base failed to attain a 100% recall, achieving a maximum recall of 71.43%. The best precision achieved at this level of recall for the recursive algorithm with word base is 6.21%. The recursive algorithm with character base achieved 100% recall levels, but with much lower precision than the other two algorithms. The best precision achieved by the recursive algorithm with character base at 100% recall is 0.47%.

Table 3: Experimental Results on Test data 2

Threshold	Positional Algorithm		Recursive Algorithm with word-base		Recursive Algorithm with character-base	
	%Recall	%Precision	%Recall	%Precision	%Recall	%Precision
0.1	100	0.17	71.43	6.17	100	0.16
0.2	100	0.19	71.43	6.17	100	0.17
0.3	100	0.31	71.43	6.21	100	0.20
0.4	100	0.53	64.29	8.26	100	0.28
0.5	100	1.06	64.29	8.26	100	0.47
0.6	100	77.78	14.29	28.57	85.71	0.92
0.7	100	87.5	7.14	16.67	71.43	2.11
0.8	71.43	100	7.14	16.67	71.43	8.40
0.9	35.71	100	7.14	16.67	42.86	35.29
1.0	21.43	100	7.14	16.67	14.29	33.33

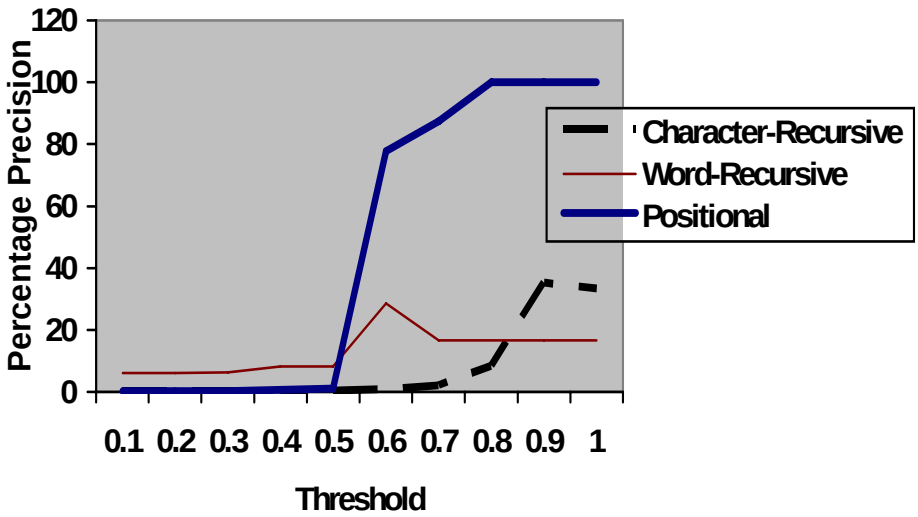


Figure 3: Percentage Precision versus Threshold on Test data 2

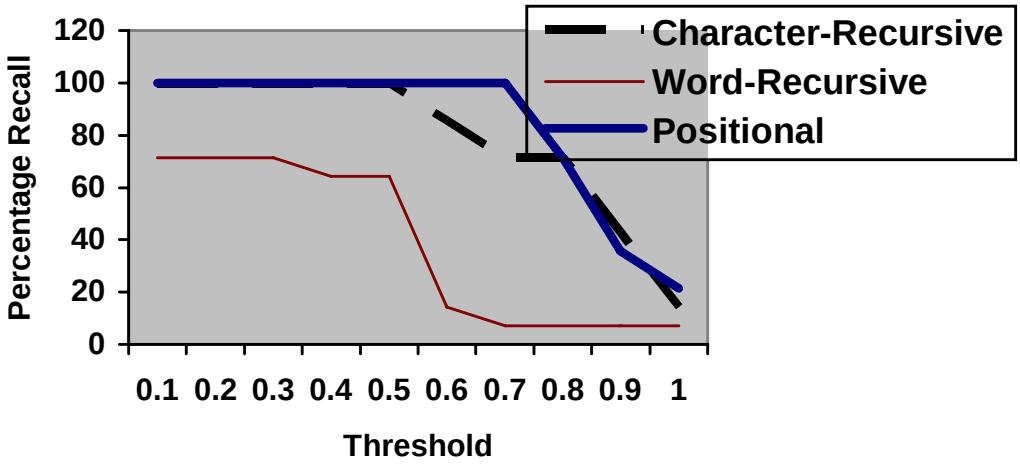


Figure 4: Percentage Recall versus Threshold on Test data 2

4. CONCLUSIONS

This paper contributes a wholesome treatment of domain-independent de-duplication in data cleaning. Domain-independent approaches to data cleaning present an interesting and worthwhile field of research. The domain-independent approaches reduce repetitive efforts required in data cleaning operations, and from a Software Engineering perspective, offer reuse and standardization of data cleaning tools. The technique proposed in this work achieves domain-independence at the field and record levels, and is shown by experiments to generate more accurate results than existing approaches to domain-independent de-duplication.

The concept of domain independence used in this work applies only to data from the unicode character set. An area of future research is to extend the domains covered to include image data, spatial/map data, biological data, *etc*.

REFERENCES

[1] R. Ananthakrishna, S. Chaudhuri, V. Ganti, Eliminating Fuzzy Duplicates in Data Warehouses, Proc. of the 28th VLDB Conference, Hong Kong, China, 2002.

[2] M. Bilenko, R.J. Mooney, Adaptive Duplicate Detection Using Learnable String Similarity Measures, Proc. of the ACM SIGKDD'03, pages 39 – 48, August 2003.

[3] D. Bitton, D.J. DeWitt, Duplicate Record Elimination in Large Data Files, ACM Transactions on Database Systems, 8(2): 255 – 65, 1983

[4] K.W. Church, W.A. Gale, Probability Scoring for Spelling Correction, Statistics and Computing, 1, pp. 93-103, 1991

[5] H. Galhardas, D. Florescu, D. Shasha, E. Simon, Declaratively Cleaning your Data using AJAX, In Journees Bases de Donnees, Oct. 2000.

[6] H. Galhardas, D. Florescu, D. Shasha, E. Simon, AJAX: An Extensible Data Cleaning Tool, Proc. ACM SIGMOD Conference, page 590, 2000.

[7] H. Galhardas, D. Florescu, D. Shasha, E. Simon, An Extensible Framework for Data Cleaning, In Proceedings of the International Conference on Data Engineering

(ICDE), San Diego, CA, pages 312 – 344, 2000

[8] H. Galhardas, D. Florescu, D. Shasha, E. Simon, C. Saita, Improving Data Cleaning Quality using a Data Lineage Facility, In Proc. of the Inter. Workshop on Design and Management of Data Warehouses (DMDW'2001), pages 3 – 16, 2001.

[9] H. Galhardas, D. Florescu, D. Shasha, E. Simon, C. Saita, Declarative Data Cleaning: Language, Model, and Algorithms, Proceedings of the 27th VLDB Conference, Roma, Italy, pages 371 – 380, 2001.

[10] L. Gu, R. Baxter, Adaptive Filtering for Efficient Record Linkage, SIAM Data Mining Conference, 2004.

[11] M. Hernandez, A Generalization of Band Joins and the Merge/Purge Problem, Ph.D. Thesis, Columbia University, 1996.

[12] M. Hernandez, S. Stolfo, The Merge/Purge Problem for Large Databases, in: Proceedings of the ACM SIGMOD International Conference on Management of Data, pages 127 – 138, May 1995.

[13] M.A. Hernandez, S.J. Stolfo, Real-world Data is Dirty: Data Cleansing and The Merge/Purge Problem, Data Mining and Knowledge Discovery 2(1): 9-37, 1998.

[14] J. Hylton, Identifying and Merging Related Bibliographic Records, Master's Thesis, Massachusetts Institute of Technology, June 1996

[15] K. Kukich, Techniques for Automatically Correcting Words in Text, ACM Computing Surveys, 24(4):377-439, 1992.

[16] M.L. Lee, T.W. Ling, W.L. Low, IntelliClean: A Knowledge-Based Intelligent Data Cleaner, In Proc. ACM SIGKDD'2000, pages 290 – 294, Boston, 2000.

[17] M.L. Lee, H. Lu, T.W. Ling, Y.T. Ko, Cleansing Data for Mining and Warehousing, In Proceedings of the 10th International Conference on Database and Expert Systems Applications (DEXA), pages 751 – 760, 1999.

[18] W. Li, Knowledge Gathering and Matching in Heterogeneous Databases, Working Notes of the AAAI Spring Symposium on Information Gathering from Heterogeneous, Distributed Environments, pages 116 – 121, 1995.

[19] J.I. Maletic, A. Marcus, Data Cleansing: Beyond Integrity Analysis, in: Proceeding of The Conference on Information Quality (IQ 2000), pages 200 – 209, October 2000.

[20] A. Monge, Adaptive Detection of Approximately Duplicate Database Records and The Database Integration Approach to Information Discovery, Ph.D. Thesis, Dept. of Comp. Sci. and Eng., Univ. of California, San Diego, 1997

[21] A.E. Monge, Matching Algorithms within a Duplicate Detection System, Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, pages 14 – 20, 2000.

[22] A.E. Monge, C.P. Elkan, WebFind: Automatic Retrieval of Scientific Papers over the World Wide Web, In Working Notes of the Fall Symposium on AI Applications in Knowledge Navigation and Retrieval, page 151, AAAI Press, 1995.

[23] A.E. Monge, C.P. Elkan, The Field Matching Problem: Algorithms and Applications, in: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, pages 267 – 270, 1996.

[24] A.E. Monge, C.P. Elkan, Integrating External Information Sources to guide Worldwide Web Information Retrieval, Technical Report CS96 – 474, Dept. of Computer Science and Engineering, Univ. of California, San Diego, January 1996.

[25] A.E. Monge, C.P. Elkan, The WEBFIND Tool for Finding Scientific Papers over the Worldwide Web, In Proc. of the Third Inter. Congress on Computer Science Research, pages 41 – 46, Tijuana, Mexico, 1996.

[26] A.E. Monge, C.P. Elkan, An Efficient Domain-Independent Algorithm for Detecting Approximately Duplicate Database Records, Proceedings of the SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery, pages 23 – 29, Tucson, Arizona, May 1997.

[27] H.B. Newcombe, J.M. Kennedy, S.J. Axford, A.P. James, Automatic Linkage of Vital Records, Science, Vol. 130: 954 – 959, October 1959.

[28] E. Rahm, H.H. Do, Data Cleaning: Problems and Current Approaches, Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, pages 3 – 13, 2000.

[29] V. Raman, A. Chou, J.M. Hellerstein, Scalable Spreadsheets for Interactive Data Analysis, ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery (DMKD), May 1999.

[30] V. Raman, J.M. Hellerstein, Potters Wheel: An Interactive Framework for Data Cleaning and Transformation, Proc. of the International Conference on Very Large Databases (VLDB), pages 381 – 390, Roma, Sept. 2001.

[31] T.F. Smith, M.S. Waterman, Identification of Common Molecular Subsequences, Journal of Molecular Biology, 147:195 – 197, 1981.

[32] S. Sarawagi, A. Bhamidipaty, A. Kirpal, C. Mouli, ALIAS: An Active Learning led Interactive Deduplication System, Proc. of the 28th VLDB Conf., Hong Kong, China, 2002.

[33] S. Tejada, C.A. Knoblock, S. Minton, Learning Domain-Independent String Transformation Weights for High Accuracy Object Identification, Proc. of the 8th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining, Edmonton, Alberta, 2002.

