Biblioteka Informatyki
Szkół Wyższych

Information Systems Architecture and Technology

Web Engineering and High-Performance Computing on Complex Environments

Library of Informatics of University Level Schools

Series of editions under the auspices
of the Ministry of Science and Higher Education

The ISAT series is devoted to the publication of original research books in the areas of contemporary computer and management sciences. Its aim is to show research progress and efficiently disseminate current results in these fields in a commonly edited printed form. The topical scope of ISAT spans the wide spectrum of informatics and management systems problems from fundamental theoretical topics to the fresh and new coming issues and applications introducing future research and development challenges.

The Library is a sequel to the series of books including Multidisciplinary Digital Systems, Techniques and Methods of Distributed Data Processing, as well as Problems of Designing, Implementation and Exploitation of Data Bases from 1986 to 1990.

Wrocław University of Technology

Information Systems Architecture and Technology

Web Engineering and High-Performance Computing on Complex Environments

Editors
Leszek Borzemski
Adam Grzech
Jerzy Świątek
Zofia Wilimowska

Wrocław 2012

Publication partly supported by
Faculty of Computer Science and Management
Wrocław University of Technology

Project editor
Arkadiusz GÓRSKI

The book has been printed in the camera ready form

OFICYNA WYDAWNICZA POLITECHNIKI WROCŁAWSKIEJ
Wybrzeże Wyspiańskiego 27, 50-370 Wrocław
http://www.oficyna.pwr.wroc.pl;
e-mail: oficwyd@pwr.wroc.pl
zamawianie.ksiazek@pwr.wroc.pl

ISBN 978-83-7493-702-3

CONTENTS

INTRODUCTION

This book consists of 14 chapters presenting a balanced coverage of challenges of current IT issues in Web Engineering and High-Performance Computing on Complex Environments.

Web Engineering is a scientific discipline that studies the theory and practice of constructing Web-based applications. The World Wide Web (the Web) has come to be the principal place for any information, data and applications. It becomes clear that the construction and evolution of the World Wide Web requires support of systematic, disciplined and quantifiable approaches that are developed in Web Engineering research. The activities of Web Engineering are focused in the cost-effective development, operation, and evolution of high-quality applications in the World Wide Web.

This book includes eleven chapters presenting selected issues related to following areas of Web Engineering:

- Web performance prediction
- Web load balancing
- Web navigation
- Component-based Web application development
- Web application usability
- Cloud computing
- Rapid development of Web applications
- Web application architectures
- Text mining
- Content retrieval and search
- Human-computer interaction

High-Performance Computing on Complex Environments (ComplexHPC) explodes following development of new technologies in computer systems to solve complex and challenging problems with high computational cost including multiprocessor and multi-core computers, GPUs (Graphic Processing Units), heterogeneous and hierarchical computer environments.

Specifically, this book presents GPU-based computations in complex environments and program scheduling in multiprocessing computer system.

The book opens with the chapter titled *Comparison of Turning Bands Method and Sequential Gaussian Simulation in Daily Analyses of Web Servers' Performance*

which presents a comparison of two geostatistical simulation methods: Turning Bands Method (TB) and Sequential Gaussian Simulation (SGS) in making daily analyses and spatio-temporal forecasts of web servers' performance. TB and SGS are novel approaches proposed by the authors to predict web performance. The analyses have been made for data measured by MWING – a Multiagent Internet Measurement System. One of MWING's agent located in Gdańsk downloaded a specific file from fifteen European web servers six times per day, at intervals of three hours, beginning at 06:00 am, during the period of February, 2009. First, preliminary and structural analyses of input data were made. Subsequently, four-day ahead spatio-temporal forecasts of downloading times from evaluated servers were carried out using TB and SGS. The results were analyzed to draw conclusions about the impact of time of day and the selection of method on forecasting.

In the following chapter, *Load Balancing in the Current Internet at the DNS Level*, the authors present the results of their experimental research to show how load balancing in now supported by Domain Name System (DNS) infrastructure. Web load balancing done by DNS infrastructure is one of the most popular ways to build a performance scalable website. DNS can assign different IP addresses (meaning different hosts) to the same domain name, splitting up the traffic already at the very first phase of the Web transaction. To test this system working in the current Internet, a computer workstation was constructed, thus allowing examining almost 3 million Web sites few times. For every DNS address numerous of IP addresses were collected. It was found that some sites are hosted on multiple servers to which requests can be routed. This examination allowed checking how often such mechanism is used in the Internet, as well as how many servers can be detected at the DNS level to host a single website. New phenomena were discovered related to the DNS mechanism, including variability of returned IP addresses.

The next chapter, *Changing the Website Navigation Structure*, deals with the website usability problem as seen from the point of view of the website navigation structure which is an essential tool for user interaction with the website. As users interact with the website, the usage statistics can be collected with an online service. In existing website usability metrics, the measurement of how usable is the navigation structure is not commonly included. The author proposes to develop a metric, called the energy of a network, to assess the usability of the website navigation structure. There were taken into account such characteristics as the availability of every page in the navigation structure, the structure of hyperlinks, and usage data of navigation structure. After valuation of the website structure with the energy of a network metric, it is possible to decide if to maintain or change the website navigation structure. The decision task on changing the website navigation structure is crucial in the task of designing the web interface.