[34] P. Vassiliadis, Z. Vagena, S. Skiadopoulos, N. Karayannidis, T. Sellis, ARKTOS: A Tool for Data Cleaning and Transformation in Data Warehouse Environments, Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, pages 42 – 47, 2000.

[35] W.E. Winkler, Advanced Methods for Record Linkage, Technical Report, Statistical Research Division, Washington DC: US Bureau of the Census, 1994.

[36] W.E. Winkler, Matching and Record Linkage, In Brenda G. Cox, editor, Business Survey Methods, pages 355 – 384, Wiley, 1995.

Received: 21 October 2004
Accepted: 21 September 2005

UDC:65.012.4
Original scientific paper

ESSENTIAL CHARACTERISTICS OF AN ERP SYSTEM : CONCEPTUALIZATION AND OPERATIONALIZATION[1]

Sylvestre Uwizeyemungu, Louis Raymond
Université du Québec à Trois-Rivières, Canada
{sylvestre_uwizeyemungu, louis_raymond}@uqtr.ca

Abstract: *The interest of firms in ERP systems has been echoed in both the scientific and professional literature. It is worth noting however that while this literature has become increasingly abundant, there does not yet exist an operational definition of the ERP concept that is, if not unanimously, at least widely accepted. This constitutes a handicap for both the research and practice communities. The present study outlines what could be considered as an ERP by first determining the essentially required characteristics of such a system: integration, flexibility and transversality. Indicators are then provided in order to operationalise these three characteristics. The study concludes by proposing a research framework on the impact of an ERP's key characteristics upon the performance of the system in a given organisational setting.*

Keywords: *ERP, enterprise systems, integration, flexibility, process orientation, transversality, information system.*

1. INTRODUCTION

Since their appearance on the packaged software market in the 1990s, ERP (*Enterprise Resources Planning*) systems have grown rapidly, be it in terms of their relative importance in the market [1; 37; 43] or in terms of their adoption by large firms [26] and even by small and medium-sized enterprises [19; 23; 45]. Both the professional and academic literature has shown a great interest in ERP, based on the high hopes placed in these systems, but also on the serious difficulties encountered by firms that have adopted these. Numerous cases of ERP project failure have been documented, including cases leading to the bankruptcy of the adopting organisations [27; 39].

Whatever the organisational impacts of ERP as seen in the literature, one fundamental question remains however: What can be qualified as an enterprise system (ES)? This can seem surprising given the abundant literature on ERP (see [18]'s annotated bibliography for an illustration of this abundance). This question appears nonetheless founded as an analysis of this literature shows that there lacks a widely-accepted operational definition of what is considered to be an ES. Klaus, Rosemann and Gable [28], following a three-level analysis, that is (i) an historical analysis, (ii) a meta-analysis of the representative IS literature on the

[1] An abridged version of this paper was presented at the *Sixth International Conference on Enterprise Information Systems*, Porto, Portugal, 2004.

subject, and (iii) a survey of academic experts, conclude on the existence of wide-ranging perspectives on the ERP phenomenon, and above all on the absence of a commonly accepted definition. It is however essential that there be a consensus in the research community on the definition of the research object, or at least on a set of common indicators. Three reasons for seeking this consensus are given [28]: it facilitates communication among researchers, and between researchers and practitioners, it allows for the development of teaching material on ERP and related concepts in university curricula and in professional training, and it facilitates communication between ERP system vendors, consultants, and their clients.

The present paper aims to contribute to the discussion on the meaning of ERP systems. In order to do this, the next section of the paper focuses on the terminological ambiguity surrounding the term ERP itself. In another section are then examined the characteristics generally attributed to ES in the literature, and a regrouping of these is proposed, in order to refine their analysis. This leads to the identification of three characteristics judged to be indispensable if a system is to be qualified as an ERP system, namely integration, flexibility, and transversality. Indications for the operationalisation of these characteristics are then proposed. Finally, the paper concludes on limitations and research orientations.

2. ERP: A TERMININOLOGICAL AMBIGUITY

In the research literature on the ERP phenomenon, even the term « ERP » itself is not unanimously accepted. However, there is a largely-established consensus [20; 28; 44] on first considering MRP (*Material Requirements Planning*), and then MRP II (*Manufacturing Resources Planning*) systems as the precursors of ERP. Hence, the « ERP » appellation directly follows « MRP II », with the word « enterprise » replacing « manufacturing » to signify that the system extends to the whole of the organisation.

Some object to the ERP label, judging it to be doubly restrictive in that it descends from MRP II and alludes only to planning. Alternative appellations have thus been proposed, including EWS (*Enterprise Wide Systems*), or simply ES (*Enterprise Systems*) [14; 32] to highlight the coverage and integration of all organisational functions, or yet again ERM (*Enterprise Resources Management*) [34, p. 25] to highlight the support for all of the firm's management activities and not only planning.

While these objections are founded, the ERP appellation remains widely used in both the research and professional literature, seemingly for reasons of convenience and antecedence. Being gradually accustomed to its use, people do not seem to be bothered by the weak correspondence between the appellation and its content. One must also note that since it was first coined in 1992 [28], the term ERP has preceded the alternative terms that appeared a few years later. In the rest of this paper, the term in-use will be employed, namely "ERP", whatever its imperfections, with the aim of better defining the content of this term.

3. CHARACTERISTICS GENERALLY ATTRIBUTED TO ERP SYSTEMS

If one refers to various authors [10; 14; 38], an ERP system can be defined as an adaptable and evolutive commercial package that supports, in real time and in an integrated manner, the management of most if not all of a firm's business processes. One can attempt to better define it by looking at its characteristics. In this regard, an attentive observer of both the research and professional literature will denote quite a number of attributes deemed to be possessed by ERP systems.

3.1. IDENTIFICATION OF ERP CHARACTERISTICS

Characteristics generally attributed to ERP systems in the literature are presented in Table 1. In doing so, an attempt has been made to include all characteristics, notwithstanding the different terminologies used by different authors in describing them.

Table 1 : Recapitulation of the main characteristics of ERP systems

Characteristics	Explanatory elements	Authors
Integration	Interconnections between functions and hierarchical levels Interaction between the various processes	[3; 4; 10; 16; 29; 31; 38]
Completeness (generic function)	Wide range of functions Applicable to various types of firms Connectivity with the outside	[10; 16; 29; 38; 44]
Homogenisation	Unique data referential Uniformity of human-machine interfaces Unicity of the system's administration	[10; 16; 29; 38]
Real-time	Real-time update and consultation	[16; 34]
Adaptability (flexibility)	Capability to follow rule and organisation changes (made possible by parametering)	[29; 38; 44]
Openness (evolutionary)	Modularity Portability	[29; 38; 44]
Transversality (process-oriented view)	System designed in regard to the business processes necessary to achieve objectives Focus on value rather than authority flows	[8; 13; 28]
Best practices	System imbeds best practices in the field	[40; 44]
Simulation	Business processes can be simulated	[38; 44]

3.2. REGROUPING ERP CHARACTERISTICS

For a better understanding, ERP characteristics have been regrouped under three dimensions in regards to their nature, namely technical, organisational and informational, as presented in Figure 1. The technical dimension regroups characteristics that refer to the capabilities or facilities for applications development offered by ERP systems in comparison to traditional systems. This includes two basic characteristics: flexibility (adaptability) and openness (evolutionary).

The organisational dimension refers to the system's deployment in the firm. These are the characteristics that best reflect the impact of an ERP system on the organisation, on its structure as well as its practices. This includes integration, completeness (generic function), homogenisation, transversality (process-oriented view) and best practices. The informational dimension regroups characteristics that relate to the quality and usefulness of the information provided by the system, namely real-time (update and consultation) and simulation (of actual business processes).

This regrouping justifies the assertion that ERP systems can be qualified as organisational systems rather than as technical or information systems [8]. Also, existing ERP systems are built upon diverse hardware and software platforms, such as Windows or UNIX. This is an argument for characterising an ERP system more by its functionalities than by its design or technical exigencies [28].

4. ESSENTIAL CHARACTERISTICS OF AN ERP SYSTEM

If one carefully considers the characteristics enumerated above, a number of questions arise. First, are these characteristics truly universal? In other words, do they all apply to all

ERP systems? Second, are all of these characteristics equal, or are some characteristics more essential than others in defining ERP systems? And what would be the indispensable or minimal characteristics required for a system to be qualified as an ERP system? The following discussion attempts to answer these questions.

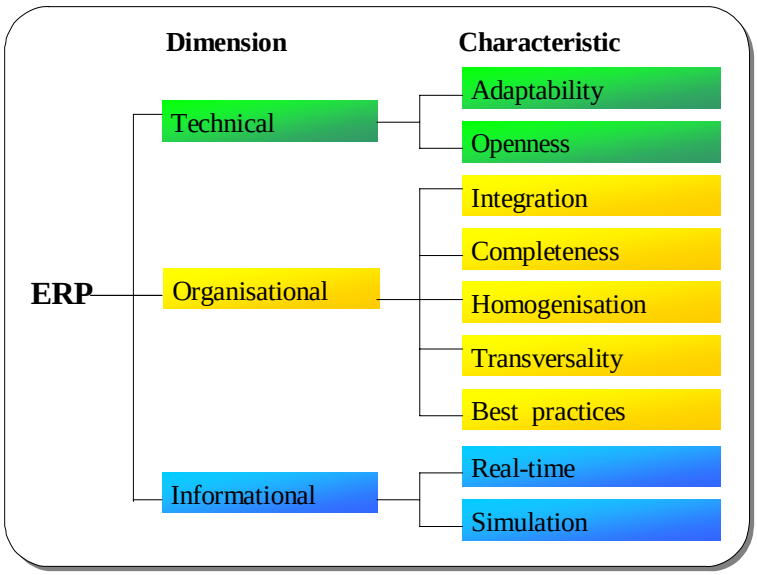


Figure 1: ERP characteristics regrouped under three dimensions

4.1. DISCUSSION OF ERP SYSTEM CHARACTERISTICS

In analysing the characteristics attributed to ERP systems in the literature, one attempts to identify those that are most significant and common to all. A characteristic can then be kept or discarded on the basis of its discriminating power in regard to other characteristics. Redundant attributes can thus be eliminated. The discussion of the characteristics is in the order pictured in Figure 1.

A flexible organisation is one that can use its existing resources and competencies to quickly respond to changing conditions in its environment without significant decreases in performance [15]. The flexibility of the IT infrastructure is a characteristic deemed to be determinant if these technologies are to be at the source of a sustained competitive advantage for the firm [17]. This explains the importance accorded to this issue by IT managers [11]. Flexibility is even more important in the case of ERP systems, given the size of the investment they require and the breadth of their organisational coverage. If an ERP system was not adaptable, it would limit the organisation's development potential in a changing environment.

Also, given that the ERP system integrates various functions of the organisation (production, finance, HRM, etc.), there seems to be both an opposing and a complementary relationship between flexibility and another characteristic, namely integration. On one hand, the more an organisation is integrated, the harder it is to "disconnect" itself [30]. Indeed, it has been found that the more firms adopt integrated technologies, the less flexible they are [9]. On the other hand, integrated processes allow for greater sharing of new information, thus insuring quicker response to changes in the environment and increasing the organisation's flexibility. Hence, even if flexibility appears to contradict integration, the

former criterion completes and relativises the latter, as integration obtained at the expense of flexibility would be problematical.

Openness is a characteristic that appears to be redundant in regard to flexibility. The modularity and portability (openess) that allow an ERP system to evolve with the organisation can be considered as factors of flexibility. In fact, anterior studies [11] include modularity in their definition of flexibility.

Integration is without a doubt the most important ERP characteristic, as all authors concur on this [3; 14; 29; 38]. It distinguishes ERP systems from traditional IS whose « informational fragmentation » has been criticized [12] in that it reflects a vision of the organisation as a set of islands or functional silos that cannot coummunicate with each other, or that communicate little or with great difficulty.

Completeness (the generic functionality) is a characteristic that, pushed to the extreme, becomes unrealistic. A generic system that would work for all types of firms and industries is in fact very difficult if not impossible to design. Forest [20] shows for example that depending upon the nature of their physical flows, manufacturing firms will have IS needs, and ERP needs in particular, that are specific to them. One could in fact develop a typology of ERP systems by taking into account the specificities of the adopting organisations and industries. Whereas Parr and Shanks [35] show that by choosing a specific type of implementation, given their initial motivation for adopting an ERP system, organisations wind up with systems that are not comparable. For their part, Klaus *et al.* [28] note that in its strictly generic form, the ERP system needs to be configured before being used. By adding or eliminating certain elements, the configuration thus creates distinct product types and makes it very difficult to have a standard or common description.

Homogenisation refers to the existence of a unique data referential, the uniformity of human-machine interfaces, and to the unicity of the application system's administration [38]. Among these elements, the first one is deemed by Lequeux [29] to be indispensable in qualifying a solution as an ERP system, in addition to integration and adaptablity. One must note however that homogenisation is subordinate to integration. Hasselbring [25] indicates that reducing the heterogeneity of IS is one dimension on which to intervene in order to achieve integration. The same can be said of a unique data referential or data base that supports the integration of information flows within the firm, in conjunction with a workflow management system [7].

Transversality refers to the process-oriented view of an ERP system [13]. Its importance comes from the fact that ERP systems are composed of functional modules, and are generally implemented module by module, that is, in a vertical manner. If care is not taken, this could threaten the horizontal design of the organisation, as it is this design that allows one to remove non value-adding activities from business [6]. Beretta [7] adds that without a process-oriented view, advantages in terms of integration would not be obtained from an ERP. A number of failures of ERP projects are in fact due to a lack of transversality in the installed system; for instance, « balkanisation » appears when each organisational entity uses the installed ERP software to consolidate its power base by accentuating its differences [8]. A process-oriented view would put the functional specificities more into the background while promoting informational integration and communication between units. This view would also render more apparent the need for and scope of process reengineering.

Imbedding best practices would not be considered as an essential characteristic of ERP systems. The reason is that this notion is based on adopting generic processes that, despite being exemplary, offer few possibilities of gaining a competitive advantage [14]. Certain authors go further by questioning the universal applicability of so-called « best practices »

[41]. Harrington [24] even qualifies as fallacious the idea that there exist business practices applicable to all organizations. Also, best practices within countries, industries, or specific firms may diverge from the practices imbedded within an ERP system.

The capacity for real-time operation and simulation in an ERP system are consequences of its successful integration. It is this integration that allows the same information to be communicated in real time to all parts of the organisation, and allows simulating the effect of an input on all activities of the organisation. Markus [30] notes for instance that non-integration limits the capacity of a firm, or of a group of cooperating firms, to take important decisions, even when the necessary data reside somewhere in a system.

4.2. OPERATIONALISATION OF ERP CHARACTERISTICS

The preceding discussion leads us to limit essential characteristics to three: integration, flexibility, and transversality. One could say that these are the minimal requirements for a system to be qualified as an ERP or enterprise system. This being said, one can note that from an operational perspective, the major difficulty resides less in the number of characteristics retained than in their qualification.

4.2.1. Operationalisation of integration

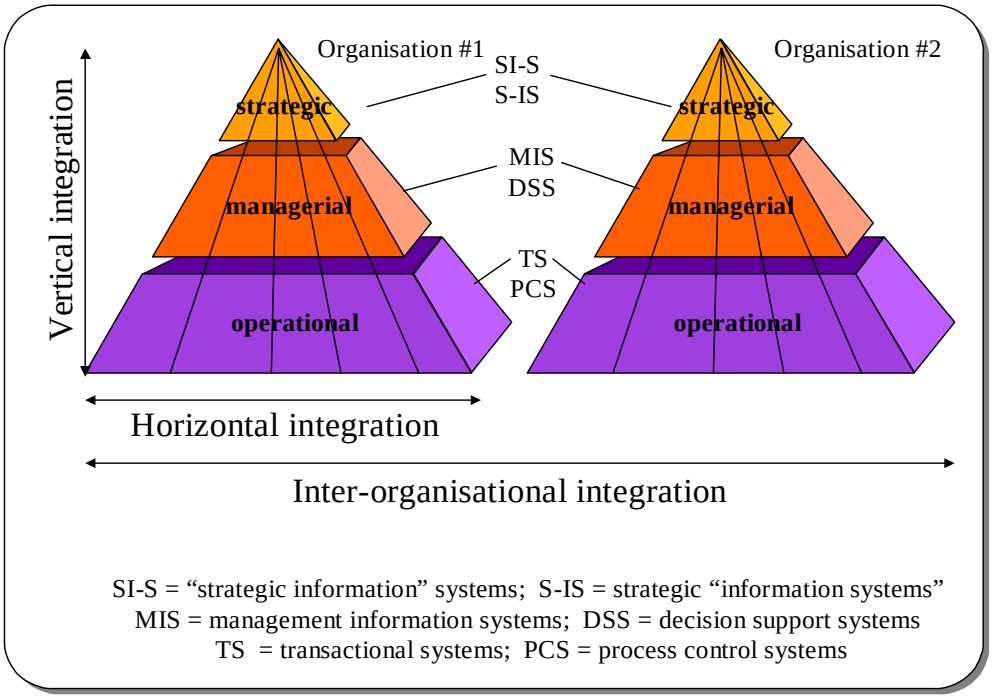
The concept of integration is not exclusive to the IS domain. As underlined by Barki and Pinsonneault [3; 4], this concept occupies a central place in many fields of research other than IS such as management, strategy, and operations management. In the IS field, integration is defined from three perspectives [3] : one is technical, referring to the interconnectivity of IT and a shared conceptual schema for data bases ; a second perspective is inter-organisational, referring to IT-based links between business processes of two or more independent organisations; a third perspective envisages integration in the form of co-ordination and co-operation within project teams and between these teams and other entities in the organisation.

For her part, Markus [30] distinguishes organisational integration (both internal and external) from systems integration. The first is viewed as a tight co-ordination of the various activities undertaken by different individuals, work groups or organisations such that a unified business process is formed. Systems integration is viewed as creating strong links between different IS and data bases. Hence, organisational integration as defined by Markus [30] regroups Barki and Pinsonneault's [4] second and third perspectives, whereas systems integration corresponds to their first perspective, namely the technical one. Systems integration is often necessary to achieve organisational integration but it is not always sufficient [30]. When one then speaks of an ERP system, the underlying type of integration is basically systems integration, but its evaluation cannot ignore the organisational integration that it induces, and that it is deemed to support.

The integration of an ERP system can be envisaged from two angles, considering the integration perimeter (organisational coverage) and the intensity of integration (depth of integration). In the first view, integration can be vertical, horizontal, or inter-organisational [36] (see figure 2). Vertical integration refers to the degree of interconnection (connectivity, compatibility) between hierarchical levels in the organisation. Horizontal integration is ascertained by the interconnection between various organisational functions or departments, whereas inter-organisational integration refers to the firm’s interconnection with its business partners.

In the second view, one distinguishes the extent to which integration is achieved, be it vertical, horizontal, or inter-organisational. In this regard, Toussaint *et al.* [42] mention the

quality of integration, this being ascertained through the co-ordination of behaviours (presentation, execution, and data access) and of communication (tracking of messages



Source : adapted from [36].

between sender and receiver, structure of the information exchanged, way in which information is exchanged and processed).

Figure 2: Integration perimeter

4.2.2. Operationalisation of flexibility

The concept of flexibility has been widely studied in the filed of operations and production management [5; 15]. It has also been approached from within the IS domain [11; 17; 22]. Whereas operations management studies obviously focus on the flexibility of the productive apparatus (manufacturing flexibility), and IS studies are preoccupied with the flexibility of the IT infrastructure, both fields show similarities in their definition of the concept, and in the difficulties confronted in order to operationalise and measure it. The operational definitions and measures of the two obviously differ however, given that they apply to two different systems. One can also note that IS flexibility may have an effect on manufacturing flexibility.

In the short term, flexibility means the capacity to adapt to changing conditions by fully using all existing resources, and in the long term, it indicates the capacity to introduce new products, new production resources and methods, and to integrate these within the existing production system. For D'Souza and Williams [15], the flexibility of the production system or manufacturing flexibility is a multidimensional construct that represents the ability of the manufacturing function to make the necessary changes in reaction to changes in the firm's environment, and to do so without significant negative effects on the firm's performance.

Synthesizing different points of view in the literature, Golden and Powell [22] propose a quadri-dimensional view of flexibility: a temporal dimension, range, intention, and focus. The first dimension refers to the time it takes the organisation to react to change. The second is ascertained by the variety of responses available to the organisation in order to face both foreseen and unforeseen changes. The third is meant to determine if the organisation is proactive or reactive, offensive or defensive in regard to change. The last dimension refers to the internal or external orientation of the organisation's flexibility.

The last two dimensions (intention and focus), as opposed to the first two (temporal and range), could be considered as qualifiers of flexibility, but not necessarily as definitional elements of this concept. As noted by Golden and Powell [22], intention and focus depend on the context. Whatever the intention or focus, flexibility is not put in question as it is a strategic choice of the organisation. In all circumstances however, a system that reacts less quickly (temporal dimension) or proposes fewer responses to change (variety) will be considered as less flexible. Intention and focus would then constitute secondary dimensions, whereas the other two dimensions would be primary.

To measure flexibility's temporal dimension, Golden and Powell [22] propose efficiency and responsiveness measures, whereas they propose measures of versatility and robustness for the range dimension. Here, efficiency means the ability to maintain the same performance level while changes occur. Responsiveness will be ascertained by the quickness with which the organisation adapts to change. Versatility and robustness both refer to the range of activities that can be accomplished by a system, the first one relating to foreseen changes whereas the second relates to unforeseen changes.

For their part, Byrd and Turner [11] have developed a measure of IT flexibility, taking into account both the technical infrastructure (data, applications, networks) and the human infrastructure (competencies for effective management of IT). The technical infrastructure is measured by four indicators: connectivity, compatibility, functionality of applications, and data transparency. The human infrastructure is ascertained through: technological management, business abilities, management knowledge, and technical knowledge. Note that the first two are redundant with the measure of integration, denoting again the intertwined relationship between these two concepts. Note also that on the criteria of connectivity and compatibility, the measure of flexibility developed by Byrd and Turner [11] is redundant with the integration measure defined in the preceding section of this paper. In fact, these two criteria were attached to integration in the latter's second-order factor analysis.