The chapter that follows is titled *Web Component for Automatic Extraction of Ontological Information from Informal Description of Web Services* and treats Web

Services development. It describes the semantic methods that can be used to create the description that is comprehensible for computers. It also presents two models supporting the automatic generation of the Web Services semantic description based on informal description. The chapter draws upon the comparison of two languages, which can be used while defining the semantic description of the Web Services and presents the way of creating, developing and using the ontology in the Web Services repositories.

In the next chapter, titled *A Method for Comparing Efficiency of the Different Usability Evaluation Techniques*, the author presents a method for comparing efficiency of different usability techniques. While performing a thorough usability audit of a particular website different usability technique such as expert evaluation, focus groups, clicktracking or eyetracking. To compare different usability methods a formal representation of a method's properties was proposed. After performing a usability evaluation it was possible to assign the obtained data, such as number of usability problems found on the website, the importance of those problems, cost and time, to the method properties model. After that, it was possible to compare models under study and show which of the used techniques are more effective for the particular web system.

The chapter that follows is titled *Automatic the Virtual Private Cloud Creation with the use of Web Services and Workflows* and presents a method for cloud service orchestration with using of workflows, which efficiently scale out administrative workload of private cloud creation. Presented solution utilizes the VMware API orchestrator's workflows and web services in order to provide the interface to self-service environment of business application systems. A concept presented in this chapter is to enable the dynamic placement of multi-tier services on public or private cloud infrastructures.

The next chapter, titled *Rapid Application Development with UML and Spring Roo*, presents an approach to evolutionary rapid prototyping of data-intensive web applications. The main idea behind the approach is to combine the benefits of UML modeling with fast source code generation for specific platform. Model-Driven Development (MDD) and Domain Specific Languages (DSLs) are becoming more popular last years. These techniques try to maximize the benefits of modeling in many ways, e.g. by eliminating the gap between analytic and design models, and by producing working code directly from models. In the chapter, an approach to combine classical, visual modeling with UML (preferred by system analysts) with the textual Spring Roo DSL (used by developers) is proposed. The approach aims at rapid development of data-oriented web applications, in which the main functionalities allow to create, delete, update, and retrieve both objects, and links between them. The aspect of user authentication and authorization is also taken into account.

The following chapter titled *Feature-Based Architecture Reviews* deals with information system architecture assessment methods and introduces the Feature-Based Architecture Reviews Method that has been elaborated to overcome problems known in the scenario-oriented methods. The scope of the analysis is defined by a set of

architecturally relevant software features. Each of these features is addressed with architectural decisions. These decisions, in turn, may cause risks concerning the system's quality attributes. The method scales very well, as any set of software features can be assessed, and so it scales from assessing just a single feature to a fully comprehensive architecture review. The method integrates naturally with RUP or agile methodologies.

The next chapter titled *Automatic Correction of Errors in Polish Texts* presents an approach to detection and correction of errors in computerized edition of texts in Polish. Modified Link Grammar equipped with inflection related linking requirements is proposed. The process of error correction and detection consists of three stages. First, erroneous word is identified and then possible correction candidate words are generated. To limit the number of correction alternatives some methods based on word statistics or technical cause of error may be used. In last stage, word dependencies are used to select the word best matched in given context. Proposed method may be used as supplement in existing text editors. It may be also used for preliminary test analysis in automated text processing systems (e.g. information extraction systems).

In the following chapter titled *Applying Fuzzy Logic to Decisional DNA Digital TV*, the authors introduce application of fuzzy logic methods to the Decisional DNA Digital TV. The integration of the Decisional DNA DTV and fuzzy logic provides the Digital TV viewer with better user experience. Decisional DNA is a domain-independent, flexible, and standard experiential knowledge representation structure that allows its domains to acquire, reuse, evolve, and share knowledge in an easy and standard way. The Decisional DNA DTV enables TV players to learn the viewer's watching habit discovered through past viewing experience and reuse such experience in suggestion of channels. The presented conceptual approach demonstrates how the Decisional DNA-based systems can be integrated with fuzzy logic technique, and how it captures and deals with the TV viewer's watching experience in a fuzzy logic way.

The next chapter, *New Content-Based Indexing Algorithm in Automatic Video Indexer AVI*, presents Automatic Video Indexer AVI research project investigating tools and techniques of automatic video indexing for retrieval systems. The main goal of the project AVI is to develop efficient algorithms of content-based video retrieval. Several strategies have been proposed, implemented and tested, and they are still being intensively developed. The most simple techniques are based on the comparison of video frames histograms. The most advanced approaches use different algorithms of content analysis based on image recognition and artificial intelligence.

The next chapter titled *Astronomical Photometric Data Reduction Using GPGPU* is concerned with the high-performance computing in complex applications, and the authors present a method that uses Graphic Processing Units for data reduction in astronomical data processing. The graphics processor that in its beginning aimed at fast screen image computation and presentation naturally adopt SIMD model of processing. This model fits very well in the reduction process of the contemporary

photometric data received with the use of CCD cameras that are in the two-dimensional form. The chapter presents the library for the photometric data reduction that uses flat field reduction, dark and bias current reduction with the use of CUDA (Compute Unified Device Architecture) environment, which enables to pass the computation onto graphics processors.