It does not appear surprising, at least theoretically, that the notion of integration appears in the conceptualization of flexibility, as these two notions are inter-related and inter-dependent: by enabling the links between the various organisational functions and levels, integration allows fore the instantaneous diffusion of information throughout the organisation, which then allows immediate adjustments made necessary by the new information. This theoretical relationship between integration and flexibility is not necessarily confirmed in practice. Beach *et al.* [5] note for example that it is as important as it is difficult to have the connectivity between internal and external systems (intra and inter-organisational integration) come into play in order to improve flexibility.

The flexibility of an ERP system could thus be measured by using a combination of the two preceding approaches, however avoiding a redundancy with the measure of integration. As shown in Figure 3, this characteristic could be operationalised under four dimensions, that is, two dimensions from Golden and Powell [22], namely the temporal dimension (efficiency and responsiveness of the system) and the system's range (versatility and robustness), and two from Byrd and Turner [11], namely part of the technical dimension that is not redundant with integration (functionality of system applications and data

transparency), and the human dimension (managerial and technical knowledge, business and managerial abilities in regard to the ERP system).

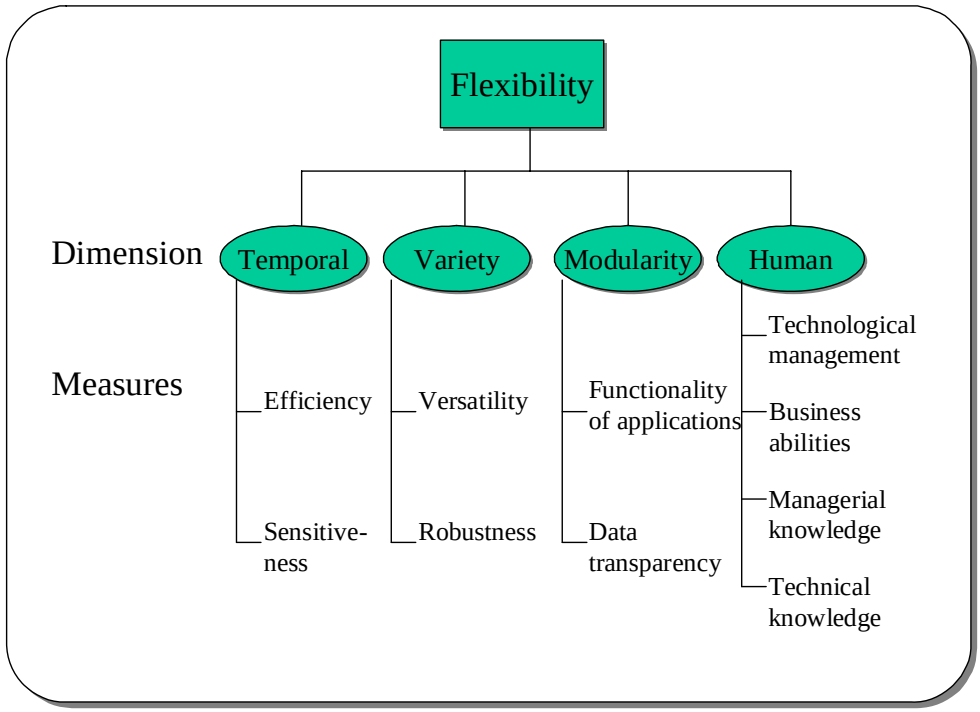


Figure 3: Flexibility dimensions and measures

4.2.3 Operationalisation of transversality

The process notion is central to transversality. This notion is defined as an activity or a set of activities that are linked in an ordered fashion, and that transform inputs into outputs for customers in a repetitive flow [21]. Such a definition emphasises value-adding activities, repetitiveness, and a customer-orientation. The process is thus a conceptual scheme that helps managers to assess the utility or value of each specific activity [7].

Various aspects of a process cannot be apprehended through traditional accounting measures but rather through performance indicators such as production cycle times, delivery delays, output quality, productivity, customer satisfaction, and learning [33]. Understanding their activities from an horizontal rather than vertical perspective would thus allow business firms to get closer to their customers while simultaneously increasing the quality of their organisation and their competitiveness [2; 21]. Operationalising the transversality of an ERP system requires measuring the extent to which it is process-oriented. For Forsberg *et al.* [21], there are various manifestations of an ERP infrastructure's embodying a process-orientation: use of a common language, customer focus, cooperation, holistic or « big picture » view, reduction of costs and delays, increased learning, standardisation and co-ordination. In order to measure the transversality of an ERP system, one could ascertain to what extent it meets these conditions (see Figure 4).

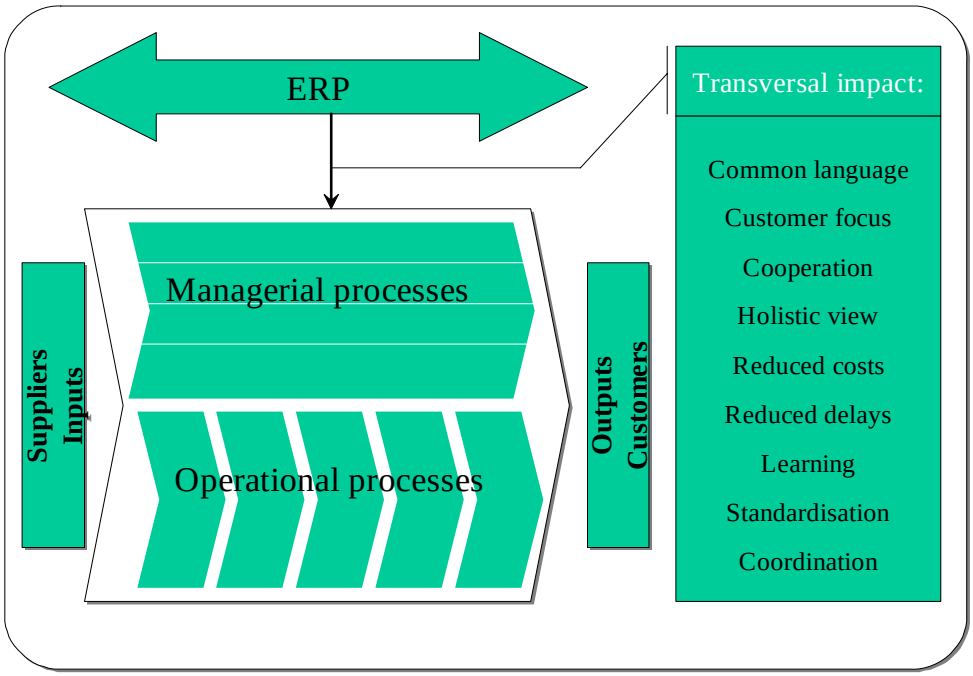


Figure 4: ERP transversality

5. LIMITATIONS AND IMPLICATIONS FOR RESEARCH

IS research can profit from a more precise definition of ERP systems. In this paper, we have attempted to take stock on what is actually meant by the ERP appellation. To do this, we have placed an emphasis on determining the minimal or indispensable characteristics that a system should possess to be qualified as an ERP system. We have then tried to define these characteristics in order to facilitate their operationalisation. This approach is strictly theoretical however, and would require empirical validation based on the characterization of ERP systems actually implemented in organisations. Another research avenue would be to further pursue the operationalisation of ERP integration, flexibility and transversality in order to measure these characteristics across actual ERP systems.

Other research paths are summarised in Figure 5. Given that previous studies have suggested the existence of mutually dependent relationships between ERP chasracteristics, notably between flexibility and integration [9; 30], it would be interesting to pursue such analysis further by examining the nature of the interdependencies between flexibility, integration and transversality, and by assessing the impact of such relationships on the ERP system's performance.

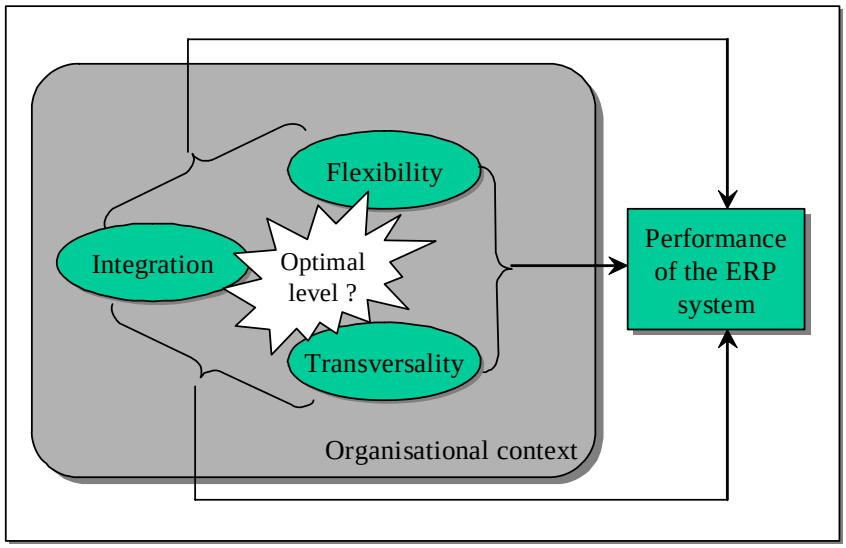


Figure 5: Research avenues on the impact of key characteristics of ERP systems

One could also envisage research that aims to evaluate a system's performance level as determined by its levels of flexibility-integration-transversality. Hitt *et al.* [26] suggest that there is an optimal level of functional integration beyond which diseconomies of scale begin to appear. The problem would then be to determine optimal levels of ERP flexibility, integration and transversality, if they exist, in relation to organisational or industry factors. This paper hoped to contribute to a better understanding not only of what is actually meant by the term "ERP" or "enterprise system", but also to raise applied research issues that are of interest to firms that have implemented or plan to implement ERP systems, and to vendors and consultants that assist in their implementation

.

REFERENCES

[1] AMR Research. AMR Research predicts ERP market will reach $52 billion by 2002. *Enterprise Resource Planning Software Report, 1997-2002*, 1998. http://www.amrresearch.com/Content/viewpress.asp?id=13376&docid=848

[2] APQC's International Benchmarking Clearinghouse, in partnership with Arthur Andersen & Co., SC. Process classification framework. *APQC, American Productivity & Quality Center*, 1996.

[3] Barki, H. and Pinsonneault, A. Explaining ERP implementation effort and benefits with organizational integration. *Cahiers du GreSI*, No. 02-01, 2002, 27 pages.

[4] Barki, H. and Pinsonneault, A. The construct of organizational integration: A research framework and its application to enterprise systems research. *Cahier du GreSI*, No. 03-04, 2003, 31 pages.

[5] Beach, R., Muhlemann, A.P., Price, D.H.R., Paterson, A. et Sharp, J.A., "Manufacturing operations and strategic flexibility: Survey and cases", *International Journal of Operations & Production Management*, Vol. 20, No. 1, 2000, pp. 7-30.

[6] Bentley, T. Systems - A process orientation. *Management Accounting*, 1995, p. 36.

[7] Beretta, S. Unleashing the integration potential of ERP systems. The role of process-

based performance measurement systems. *Business Process Management Journal*, Vol. 8, No. 3, 2002, pp. 254-277.

[8] Besson, P. Les ERP à l'épreuve de l'organisation. *Systèmes d'Information et Management*, Vol. 4, No. 4, 1999, pp. 21-51.

[9] Brandyberry, A., Rai, A. and White, G.P. Intermediate performance impacts of advanced manufacturing technology systems: An empirical investigation. *Decision Sciences*, Vol. 30, No. 4, 1999, pp. 993-1020.

[10] Brown, J. P. Is ERP a Silver Bullet? *APICS – The Performance Advantage*, Vol. 10, No. 13, 2001. http://www.apics.org/magazine.

[11] Byrd, T.A. and Turner, D.E. Measuring the flexibility of information technology infrastructure : Exploratory analysis of a construct. *Journal of Management Information Systems*, Vol. 17, No.1, 2000, pp. 167-208.

[12] Caldas, M.P. et Wood Jr. T. How consultants can help organizations survive the ERP frenzy. *Academy of Management's Annual Meeting*, Chicago, august, 6th-11th, 1999. http://www.gv.br/prof_alunos/thomaz/ingles/paper6.htm

[13] Carbonel, M. Dérives organisationnelles dans les projets ERP : les cas de Guerbet et Gaumont. *Systèmes d'Information et Management*, Vol. 6, No. 1, 2001, pp. 71-85.

[14] Davenport, T.H. Putting the enterprise into the enterprise system. *Harvard Business Review*, Vol. 76, No. 4, 1998, pp. 121-131.

[15] D'Souza, D.E. and Williams, F.P., Toward a taxonomy of manufacturing flexibility dimensions. *Journal of Operations Management*, Vol. 18, No.5, 2000, pp. 577-593.

[16] Deloitte Consulting. ERP's second wave: Maximazing the value of ERP-enabled processes. Deloitte Consulting, Atlanta, 1999.

[17] Duncan, N.B. Capturing flexibility of information technology infrastructure : A study of resource characteristics and their measure. *Journal of Management Information Systems*, Vol. 12, No. 2, 1995, pp. 37-57.

[18] Esteves, J. and Pastor, J.A. Enterprise Resource Planning systems research: An annotated bibliography. *Communication of AIS*, Vol. 7, No. 8, 2001.

[19] Everdingen, Y.V., Hillegersberg, J.V. et Waarts, E. ERP Adoption by european midsize companies. *Communication of the ACM*, Vol. 43, No. 4, 2000, pp. 27-31.

[20] Forest, G. Généalogie des ERP et gestion des flux physiques. *Systèmes d'Information et Management*, Vol. 4, No. 4, 1999, pp. 71-89.

[21] Forsberg, T., Nilsson, L., and Antoni, M. Process orientation : The swedish experience. *Total Quality Management*, Vol. 10, No. 4&5, 1999, pp. 540-547.

[22] Golden, W. and Powell, P. Towards a definition of flexibility: In search of the holy grail? *Omega*, Vol. 28, 2000, pp. 373-384.

[23] Greenemeier, L. ERP : It's not just for big companies. *InformationWeek*, October 29, 2001.

[24] Harrington, H.J. The fallacy of universal best practices. *The TQM Magazine*, Vol. 9, No. 1, 1997, pp. 61-75.

[25] Hasselbring, W. Information system integration. *Communications of the ACM*, Vol. 43, No. 6, 2000, pp. 32-38.

[26] Hitt, L.M., D.J. Wu and Zhou, X. ERP investment : Business impact and productivity measures. *Journal of Management Information Systems*,Vol. 19, No. 1, 2002, pp. 71-98.

[27] Jesitus, J. Broken Promises? FoxMeyer's project was a disaster. Was the company too aggressive or was it misled? *Industry Week*, November 3, 1997, pp. 31-37.

[28] Klaus, H., Rosemann, M. and Gable. G.G. What is ERP? *Information Systems Frontiers*, Vol. 2, No. 2, 2000, pp. 141-162.

[29] Lequeux, J.L. Manager avec les ERP. Progiciels de gestion intégrés et Internet.

Architecture applicative, 2ème édition, Éditions d'Organisation, Paris, 2002.

[30] Markus, M.L. Paradigm shifts - E-business and business/systems integration. *Communications of the AIS*, Vol. 4, (article10), 2000.

[31] Markus, M.L. Reflections on the systems integration enterprise. *Business Process Management Journal*, Vol. 7, No. 3, 2001, pp. 171-180.

[32] Markus, M.L. and Tanis, C. The enterprise system experience – From adoption to success. In Zmud, R.W. (Editor). *Framing the domains of IT management. Projecting the future… through the past.* Pinnaflex Education Resources, Inc., Cincinnati, Ohio, 2000.

[33] McCormack, K. and Johnson, B. Business process orientation, supply chain management, and the e-corporation. *IIE Solutions*, Vol. 35, No. 10, 2001, pp. 33-37.

[34] Österle, H., Fleisch, E. and Alt, R. *Business networking. Shaping collaboration between enterprises.* Éd. Springer, Heidelberg. Second, revised and extended edition, 2001.

[35] Parr, A.N. and Shanks, G. A Taxonomy of ERP implementation approaches. *Proceedings of the 33rd Hawaii International Conference on System Sciences*, Hawaii, 2000.

[36] Prosser, A. and Ossimitz, M.L. Data Warehouse Management. Department of Production Management, University of Economics and Business Administration, Vienna, Austria, 2000. http://www.indi.wu-wien.ac.at/download/data_warehouse.pdf

[37] Rosemann, M. and Wiese, J. Measuring the performance of ERP software : a balanced scorecard approach. *Proceedings of the Australasian Conference on Information Systems*, Wellington, 1-3 december, 1999.

[38] Rowe, F. Cohérence, intégration informationnelle et changement : esquisse d'un programme de recherche à partir des Progiciels Intégrés de Gestion. *Systèmes d'Information et Management*, Vol. 4, No. 4, 1999, pp. 3-20.

[39] Scott, J.D. The FoxMeyer Drugs' bankruptcy : Was it a failure of ERP? Proceedings of The Association For Information Systems, Fifth Americas Conference on Information Systems, 1999.

[40] Smyth, R.W. Challenges to successful ERP use (research in progress). *Proceedings of the 9th European Conference on Information Systems*, Bled, Slovenia, June 27-29, 2001, pp. 1227-1231.

[41] Soh, C., Kien, S.S. and Tay-Yap, J. Cultural Fits and Misfits :Is ERP A Universal Solution? *Communications of the ACM*, Vol. 43, No. 4, 2000, pp. 47-51.

[42] Toussaint, P.J., Bakker, A.R. and Groenewegen, L.P.J. Integration of information systems: Assessing its quality. *Computer Methods and Programs in Biomedecine*, Vol. 64, No. 1, 2001, pp. 9-35.

[43] Truex, D. ERP Systems as facilitating and confounding factors in corporate mergers : The case of two canadian telecommunications companies. *Systèmes d'Information et Management*, Vol. 6, No. 1, 2001, pp. 7-21.

[44] Tuteja, A. Enterprise Ressource Planning : What's there in it? 1998. http://www.geocities.com/CollegePark/Library/6045/erp.html

[45] Willis, T. H., Willis-Brown, A.H. and McMillan, A. Cost containment strategies for ERP system implementations. *Production and Inventory Management Journal*, Vol. 42, No. 2, 2001, pp. 36-42.

Received: 30 September 2004
Accepted: 27 October 2005

UDC:007:681.3:651(082)
Original scientific paper

SELECTING NEURAL NETWORK ARCHITECTURE FOR INVESTMENT PROFITABILITY PREDICTIONS

Marijana Zekić-Sušac, Nataša Šarlija
Faculty of Economics, University of J.J. Strossmayer in Osijek, Croatia
{marijana, natasa}@efos.hr

Mirta Benšić
Faculty of Organization and Informatics
Department of Mathematics, University of J.J. Strossmayer in Osijek, Croatia
mirta@mathos.hr

Abstract: *After production and operations, finance and investments are one of the most frequent areas of neural network applications in business. The lack of standardized paradigms that can determine the efficiency of certain NN architectures in a particular problem domain is still present. The selection of NN architecture needs to take into consideration the type of the problem, the nature of the data in the model, as well as some strategies based on result comparison. The paper describes previous research in that area and suggests a forward strategy for selecting best NN algorithm and structure. Since the strategy includes both parameter-based and variable-based testings, it can be used for selecting NN architectures as well as for extracting models. The backpropagation, radial-basis, modular, LVQ and probabilistic neural network algorithms were used on two independent sets: stock market and credit scoring data. The results show that neural networks give better accuracy comparing to multiple regression and logistic regression models. Since it is model-independant, the strategy can be used by researchers and professionals in other areas of application.*

Keywords: *neural networks, non-linear forward strategy, stock market prediction, credit scoring.*

1. INTRODUCTION

Artificial intelligence based on learning theory improves the ability of using the prior knowledge and data in order to make effective decisions. Neural networks (NNs) can be used for prediction, classification, and association problems in various problem areas. Finance and investing is the second most frequent business area of neural network applications after production/operations [17]. The lack of standardized paradigms that can determine the efficiency of certain NN algorithms and architectures in particular problem domains is emphasized by many authors [5]. The focus of the paper is in improving NN prediction and classification ability by selecting the best NN architecture for a certain problem. Since the aim of the research is to obtain the best model for the observed data,

NNs were compared to a method that is classically used for that type of problem (multiple regression for the stock return prediction problem, and logistic regression for the credit scoring problem). The next section provides the description of previous research results regarding NN architecture selection. A brief overview of NN methodology used including five NN algorithms for prediction and classification is described in the third part. Part 4 presents the suggested forward strategy for selecting NN architecture, while the data and results are described in parts 5 and 6. After the conclusion, some guideliness for future research in this area are suggested.

2. DESCRIPTION OF PREVIOUS RESEARCH

Some of the most representative problems being solved by NNs are bankruptcy predictions, risk assessments of mortgage and other loans, stock market predictions (stock, bond, and option prices, capital returns, commodity trade, etc.), financial prognoses (returns on investments) and others [18]. NNs in finance are frequently applied in predicting stock performance and selecting stocks for trading on stock markets. Stock markets are suitable area for NN application because of the existence of nonlinear dependencies assumed on the basis of market microstructure, feedback effects in market prices and empirical observations, and confirmed for most of the world major markets [13]. Linear models cannot serve as efficient forecasting systems in such situations. There are three main groups of problems that NN applications frequently deal with in stock markets: (1) classifying stocks into the classes such as: positive return stocks and negative return stocks, (2) predicting the exact stock price for one or more days in advance, based on previous stock prices and related financial ratios, and (3) modeling stock performance and forecasting. The last group of applications is not only focused on the prediction of future values, but also on the factor significance estimation, sensitivity analysis among the variables that could impact the result, and other analyses of mutual dependencies. NN applications in banking are mostly concentrated on credit assignment problems, bankruptcy predictions, customer segmenting, detecting credit card frauds, and other problems.

The analysis of previous research also shows a variety of NN algorithms and evaluation measures used in financial applications [19]. The backpropagation (multi-layer perceptron) algorithm is the most frequently used, although other algorithms are present in some rare applications. The three-layer structure seems to be effective according to many authors, with the exception of a few applications [13] where the four-layer structure outperforms other structures. The majority of NN applications use the sigmoid transfer function, with changeable learning parameters and momentum. One step toward investigating other NN algorithms than backpropagation is the usage of time-delayed, recurrent and probabilistic networks with great success in false alarm accuracy [14]. The comparison of NN's performance on stock market predictions with statistical multiple regression is done by several authors including White [16] and Refenes et al. [12].