The following chapter titled *Efficiency of Parallelization of Neural Network Algorithm on Graphic Cards* also is concerned with HPC on CUDA-based GPUs. The chapter shows how the run-time layer of CUDA technology can be exploited in speeding up calculations. Because of differences in architectures of systems, running sequential and parallel versions of applications there was necessity to redefine the original definition of efficiency to compare the heterogeneous systems. The authors tested their solutions on selected graphics cards with CUDA capability running two parallelized neural network learning algorithms. Input data for neural network were global features extracted from histopathological images.

The next chapter, *Programs Scheduling in Multiprocessing Computer System with Position Dependent Processing Times*, presents results of research on the problem of time-optimal programs scheduling and primary memory pages allocation in multiprocessing computer system when task processing times are position dependent. Heuristic algorithm to minimize schedule length is proposed and evaluated in some computational experiments.

The last chapter, *The Estimation of Remotely Monitored Network Service Execution Parameters*, presents a mechanism for monitoring of network services with use of analysis of service request processing on TCP session level. The presented method permits to estimate values of some non-functional service parameters on remote server. There are considered synchronous services that are commonly used in Service Oriented Architecture-based systems. The chapter also presents results of experiments performed in network environment that show effectiveness of described method.

This book contains the contributions accepted after the review of authors' submissions. We hope that the book will be considered as a forum for presentation of original work in up-to-date research areas in Web systems, Internet, software engineering, information systems design paradigms and high performance processing on hybrid architectures.

We would like to express many thanks to revisers who helped to evaluate the submissions.

We thank all the authors who have submitted their contributions to be published in this book.

Wrocław, September 2012

Leszek Borzemski

Turning Bands Method, Sequential Gaussian Simulation,
spatio-temporal forecasts of web servers' performance

Leszek BORZEMSKI, Michał DANIELAK,
Anna KAMIŃSKA-CHUCHMAŁA*

COMPARISON OF TURNING BANDS METHOD AND SEQUENTIAL GAUSSIAN SIMULATION IN DAILY ANALYSES OF WEB SERVERS' PERFORMANCE

This research is a comparison of two geostatistical simulation methods: Turning Bands Method (TB) and Sequential Gaussian Simulation (SGS) in making daily analyses and spatio-temporal forecasts of web servers' performance. The historical data, essential to conduct forecasts, were obtained using the Multiagent Internet Measurement System (MWING). Namely one of MWING's agents (located in Gdansk) had been continuously trying to obtain the same resource (i.e. RFC text file) from fifteen European web servers. The measurements of resource download times were made six times per day, at intervals of three hours, beginning at 06:00 am, during the period of February, 2009. First, preliminary and structural analyses of input data were made. Subsequently, four day ahead spatio-temporal forecasts of downloading times from evaluated servers were carried out using TB and SGS. Then, obtained results were analysed in detail to draw conclusions about the impact of time of day and the selection of method on forecasting.

1. INTRODUCTION

The amount of traffic generated on the Internet continuous to grow. In the nineties, only a few households were connected to the network. This situation, however, has changed since the Internet has become ubiquitous. Not only have modern households more than one network-connected device, but companies also commenced to support BYOD model (bring your own device). This brings about considerable number of devices generating enormous network traffic (especially in the Web) and makes IT administrators snowed under with their job.

* Institute of Informatics, Wrocław University of Technology,
{leszek.borzemski, michal.danielak, anna.kaminska-chuchmala}@pwr.wroc.pl

To deliver quality-based services, administrators not only need to constantly monitor their resources, but also try to predict possible situations. This paper presents geostatistical approaches, **namely** Turning Bands and Sequential Gaussian Simulation as a solution to that problem, because these methods have already proven themselves in computer science [5], [6]. These approaches allow to make a spatio-temporal forecasts using only historical data, gathered during daily servers monitoring. To put in a nutshell, we have collected the data concerning performance of fifteen evaluated European web servers, between 06 and 28 February, 2009. A server's performance in this case corresponds to the time required to download the resource from the server. At the outset, these data were subjected to thorough analyses; subsequently, they were used to forecast evaluated web servers' performance from 1 to 4 March, 2009.

Section 2 and 3 briefly explain Sequential Gaussian Simulation and Turning Bands methods respectively. Section 4 presents preliminary analysis of data (such as basics statistics of historical data of evaluated servers) while section 5 shows structural analysis of data (i.e. directional variograms and their models). Finally, sections 6 and 7 present obtained results and conclusions, and propose future research directions.

2. SEQUENTIAL GAUSSIAN SIMULATION

The Sequential Gaussian Simulation is one of the most simple methods for simulating a multivariate Gaussian field. Each value is simulated sequentially, according to its normal conditional cumulative distribution function which must be determined at each location to be simulated. The conditioning data comprise all the original data and all previously simulated values within the neighbourhood of the point being simulated. The Sequential Gaussian Simulation starts with the assumption that the kriging error is normally distributed, with variance $\sigma_K^2(x_0)$ and mean equal to 0 which can be described as $N\left(0, \sigma_K^2(x_0)\right)$. In these circumstances $N\left(\overline{Z}, (x_0), \sigma_K^2(x_0)\right)$ is the probability distribution for actual data [4].