More recent research shows that combining NNs with other data mining techniques is very effective. Lam [6] used a backpropagation NN in combination with a rule extraction technique to predict stock return for 364 S&P companies on the basis of fundamental and technical data. Her results indicate that NN using one year's or multiple years' financial data significantly outperform the minimum benchmark (the overall market average return), but not the maximum benchmark (the top one-third returns in the market). Enke and Thawornwong [2] introduce a combination of neural networks and data mining techniques in forecasting stock market return. The information gain technique is used to evaluate the predictive relationships of input variables, while the NN models were then used to forecast the future values.

2.1. PREVIOUS RESEARCH ON NEURAL NETWORK TOPOLOGY AND ALGORITHM SELECTION

Previous research in finding the best NN architecture was mostly focused on network topology, resulted with some heuristic rules [7], [8], [10], and optimizing algorithms such as A* algorithm [1], cascading, pruning, genetic algorithms and others. It is proved (by Hornik et al., 1989. in [8]) that three-layer networks are able to approximate any discontinuous function. In many cases, more layers slow down the training time because the gradient of the error is more unstable and there is a higher risk of local minima. Mozer (in [11]) showed that processing units (nodes) vary in their functional importance for solving a problem, indicating that the relevance of a unit can be determined by comparing performance of the NN when the unit is included to the performance of the NN when the unit is removed. The overview of structure optimization techniques is given by Quinlan [11], who synthesizes a number of optimization techniques into two main groups: (a) backpropagation derivatives, and (b) algorithms for discrete unit networks. Backpropagation derivatives are further divided into: (1) techniques based on units removal (skeletonisation, artificial programmed cell death, and gauging the optimal size of network in terms of generalization capability), (2) pruning techniques (Thodberg's technique of pruning the connections with small weights, and optimal brain damage), (3) adding units and connections (Meiosis networks), (4) both adding and pruning units and connections (Bartlett's technique, and Hirose's technique). The second group of algorithms for discrete unit networks includes the techniques such as tiling algorithm, pointing and tower algorithms, neural tree approach, and upstart algorithm. Most of those techniques are based on two common principles: gradually adding new nodes or gradually excluding the nodes, using different criteria. Another shared characteristic of those techniques is that they make the network dynamic in the sense of changing its structure. The idea of self-pruning is in simultaneous minimization of output error and minimization of the number of hidden neurons. The main disadvantage of both cascading and pruning procedures is that the upper limit of number of hidden nodes should be determined in advance, while there is no rule of how to determine it. One way to overcome this limitation is to use genetic algorithms that start with different, randomly chosen topologies. A*- algorithm, proposed by Nilson, 1980 (in [1]) applies graph theory to find the optimal path that presents the best NN structure. Its advantage is in the fact that it does not imply any restrictions on generated structure i.e. there is no limited number of hidden neurons.

Besides dynamic optimizing techniques, there are some static heuristic rules for determining a NN structure, such as Masters' formula [8] that computes the number of hidden units on the basis of the number of input and output units. Some of the above techniques and rules were tested in relation to underlying data in a stock prediction model [20], and it was shown that Masters' rule performs good on the training set of the models, but those NNs have low performance on the validation set. It was also shown that cascading technique performs well on data with small variance, while pruning technique performs well on most of the models, and generalization capability of such models is acceptable.

The selection of NN algorithm is primarily subject to input and output functions built in an algorithm. They determine the main selection of algorithms according to the type of the problem: prediction, classification, association, data compression, data conceptualization, etc. Some general-purpose algorithms allow adding specific functions that will adjust them to a specific type of problem. For example, backpropagation and radial-basis algorithms that are primarily aimed for prediction can be used for classification by adding a softmax activation function to the output layer [10]. Another criterion for algorithm selection is the nature of the data, since researchers report different results depending on the dataset and models used. Masters suggests the probabilistic neural network as a good selection for

classification problems when there are outliers in data [8], and when the learning speed is important, since this algorithm does not learn iteratively, but use only one pass through the dataset. NeuralWare [10] brings some heuristic rules for selecting NN algorithm based on a problem type and data constraints. According to [10] the backpropagation and the modular algorithms are the most suitable for noisy data, they are also recommended together with LVQ for the models with many input variables, which was also shown by Zekic and Klicek [20]. Although none of the algorithms is evaluated as excellent for sparse data, the counter propagation, general regression, LVQ, and probabilistic are suggested as good choices in such datasets. When the data is nonstationary, the general regression is selected as the best solution, followed by fuzzy ARTMAP and modular algorithms. The same authors [20] test eight NN algorithms according to the nature of data in a stock market dataset, and obtains that the performance of NN algorithms significantly (0.5 level) differs for the models with sparse data and data with outliers. In the prediction type of problem, the modular network is extracted as the best for such data followed by backpropagation on the basis of the average trade result (ATR). In case of the existence of nonstationary variables in the model, the general regression network showed the worst result, and is not suggested for such type of data. None of the NN algorithms significantly outperforms others in classification problems, although the LVQ showed the lowest result. These findings reveal that the selection of algorithms depends on both data nature and model used; therefore they support the strategy for selecting NN algorithms based on testing more algorithms on the same dataset.

3. FORWARD STRATEGY FOR SELECTING NN ARCHITECTURE

The selection of NN architecture needs to take into consideration the type of the problem, the nature of the data in the model, as well as some cross-validating strategies based on result comparison. The type of the problem (classification, prediction, function approximation, association) makes the general selection based on the theoretical foundation of NN algorithms, although there are a number of general-purpose algorithms that can be used for all types of problems (such as backpropagation, radial basis function, general regression). Besides that, most of the prediction problems can be also defined as classification problems, with some modifications in expressing the output variable (for example, prediction of stock prices can be easily modified as classification of stocks into poorly-performed and good-performed stocks). For the above reasons, it was necessary to define a strategy for selecting NN architecture depending on the model tested. The suggested strategy is to build NN models by starting from a single variable, then gradually add another one and test if it improves the model performance. The variables that improve the performance of the model are included in the model. The strategy is not only used for modeling purpose, but also for the purpose of finding the best NN algorithm and structure. Prediction algorithms were backpropagation, radial-basis, general regression, and modular, while the following four algorithms were tested for classification problems: backpropagation and radial basis function network (with softmax activation function in the output layer in order to obtain probabilities), probabilistic and learning vector quantization. Each more-layered NN algorithm (backpropagation and radial-basis with and without softmax) is structured separately including 1 hidden layer and 2 hidden layers (in order to see if the additional hidden layer can improve the detection of nonlinear dependencies

among the data). Additional structure and parameter optimization is done in the training phase of the network. The strategy can be described in the following steps:

1) select the first NN algorithm based on the problem type

2) perform forward modeling

The modeling procedure starts by using a single input variable. The NN is trained on in-sample and tested on out-of-sample data, while the first test result serves as the baseline measure (or reference) for the next NN. The second NN model adds another exogenous variable. The out-of-sample test result is compared to the reference, and if the new result is better than or equal to the reference, the variable is sustained in the model and the new result becomes the reference. If the result is worse than the reference, the variable is not included in the model, and the next variable is tested. Such iterative process is repeated until all the exogenous variables are tested and the best model is saved.

3) test the model with all available variables

After the forward modeling strategy extracts the model, the whole universe of input variables is tested together in one model and the performance of such maximal model is compared to the best result obtained in step 2.

4) repeat steps 1-3 for the next NN algorithm (or for the next parameter within the same algorithm) until all available NN architectures were tested

In this step, it is possible to change only one parameter in a NN algorithm used in step 1 (for example to change the transfer function in a layer, or to change a learning parameter etc.), or to change a NN algorithm type.

Completion of the first three steps of the above process extracts the best model using one NN architecture. Since the change of one parameter in the architecture can produce different results regarding the variable selection, it is necessary to test all available NN architectures suitable for a problem. The overall best NN model is identified at the end of the procedure by choosing the model with the overall best result. All the examined models were tested on the out-of-sample data. The above iterative strategy is conducted a number of times in our experiments testing different NN architectures. The above procedure is graphically presented in Figure 1. In order to perform the strategy a VB program is created, which controls the whole iterative process of selecting the NN architectures and variables, evaluating the results, and presenting it to the user, while the neural computation is done by the NeuralWare software.

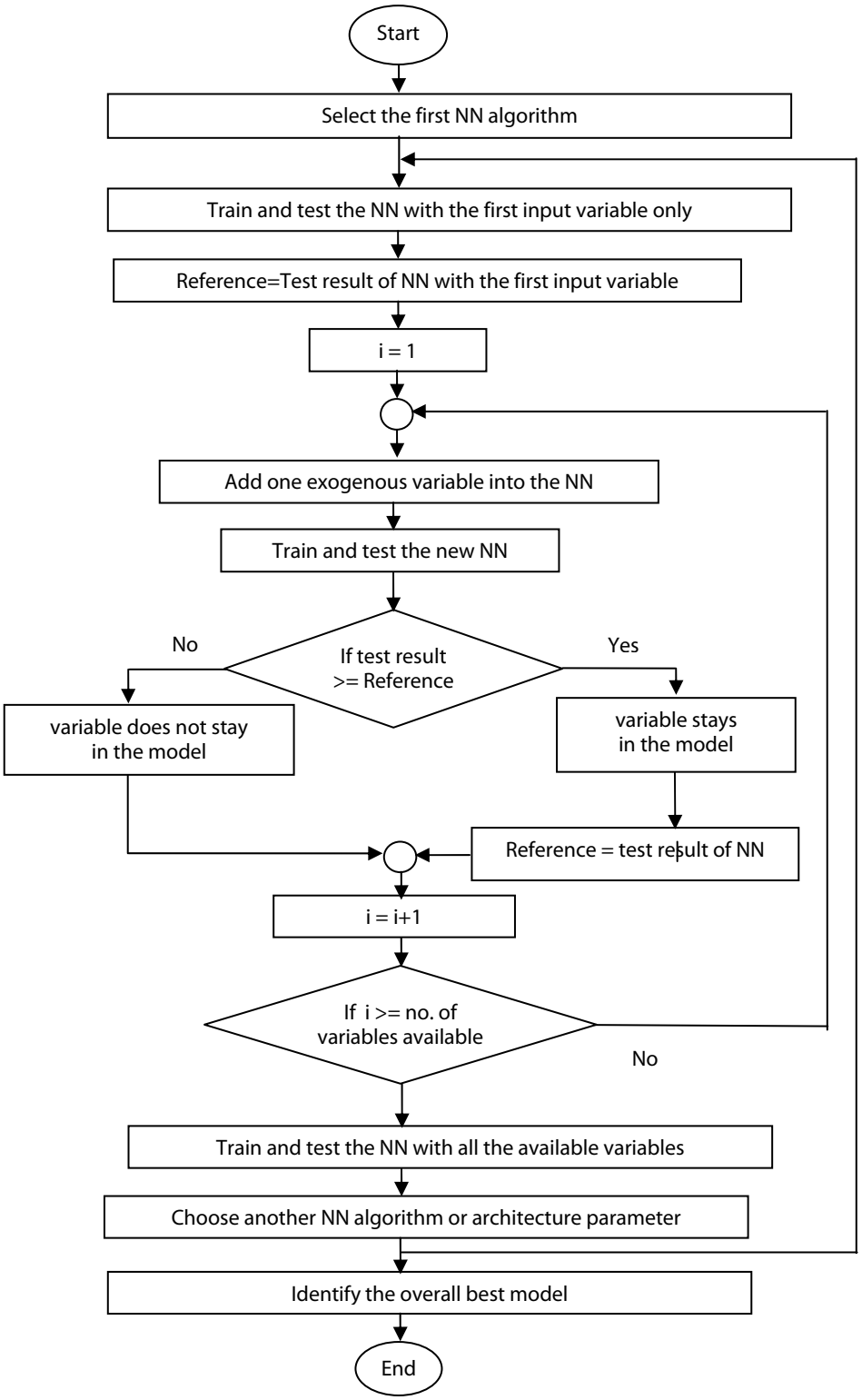


Figure 1: Forward strategy for selecting NN architecture and model

4. NEURAL NETWORK METHODOLOGY

Four prediction and four classification algorithms were tested. They were selected due to their well-established and confirmed computational foundation. In the prediction type of problem, backpropagation, radial basis, general regression and modular (proposed by Jacobs et al, 1991) algorithms were used for predicting stock return rate. For classification type of problem, softmax activation function is added to the standard backpropation and radial-basis function algorithms in order to obtain the probabilities in the output classes. LVQ, as a supervised version of the Kohonen algorithm with unsupervised kernel in the hidden layer is also tested. In order to make the learning process in backpropagation algorithm faster, the extended delta-bar-delta (EDBD) rule is used instead of the standard delta rule. The initial learning rates for the EDBD learning rule used in our experiments were set as follows: 0.3 for the first hidden, 0.2 for the second hidden layer, 0.15 for the output layer, and the inital momentum term was set to 0.5. The tangent hyperbolic transfer function was used in the hidden layer. To overcome the problem of local minima in the backpropagation, a stochastic method of simulated annealing is applied by adding noise into the result of the input function before sending output to other units in the network. The range of noise is determined by the temperature (-0.01*t, 0.01*t). The temperature is initially set to a high value, then decreased gradually in order to lead the transfer function to the area of the global minimum. Overtraining is avoided by the so-called "save best" procedure, a sort of cross-validation process which alternatively trains and tests the network (using a separate test sample) until the performance of the network on the test sample does not improve for n number of attempts. The maximum number of iterations in our experiments was set to 100000, n=10, while the number of iterations in each training session was 5000 for the first dataset, and 1000 for the second dataset. After the best network is selected, it is tested on a new validation sample to determine its generalization ability. Saturation of weights is prevented by adding a small F'offset value to the derivative of the sigmoid transfer function. It is experimentally proved that the value 0.1 is adequate for the sigmoid, and 0.3 for the tangent hyperbolic transfer function (Fahlmann in [10]).

The topology of the backpropagation NN consisted of maximum 105 hidden units in the first hidden layer and 52 hidden units in the second hidden layer. The same was used in the radial-basis function network. The general regression networks consisted of 1 hidden layer with maximum 105 hidden units. The modular network consisted of 1 hidden layer, 3 local experts (consisted of backpropagation networks) with maximum 105 hidden units in each, while the gate network had 35 hidden units.

Probabilistic network is a classifier that uses Parzen windows for clustering and produces a number of classes in the output. The parameter σ is optimized by cross-validating a heuristically set range of σ values, such that each σ value is used to train and test the network. The minimum value of sigma was 0.1; the maximum value was 3, with the step of 0.2. Euclidean summation function in the pattern unit is used, radius of influence was 0.25, and output mode for the output layer was competitive. LVQ network consisted of a Kohonen layer, and an output layer with predetermined classes. The number of Kohonen units was set to 10% of the size of the train sample. The number of learning iterations in the LVQ1 phase is set to 25000, the number of learning iterations in the LVQ2 phase is set to 10000, the initial learning rate for LVQ1 was 0.06, and the initial learning rate for LVQ2 was 0.03. The LVQ2 width parameter was 0.2, the option "in-class winner always learns" is used, with the conscience factor 1.

The output layer in classification networks consisted of three neurons representing three classes of stocks in the stock market dataset, and two neurons representing two classes of applicants in the credit scoring dataset.

According to Masters' suggestions for appropriate evaluate measures in financial applications (1998), we have evaluated stock market neural networks using the ATR, although the total return, average return per trade, and correlation between the computed and the desired output were also computed. The performance of credit scoring neural networks was measured by the total hit rate. The NeuralWare software was used to run neural networks.

5. DATA

The first dataset consisted of daily data on IBM stock return, financial ratios and macroeconomic variables. Daily close prices were collected by i-Soft Inc.[1], financial ratios and dividends were computed or collected from the IBM quarterly balance sheets and income statements provided by Value Line Inc., while macroeconomic variables were provided by Econ*magic*.com, an economic time series source[2]. The total data sample, after data pre-processing, consisted of 848 cases. The train sample consists of the first 80% of the cases, while the other 20% of the cases covering the rest of the period were used to form the out-of-sample test data. For the purposes of optimizing NN structure and training time, the train sample is further divided into the train1 sample (85% of the train sample), and the test1 sample (15% of the train sample). Test1 sample is needed in NNs in order to avoid the influence of out-of-sample data to the optimizing procedure in NNs, which is necessary according to [9]. The available space of input variables is created on the basis of the arbitrage pricing theory or the factoral model [12], consisted of variables: V1 - moving average term (MA(1)) - obtained by ARIMA, V2 - volume, V3 - weekly stock volatility, V4 - beta, V5 - return on investment (ROI), V6 - capital intensiveness (CI), V7 - financial leverage (FL), V8 - receivables intensiveness (RI), V9 - inventory intesiveness (II), V10 - current ratio (CR), V11 - cash position (C), V12 - 30-year mortgage rate (MR), V13 - 3-month US treasury bill rate (TBR), V14 - bank prime loan rate (BPLR), V15 - exchange rate of Japanese yen according to US dollar (ER), V16 - federal funds rate (FFR), V17 - S&P 500 total return rate (S&P500 TRR), V18 - industrial production index (IPI), V19 - unemployment rate (UR), V20 - index of leading indicators (ILI), and V21 - consumer price index (CPI). The output variable was the stock return, defined as the actual rate of return realized over some evaluation period. It was represented in two ways: (a) as the absolute value of stock return for the next day $(t+1)$ in prediction problems (first differences are used to eliminate the trend influence) computed according to White (1996):

$$R_{t+1}=(p_{t+1}-p_t+d_{t+1})/p_t, \quad (1)$$

where p_t is the closing price on day t, d_t is the dividend paid in day t, and (b) as a classification problem, with three output variables representing three classes of stock return: positive, neutral, and negative stock return. In order to classify the stock return rate into three classes, 1 of N binary code is used together with the threshold values -0.5 and 0.5. The rate of return is classified as positive if it is greater than 0.5, negative if it is less than –0.5, and neutral if it is greater than or equal to –0.5 and less than or equal to 0.5.

The second dataset consisted of a small business credit scoring data, and was collected randomly in a Croatian savings and loan association. The sample size was 166 credit applicants – small entrepreneurs - that entered the bank requesting for a loan. The selection of input variables was guided by the previous research (see for details [21]), including both business and personal characteristics of entrepreneur. The input variables were: main activity of the firm, new firm (yes/no), number of employees, entrepreneur's occupation,

[1] StockWiz, a registered trade mark of i-Soft Inc. is used for collecting daily stock prices
[2] URL: http://www.economagic.com

entrepreneur's age, business location, credit request (first time or not), way of interest payment, grace period, principal payment, repayment period, interest rate, credit amount, planned value of the reinvested profit, cleat vision of business, better than competition, sale on goods and services, advertising, and awareness of competition. The output variable was the credit scoring in the form of a binary variable with one category representing good applicants and the other one representing bad applicants. An applicant is classified as good if there have never been any payments overdue for 45 days or more, and bad if the payment has at least once been overdue for 46 days or more. The sample consisted of 66% goods and 34% bads. The accuracy of models is measured by total hit rates for correctly classified applicants using the threshold 0.5, but individual hit rates for good and bad applicants were also captured. The dataset was divided into the in-sample data (app.75% of data), and the out-of sample data (25% of data) used for the final validation. NNs, logistic regression, and CART decision trees were applied using the same in-sample and out-of-sample data.

6. RESULTS

6.1. RESULTS ON THE STOCK MARKET DATASET

In the stock market dataset the comparison was conducted separately for prediction and classification problem. The best overall NN result is obtained by a 3-layered backpropagation starting with 105, ending with 50 hidden units after pruning (see Table 1). The ATR of 0.841 is accompanied by a very high correlation of 0.9697. Variables V1 (moving average term), V2 (volume), V3 (volatility), V4 (beta), V6 (capital intensiveness), V8 (receivable intensiveness), and V9 (cash position) entered the best model. It is interesting that the forward strategy has selected only stock-related variables and two financial ratios, and none of the macroeconomic variables, indicating their insignificant influence to the stock return rate. Regarding the network topology, our findings are that 1 hidden layer is enough in all architectures to predict stock market return with a very high accuracy (83.5% by modular and 75.3% by radial-basis network). Both modular and radial-basis networks ended with 50 hidden units after pruning.

Table 1: Results of different NN architectures tested on the stock return prediction model

NN architecture	1 hidden layer			2 hidden layers		
	Correlation	Av. trade result	Input vars	Correlation	Av. trade result	Input vars
Backprop	0.9697	0.841*	V1,V2,V3,V4,V6,V8,V9	0.9175	0.771	V1,V2, V4,V5, V6,V8
Radial-basis function	0.9193	0.753	V1,V2,V4	0.9099	0.741	V1,V4
General regression	0.5103	0.488	V1,V4	-	-	
Modular	0.9733	0.835	V1,V2,V4,V6,V8,V9	-	-	

* the best average trade result {ATR) obtained in experiments

The results of the best NN architectures obtained by the four NN algorithms for classification are presented in Table 2.

Table 2: Results of different NN architectures tested on the the stock classification model

NN architecture	1 hidden layer			2 hidden layers		
	Correlation	Av. trade result	Input vars	Correlation	Av. trade result	Input vars
Backprop	0.511974	0.624	V1,V3	0.481883	0.612	V1,V3
Radial-Basis	0.50731	0.629	V1,V4, V6, V8, V10, V14, V16, V17, V19	0.51636	0.629	V1,V3, V4,V5
Probabilistic	0.448204	0.635*	V1,V4	-	-	
LVQ	0.470044	0.529	V1,V3,V4,V6,V9, V10, V11	-	-	

* the best average trade result {ATR) obtained in experiments

The best classification result on the model is obtained by the probabilistic network (ATR of 0.635), followed by radial-basis and backpropagation networks. It consisted of 1 hidden layer with 576 pattern units. The optimal value of parameter sigma was 2.1. The performance of the radial-basis network was not dependent on the number of hidden layers, while the backpropagation was better when only 1 hidden layer is used. Backpropagation and LVQ have selected a relatively large number of input variables, while others have chosen only 2 variables. The best architecture (obtained by the probabilistic network) uses only moving average term and beta as relevant inputs. The results obtained by classification of stock returns are surprisingly worse than by prediction. Although it is expected that the network will more easily identify classes than actual values of stock return, the classification NN algorithms seem to be less efficient. When looking into NN accuracy for each class: positive, hold and negative trade class, the hit rates obtained by the best NN model were the following: 90% for positive trades (buy action), 67.74% for hold trades (hold action), and 84.13% for negative trades (sell action). It can be seen that the individual trade actions accuracy is higher for buy and sell actions, and lower for hold actions. In order to see if NNs perform better than statistical methods, the best overall NN result is compared to the multiple stepwise linear regression. The best ATR obtained by the regression is 0.724 while the total return rate was 162.20. Model fitting was also high (R^2=96.2%) and F-test shows that all the selected variables are significant at the 15% level.