SGS algorithm can be presented in the following way:

1. Ensure that data are approximately normal; if necessary, transform data to standard normal distribution.
2. Calculate and model variogram.
3. Specify the coordinates of points to be simulated.
   Determine the sequence, in which points $x_j (j = 1,2,...)$, will be visited in the simulation. To maximise the diversity of different realizations, choose points randomly.

4. Simulate at each of these points as follows.
   a) Use simple kriging with the variogram model to obtain $\overline{Z}(x_i)$ and $\sigma_K^2(x_i)$.
   b) Draw a value from a normal distribution $N\left(\overline{Z},(x_i),\sigma_K^2(x_i)\right)$.
   c) Insert drawn value into the grid at x_i, and then add it to data.
   d) Proceed to the next node and simulate the value at this point in a grid.
   e) Repeat steps a) to c) until all of the nodes have been simulated.
5. Transform back the simulated values (using Gaussian Anamorphosis) if necessary.

More information about SGS method can be found in [7], [8].

3. TURNING BANDS METHOD

The Turning Bands method, originally initiated by Matheron, is stereological tool that allows to reduce multidimensional simulation to one-dimensional [10], [11].

A stationary Gaussian random function with mean equal to 0, variance equal to 1 and covariance C that is continuous in $D \in R^d$. According to the Bochner's theorem, covariance C can be define as the Fourier transform of positive measure, for instance χ:

$$C(h) = \int_{R^d} e^{i(h,u)} d\chi(u) \tag{1}$$

Also $C(0) = 1$, so χ is a measure of the probability. After the introduction of the polar coordinate system $u = (\theta, \rho)$, where θ is the directional parameter of the hemisphere S_d^+ and ρ is the location parameter $(-\infty < \rho < \infty)$ spectral measure $d(u)$ can be expressed as the product of decomposition $d\varpi(\theta)$ and conditional distribution $d_{\chi\theta}(\rho)$ for a given θ. After using this distribution to develop the spectral covariance C and the introduction of one-dimensional function $C_\theta(r)$ Bochner's theorem was used, so that the covariance function $C(h)$ can be expressed as:

$$C(h) = \int_{S_d^+} C_\theta((h,\theta)) d\varpi(\theta) \tag{2}$$

where C_θ is also a covariance. Therefore TB consists in reducing the simulation of a Gaussian function with covariance C to the simulation of an independent stochastic process with covariance $C(h)$.

Let $(\theta_n, n \in N)$ be a sequence of directions S_d^+ and let $(X_n, n \in N)$ be a sequence of independent stochastic processes of covariance C_{θ_n}. Then random function:

$$C^n(h) = \frac{1}{\sqrt{n}} \sum_{k=1}^{n} X_k((x, \theta_k)), x \in R \tag{3}$$

takes covariance that is equal to:

$$C^n(h) = \frac{1}{n} \sum_{k=1}^{n} C_{\theta_k}((h, \theta_k)) \tag{4}$$

The central limit theorem shows that for very large $n, Y(n)$ tends to Gaussian distribution with variance $\lim_{n\to\infty} C^n$. When series $\frac{1}{n}\sum_{k=1}^{n}\delta_{\theta_k}$ converges weakly to ϖ; this limit is exactly C.

Turning Bands algorithm may be presented in the following way:

1. Transform input data using Gaussian anamorphosis.
2. Select directions $\theta_1,...,\theta_n$ so that $\frac{1}{n}\sum_{k=1}^{n}\delta_{\theta_k} \approx \varpi$.
3. Generate standard, independent stochastic processes $X_1,...,X_n$ with covariance functions $C_{\theta_1},...,C_{\theta_n}$.
4. Calculate $\frac{1}{\sqrt{n}}\sum_{k=1}^{n} X_k((x, \theta_k))$ for every $x \in D$.
5. Make kriged estimate $y^*(x) = \sum_c \lambda_c(x) y(c)$ for each $x \in D$.
6. Simulate a Gaussian random function with mean 0, covariance C in domain D on condition points. Let $(z(c), c \in C)$ and $(z(x), x \in D)$ be the obtained results.
7. Make kriged estimate $z^*(x) = \sum_c \lambda_c(x) z(c)$ for each $x \in D$.
8. Obtain the random function $W(x) = (y^*(x) + z(x) - z^*(x), x \in D)$ as the result of conditional simulation.
9. Perform a Gaussian back transformation to return to the original data.

TB and conditional simulations are discussed in more detail in [9], [12].

4. PRELIMINARY ANALYSIS OF DATA

To successfully perform forecasts and daily analyses of web servers' performance, it is necessary to create database containing historical performance data of evaluated servers. To achieve this, measurements obtained using Multiagent Internet Measuring System (MWING) were used. This system consists of many distributed throughout the world agents – computer systems equipped with software designed for making measurements. Their main task consists in measuring times needed to download a copy of the same resource (in this case it is a text document – Request for Comments file). Detailed description of MWING system can be found in [1], [2] and [3].

In this paper, the used measurements were taken by the agent located in Gdańsk, Poland. The agent had been querying fifteen European web servers six times a day with a three-hour interval, starting at 6:00 am, every day between 07 February 2009 and 28 February 2009. Then, the obtained results and information such as servers' locations (i.e. their latitudes and longitudes), timestamps of measurements were used alongside to create the aforementioned database.