Table 3: Results of the best neural network and multiple regression models

Model	Estimated equation	Av. trade result	Pos. hit rate (buy) %	Hold hit rate (hold) %	Neg. hit rate (sell) %	TRR*
Backpropagation NN (input vars: V1,V2,V3,V4,V6,V8,V9)	-	0.841	90.00	67.74	84.13	171.77
stepwise multiple regression (input vars: V1, V2, V4)	$y = 0.0005 + 1.932 * MA(1) + (- 0.000002) * Volume + (-104.757) * Beta$ $R^2 = 0.96187181, F = 5667.74$	0.724	80.26	41.94	77.78	162.20

* TRR is the total return rate

The results show that NNs outperform multiple regression yielding better average trade result, higher total return rate, as well as higher hit rates for each class of the stocks.

6.2. RESULTS ON THE CREDIT SCORING DATASET

When the suggested strategy for selecting NN architectures is applied on the second dataset, only NN algorithms for classification were tested, since the problem was to classify credit applicants into two categories: "good" and "bad". The backpropagation and radial-basis algorithms were used with additional softmax function in the output layer in order to obtain probabilities. The probabilistic and LVQ algorithms were also tested as standard classifiers. The strategy extracted the best NN architecture for each one of the four algorithms. Logistic regression and CART decision trees were conducted in order to compare the results with NNs (see more detailed description of this research in [21]), and the hit rates of all models obtained on the same out-sample data are shown in Table 4.

Table 4. NN, LR and CART results on the credit scoring dataset

Model	total hit rate (%)	hit rate of bads (%)	hit rate of goods (%)
Backprop NN, 6-50-2	73.80	53.33	85.19
RBFN, 5-50-2	71.40	73.33	70.37
Probabilistic NN, 10-106-2,	83.30*	80.00	85.19
LVQ NN, 2-20-2	61.90	13.33	88.89
Logistic regression	57.14	66.67	51.85
CART	66.67	66.67	66.67

* the highest total hit rate obtained in experiments

As in the first dataset, the overall highest result is obtained by the probabilistic NN model (total hit rate of 83.3%). This network also showed the highest hit rate in classifying bad applicants (bads hit rate of 80%). Among other NN architectures, the LVQ was the worst in perfomance, unable to classify more than 13.33% of bad applicants, while the backpropagation was the second best, followed by the RBFN. The strategy showed that the best structure for the probabilistic NN applied in this dataset was 10 input units and 106 pattern units. Concerning the variable selection, the best NN model extracted 10 input variables as important. The selection of variables showed that both personal and business characteristics are relevant in small business credit scoring systems.

7. CONCLUSION

The motivation for this research was the lack of paradigms in the area of NN methodology for selecting NN architectures, as well as a variety of results obtained by the researchers depending on the problem areas and algorithms used. The paper describes some previous efforts to overcome this limitation, and suggests a non-linear forward strategy for selecting NN architectures that is problem and algorithm independent. The strategy was tested on two independent datasets: the stock market data, and the credit scoring data, using four NN algorithms for prediction and four NN algorithms for classification type of problems. Among different NN architectures for prediction of stock return tested in our research, the most efficient NN architecture selected on the basis of the forward strategy was the 3-layered Backpropagation with 7 input, 50 hidden, and 1 output neurons, hyperbolic tangent transfer function and the EDBD learning rule. The results show that NNs' best result is 11.7% higher than the best stepwise regression's result, which can be considered as an important difference for investors on stock markets. Applied on the credit

scoring dataset, the strategy extracted the probabilistic neural network, with 10 input units and 106 pattern units as the best NN architecture with the hit rate of 83.3%. The model obtained by this architecture showed the best overall result, comparing to the multiple regression and CART decision tree models.

It is obvious that there is no unique method which can be considered the best for prediction or classification. The choice of method, as well as the result depend on the observed data. The fact that NNs give better result on the observed data confirms the advantage of NNs over the multivariant linear regression in stock return data and over the logistic regression in credit scoring data. Although tested on limited data, the suggested selecting strategy enables an extensive test of NN architectures on any theoretical model, therefore providing a useful tool for NN researchers and practitioners.

The research can be further improved by a sensitivity analysis that could bring additional information for effective modelling. The stability of NN results should also be considered, as well as the inclusion of other NN algorithms. The selection of methods can be extended by including other data mining techniques, such as support vector machines, regularization and other algorithms that should be tested in order to find some latent characteristics of the observed data.

REFERENCES

[1] A.Doering, M. Galicki, H. Witte. Structure Optimization of Neural Networks with the A*-algorithm. IEEE Transactions on Neural Networks, Vol. 8, No. 6, 1997, pp. 1434-1445.

[2] D. Enke, S. Thawornwong, The use of data mining and neural networks for forecasting stock market returns, Expert Systems with Applications, Vol. 29, No. 4, 2005, pp. 927-940.

[3] R.A. Jacobs, M.I. Jordan, S. Nowlan, G.E. Hinton. Adaptive Mixtures of Local Experts. Neural Computation, No. 3, 1991, pp. 79-87.

[4] O. Korn, U. Anders. Model selection in neural networks. Neural Networks, Vol. 12, 1999, pp. 309-323.

[5] E.Y. Li. Artificial Neural Networks and Their Business Applications. Information & Management, Vol. 27, 1993, pp. 303-313.

[6] M. Lam. Neural network techniques for financial performance prediction: integrating fundamental and technical analysis. Decision Support Systems, Vol. 37, No. 4, pp. 567-581.

[7] D. Marcek. Neural Network Model for Stock Prices Forecasting. Neural Networks World, No. 3, 1997, pp. 347-352

[8] T. Masters. Advanced Algorithms for Neural Networks. A C++ Sourcebook, John Wiley & Sons, New York, 1995.

[9] T. Masters. Just what are we optimizing anyway?, International Journal of Forecasting, Vol. 14, 1998, pp. 277-290.

[10] NeuralWare. Neural Computing, A Technology Handbook for NeuralWorks Professional II/Plus and NeuralWorks Explorer, NeuralWare, Aspen Technology, Pittsburgh, 1998.

[11] P.T. Quinlan. Structural Change and Development in Real and Artificial Neural Networks. Neural Networks, No. 11, 1998, pp. 577-599.

[12] A.N. Refenes, A.N. Burges, Y. Bentz. Neural Networks in Financial Engineering: A Study in Methodology, IEEE Transactions on Neural Networks, Vol. 8, No. 6, 1997, pp. 1222-1267.

[13] A.N. Refenes, A. Zapranis, G. Francis. Stock Performance Modeling Using Neural Networks: A Comparative Study with Regression Models, Neural Networks, Vol. 7, No. 2, 1994, pp. 375-388.

[14] E.W. Saad, D.V. Prokhorov, D.C. Wunsch. Comparative Study of Stock Trend Prediction Using Time Delay, Recurrent and Probabilistic Neural Networks, IEEE Transactions on Neural Networks, Vol. 9, 1998, pp.1456-1470.

[15] R.R. Trippi, E. Turban. Neural Networks in Finance and Investing. Irwin Professional Publishing, Chicago, 1998.

[16] H. White. Economics Prediction Using NN; The Case of IBM Daily Stock Returns. In R.R. Trippi, E. Turban, ed., Neural Networks in Finance and Investing, Irwin Professional Publishing, Chicago, 1996, pp. 469-481.

[17] B.K. Wong, T.A. Bodnovich, Y. Selvi. Neural Network Applications in Business: A Review and Analysis of the literature (1988-95), Decision Support Systems, Vol. 19, 1997, pp. 301-320.

[18] F. Zahedi. A Meta-Analysis of Financial Applications of Neural Networks, International Journal of Computational Intelligence and Organizations, Vol. 1, No. 3, 1996, pp. 164-178.

[19] M. Zekic. Neural Network Applications in Stock Market Predictions – A Methodology Analysis, Proceedings of the 9th International Conference on Information and Intelligent Systems '98, Varazdin, 1998., pp. 255-263.

[20] M. Zekic-Susac, B. Klicek. A Nonlinear Strategy of Selecting NN Architectures for Stock Return Predictions. Finance, Proceedings from the 50th Anniversary Financial Conference Svishtov, Bulgaria, 11-12 April, 2002, pp. 325-355.

[21] M. Zekic-Susac, N. Sarlija, M. Bensic. Small Business Credit Scoring: A Comparison of Logistic Regression, Neural Network, and Decision Tree Models, Proceedings of the 26th International Conference on Information Technology Interfaces, June 7-10, 2004, Cavtat, pp. 265-270.

Received: 20 January 2005
Accepted: 18 November 2005

INSTRUCTIONS TO AUTHORS

The journal *JIOS: Journal of Information and Organizational Sciences* of the Faculty of Organization and Informatics in Varaždin publishes original scientific papers, preliminary communications, review articlcs, conference papers and professional papers from the field of information and organizational sciences, as well as relevant contributions from related scientific disciplincs. Papers are submitted as follows:

1. Manuscripts, written in English, should be sent to the Editorial Board, primarily submitted by e-mail *(jios@foi.hr)*. The length of a paper should not exceed 20 typed pages of B5(JIS) format. Illustrations of all kind are submitted on separate files in one of the standard graphic file formats (e.g. WMF, BMP, EPS, JPEG, TIFF,...), with marks to indicate their position in the text. The same marks must be included in the text to indicate where the illustrations should be placed.

2. The heading of the paper should inelude: the title of the article, the full name of the author(s), as well as their institutional affiliations and addresses. A short form of the title should also be provided if the original title is rather long.

3. The author(s) should list 3-10 keywords that describe the content of the paper.

4. The author(s) should submit an abstract of the paper (100-150 words) written in English.

5. All papers that are considered for publication are reviewed by two independent referees.

6. References are indicated in the text by Arabic numerals which are enclosed in square brackets, e.g. [1]. Page umber(s) may be added if necessary, e.g. [1, p. 25] or [1, pp. 25-26]. Full citation of the literature referred to should be given at the end of each article. The bibliography should have the heading REFERENCES. Bibliographical references are to be listed in alphabetical order by surnames of authors and should be preceded by Arabic numerals in square brackets. References should take the following form:

REFERENCES

MONOGRAPHS

[1] R. E. Blahut. Digital Transmission of Information. Addison-Wesley, Menlo Park, 1990.

COLLECTIONS

[2] H. Fiedler. Functional. reiations between legal requirements and software. In B. Nibett, ed. Computer Science and Law. Cambridge University Press, Cambridge, 1990, pp. 137-146.

JOURNAL ARTICLES

[3] Small. A functional approach to database updates. Information Svstems, Vol. 18, No. 3, 1993, pp. 581-589.

Address of the Editorial Office:

Faculty of Organization and Informatics
JIOS Journal, Editorial Board, 42000 Varaždin, Pavlinska 2, CROATIA
Phone: +385-42-213 -777
Fax: +385-42-213-413
E-mail: jios@foi.hr

Text processing

Mirko Čubrilo

Proofreading

Authors

Printing

Faculty of Organization and Informatics, Varaždin, Croatia
JIOS is published with financial support of the Ministiy of Science and Technology of the Republic of Croatia.

Address of the Editorial Office

Faculty of Organization and Informatics, Varaždin
Pavlinska 2, 42 000 Varaždin
Croatia
Tel. +385 42 213 777, fax +385 42 213 413, E-mail: jios@foi.hr

---