Table 1 presents basic statistics of Web performance for considered servers. The largest data span occurs for 06:00 am where the difference between minimum and maximum value is 28.95 seconds; for the sake of comparison, data span for 09:00 pm equals only 1.5 seconds. After a thorough analysis, it turned out that the lowest performance of evaluated web servers could be observed at 12:00pm. This happened because at that time most people were at work trying to obtain many Web resources, generating a surge of network traffic. A different scenario occurred at 09:00 pm when substantially less network traffic was generated and consequently evaluated server worked more efficiently.

Moreover, high value of kurtosis (more than 3) indicate the great variability of the examined process for each hour except 09:00 pm. Taking into account both high skewness and the fact that the whole idea consists in achieving a distribution as close as possible to a symmetric distribution, logarithmic values of obtained data were calculated for all hours, except 09:00 pm.

Table 1. Basic statistics of download times from evaluated European web servers, taken between 07.02.2009 and 28.02.2009

Statistical parameter	06:00 am	09:00 am	12:00 pm	03:00 pm	06:00 pm	09:00 pm
Minimum value X_{min}[s]	0.11	0.12	0.12	0.09	0.12	0.11
Maximum value X_{max} [s]	29.06	10.33	12.15	5.00	7.93	1.61
Average value X [s]	0.60	0.54	0.62	0.46	0.60	0.47
Standard deviation S [s]	1.59	0.66	1.08	0.37	0.77	0.31
Variability coefficient V [%]	266	123	174	80	129	67
Skewness coefficient G	15.35	10.98	7.25	6.61	4.99	2.61
Kurtosis coefficient K	265.65	156.72	64.16	76.29	34.61	7.34

Figure 1 presents histograms of download times for 09:00 am before and after the calculation of logarithms. Before the calculation of logarithms (a), histogram was asymmetric, single-wing, and positively skewed; this indicates the large variation of input data. After the calculation (b), however, the histogram had a shape slightly

similar to a symmetric distribution. This allows to perform more accurate forecasts and analyses of evaluated web servers performance.

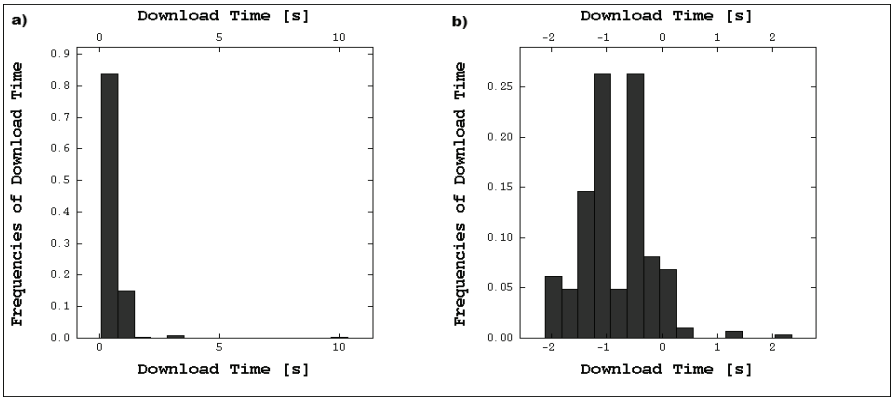


Fig. 1. Histograms of web-servers' performance for 09:00 am

5. STRUCTURAL ANALYSIS OF DATA

Calculation of Gaussian anamorphosis is the first step in the structural analysis of data. To calculate Gaussian transformation frequency, the inversion model was used and the number of adopted Hermite polynomials was equal to 100.

The next step in structural data analysis is modeling of a theoretical variogram function. Directional variogram was calculated along the time axis (for 90^o direction). Table 2 presents the best basic structures with their distance classes, used to model variograms for every considered hour. These structures are the best that we have managed to get so far.

Table 2. Approximated theoretical variograms with their distance classes for every evaluated hour

	Method used	Basic structures used to model the variogram	Distance class [^{o}]
06:00 am	TB	J-Bessel, nugget effect	5.69
	SGS	K-Bessel, nugget effect	8.66
09:00 am	TB	J-Bessel, nugget effect	4.33
	SGS	J-Bessel, nugget effect	4.33
12:00 pm	TB	K-Bessel, nugget effect	7.76
	SGS	Gaussian function, nugget effect	9.33
03:00 pm	TB	K-Bessel, nugget effect	5.73
	SGS	K-Bessel, nugget effect	5.73
06:00 pm	TB	K-Bessel, nugget effect	4.34
	SGS	K-Bessel, nugget effect	3.93
09:00 pm	TB	K-Bessel, nugget effect	6.59
	SGS	K-Bessel, nugget effect	6.59

Figure 2 illustrates variograms of web servers' performance for 09:00 am (a) and 09:00 pm (b). The variograms were approximated by the models of nugget effect and J-Bessel, and nugget effect and K-Bessel for 09:00 am and 09:00 pm respectively.

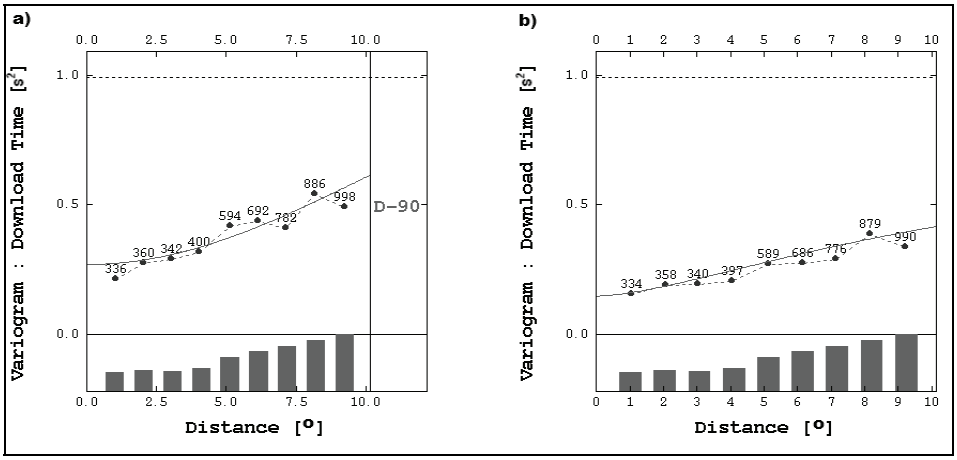


Fig. 2. Directional variogram along the time axis, of web servers' performance for 09:00 am (a) and 09:00 pm (b)

6. DAILY ANALYSES OF WEB SERVERS' PERFORMANCE

Table 3 and 4 present global statistics for four-day forecasts of daily web servers' performance, made using TB and SGS. Based on the obtained results, it can be stated that web servers' performance was generally the lowest at 12:00 pm when the average times needed to obtain resource from evaluated servers were equal to 0.48 and 0.47 for TB and SGS respectively. Table 5 presents mean forecasts errors of web servers' performance for all considered hours. Generally, the results obtained with SGS are slightly better than those obtained using TB.

Table 3. Global statistics for the four-day forecasts of daily web servers' performance, made using TB method

Geostatistical parameter / Mean forecasted value Z for:	Min. value Z_{min} [s]	Max. value Z_{max} [s]	Average value Z [s]	Variance S^2 [s]^2	Standard deviation Z [s]	Variance coefficient V [%]
06:00 am	0.15	1.04	0.45	0.03	0.16	36
09:00 am	0.13	1.59	0.47	0.06	0.23	49
12:00 pm	0.15	1.28	0.48	0.03	0.16	35
03:00 pm	0.12	1.27	0.45	0.05	0.22	48
06:00 pm	0.14	1.61	0.45	0.04	0.20	45
09:00 am	0.18	1.60	0.54	0.06	0.23	44

Table 4. Global statistics for the four-day forecasts of daily web servers' performance, made using SGS method

Geostatistical parameter / Mean forecasted value Z for:	Min. value Z_{min} [s]	Max. value Z_{max} [s]	Average value Z [s]	Variance S^2 [s]^2	Standard deviation Z [s]	Variance coefficient V [%]
06:00 am	0.11	11.15	0.45	0.03	0.16	36
09:00 am	0.12	7.19	0.46	0.03	0.18	39
12:00 pm	0.12	11.72	0.47	0.03	0.18	38
03:00 pm	0.10	3.32	0.39	0.02	0.12	32
06:00 pm	0.12	7.02	0.45	0.03	0.17	38
09:00 pm	1.12	4.98	1.67	0.03	0.17	37

Table 5. Mean forecasts errors for web servers' performance forecasts, conducted using TB and SGS

	06:00 am	09:00 am	12:00 pm	03:00 pm	06:00 pm	09:00 pm
TB	26.91%	29.43 %	20.00%	25.03%	17.55%	28.73%
SGS	24.83%	26.31 %	16.06%	22.31%	18.53%	27.43%

Figure 3 presents actual and forecasted performance, calculated using TB, of the server located in Strasbourg. Some regularity may be observed not only in actual, but also in forecasted data. Nevertheless, the measurements taken on 4 March, at 06:00 am is an exception – due to connection problems measured value was almost three times higher than on other days. But if one strips this day out, the mean forecast error for the whole considered period was 24.96.

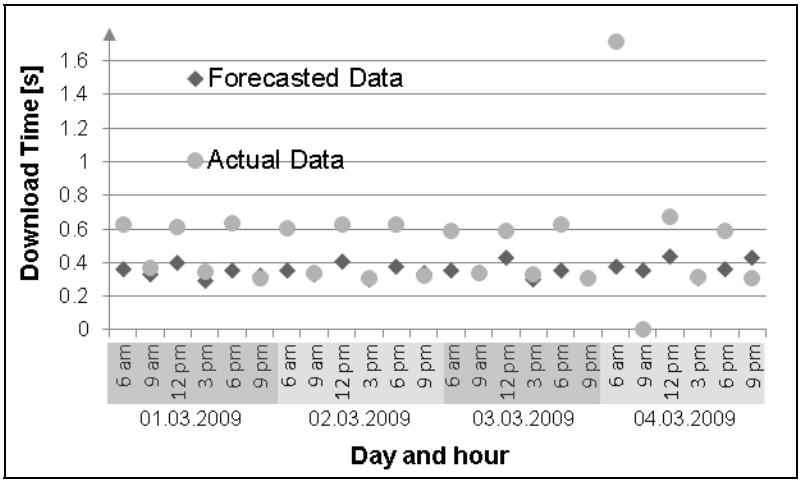


Fig. 3. Actual and forecasted performance of Strasbourg's server, calculated using TB

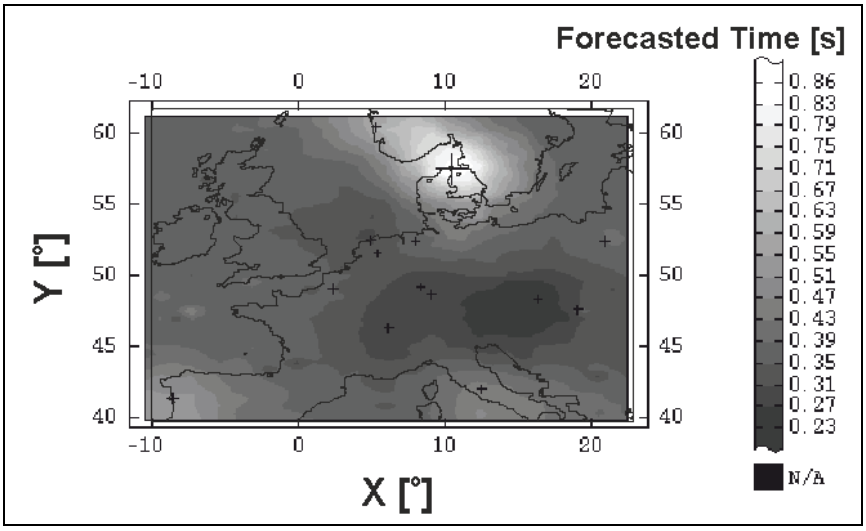


Fig. 4. Forecasted web servers' performance, calculated using SGS, for 1 March 2009, at 03:00 pm

The forecasted web servers' performance for the whole considered area for 03:00 pm is presented as sample raster map in figure 4. Crosses shown on the map represent examined servers and the size of these crosses corresponds to actual web server's performance – the larger the cross, the lower the performance of a server. The server with the lowest performance was located in Frederikshavn, Denmark.

7. CONCLUSIONS

This paper presented TB and SGS in making daily analyses and spatio-temporal forecasts of web servers' performance. Such analyses and forecasts may be very helpful for IT administrators, especially in analysing both network traffic and web servers performance. What is more, the obtained results justify the usage of both of these methods in making daily analyses and forecasts of web servers' performance.

Nevertheless, it can be stated that there is still a need to improve the accuracy of forecasts, especially those carried out using TB. This could be achieved by making forecasts in different scenarios, varying in the type of measured values, their timestamps, and the length of time horizons.

REFERENCES

[1] BORZEMSKI L., CICHOCKI L., KLIBER M., FRAS M., NOWAK Z., *MWING: a multiagent system for Web site measurements*, In: Lecture Notes in Computer Science, 4496, 2002, 278–287.

[2] BORZEMSKI L., CICHOCKI L., KLIBER M., *A distributed system to measure the Internet based on agent architecture*, In: Information systems architecture and technology, Web-age information systems, eds Leszek Borzemski [et al.], Oficyna Wydawnicza Politechniki Wrocławskiej, 2009.

[3] BORZEMSKI L., NOWAK Z., *Empirical Web performance evaluation with using a MWING system*, In: Information systems architecture and technology: advances in Web-Age Information Systems, eds: Leszek Borzemski [et al.], Oficyna Wydawnicza Politechniki Wroclawskiej, 2009, 25–34.

[4] BORZEMSKI L., KAMINSKA-CHUCHMALA A., *Knowledge Engineering Relating to Spatial Web Performance Forecasting with Sequential Gaussian Simulation Method*, Lecture Notes in Artificial Intelligence, 2012 (in print)

[5] BORZEMSKI L., KAMINSKA-CHUCHMALA A., *Knowledge Discovery about Web Performance with Geostatistical Turning Bands Method*, In: Knowlege-Based and Intelligent Information and Engineering Systems Lecture Notes in Computer Science, Volume 6882/2011, DOI: 10.1007/978-3-642-23863-5_59, 2011

[6] BORZEMSKI L., KAMINSKA-CHUCHMALA A., *Distributed Web Systems Performance Forecasting Using Turning Bands Method*, In: IEEE Transactions of Industrial Informatics, 2012

[7] HICKS P.J., *Unconditional sequential Gaussian simulation for 3-D flow in a heterogeneous core*, Journal of Petroleum Science and Engineering 16, 1996, 209–219.

[8] KING S.L., *Sequential Gaussian simulation vs. simulated annealing for locating pockets of high-value commercial trees in Pennsylvania*, Annals of Operations Research 95, 2000, 177–203

[9] LANTUEJOUL Ch., *Geostatistical Simulation: Models and Algorithms*, Springer-Verlag, 2002.

[10] MATHERON G., *Quelques aspects de la montée*, Internal Report N-271, Centre de Morphologie Mathematique, Fontainebleau., 2002.

[11] MATHERON G., *The intrinsic random functions and their applications*, In: JSTOR Advances in Applied Probability, Vol. 5, 1973, 439–468.

[12] WACKERNAGEL H., *Multivariate Geostatistics: an Introduction with Applications*, Springer, 2003

DNS, Web, load balancing,
traffic distribution, routing

Marcin STĘPNIAK, Tomasz SALWA, Ziemowit NOWAK*

WEB LOAD BALANCING AT DNS LEVEL IN 2012

In this chapter methods of Web servers' load balancing with the DNS support are described. In 1994 when first Web sites began to face very high traffic, it became obvious that single server for a site will not be sufficient to handle growing numbers of requests. Traffic volume had to be divided into multiple globally and locally distributed Web servers. The problem was how to ensure that. One of the solution involved Domain Name System which was suppose to assign different IP addresses (meaning different hosts) to the same domain name, splitting up the traffic load already at the very first phase of the Web transaction. Additionally, Content Delivery Network providers also utilize DNS redirection. To test this system working in the current Internet, a computer workstation was constructed, thus allowing to examine almost 3 million Web sites few times. For every DNS address numerous of IP addresses were collected. Therefore it was found that some sites are hosted on multiple servers to which requests can be routed. This examination allowed to check how often such mechanism is used in the Internet, as well as how many servers can be detected at the DNS level to host a single Web site. Moreover, few phenomena were discovered related to the DNS mechanism, including variability of returned IP addresses.

1. LOAD BALANCING

1.1. IMPORTANCE OF TRAFFIC DISTRIBUTION

Each server that hosts a Web site has capacity. Even though nowadays technology is developing very quickly and servers can be provided with the newest and the fastest hardware possible, they will still have limits that cannot be exceeded. Therefore, to extend that limit and allow more users to access a Web site, it has to be hosted on multiple servers [3, 4, 9, 10].

* Institute of Informatics, Technical University of Wrocław, Wybrzeże Wyspiańskiego 27, 50-370 Wrocław.

The problem that emerged was balancing the traffic volume for these servers. Ideally, each server should handle the same amount of users and each user should be redirected to the server that can answer the quickest for him. In that problem though resources of the hosting machines come into play as well as distance between the user and the host, especially considering globally distributed Web servers. When talking about load balancing, meaning of the word "load" needs to be determined. Secondly, there has to be a method of choosing the best host for the user. This caused the need for effective traffic distribution systems [5, 11].

Results of the research that took place in April and May 2012 show mechanisms which answer above problem. Moreover, effectiveness of these mechanisms is tested. This research extends previous studies on that subject done by Department of Distributed Computer Systems of Wrocław University of Technology [1, 2].

1.2. DNS MECHANISMS

In the RFC 1794 DNS Support for Load Balancing few of the criteria that must be fulfilled are described. These criteria correspond to the overall requirements for load balancing in the Internet. They are as follows: backward compatibility with the existing DNS RFC, information changes frequently, multiple addresses must be send out, must interact with other RRs appropriately, must be able to represent many types of "loads", must be fast [8].

DNS allows to send out multiple IP addresses linked to one DNS address [9]. These addresses changes accordingly to the location in the Internet, from which DNS query is received, and time [3]. Multiple requests from one computer to resolve single DNS address may give different results, both with completely new address pool as well as same pool but with other first answer. The latter inherently supports Round Robin system.

Another significant system in load balancing is Content Delivery Network which uses DNS redirection and provides possibility to utilize globally distributed servers [7]. One of the approach in CDN is to create surrogate servers that perform entire replication of the content. Then DNS is configured by a content provider to allow all requests to be resolved by a CDN server. In that way the latter delivers content to the end users [9]. Content Delivery Network is also utilized by most of video sharing services [11].

As explained in the previous paragraphs, many methods of load balancing are implemented already at the DNS level. Various answers from the Domain Name Server are possible thus redirecting clients to the different servers. Answers can contain one or many IP addresses from which one is chosen every time request is sent. Additionally, queries from different locations or sent at different time may result in entirely different address or addresses. All of these phenomena are effects of described mechanisms.

2. RESEARCH

2.1. PURPOSE OF THE RESEARCH

The Internet has grown and nowadays many Web sites are being visited by such amount of users that traffic distribution is essential [3, 6]. DNS introduced support for that many years ago. The following research tries to answer the question: To what extend current Internet is filled with DNS mechanisms supporting load balancing?

2.2. COMPUTER WORKSTATIONS

One computer workstation dedicated to that research was created. It was running Windows 7 Enterprise, 64 bits; Intel Xeon X5570, 2.93 GHz processor; 1 GB RAM. That computer was located in the laboratory of Wrocław University of Technology and it was using diament.iit.pwr.wroc.pl as a DNS server.

Standard Windows' *nslookup* command was used to resolve DNS addresses. A batch script was created to query 2 943 733 addresses. These domain names were taken from an open directory project *dmoz* (www.dmoz.org), which is created by independent authors and consist many Web sites in different languages and about different subjects, thus providing good sample of the current Internet Web base. On the project Web site there is XML file containing links to all of sites that can be found in the open directory. DNS addresses were filtered out of that file and duplicates were removed, which left almost 3 million unique domain names. These addresses were queried from the machine 3 times, from April 20th, 2012 to May 9th, 2012. Each iteration started at 1 AM on Friday and lasted less than a week, as shown in Fig. 1.

Additionally, 2 more computers were used to gather some additional data. Private computers of Marcin Stępniak and Tomasz Salwa queried addresses in .pl domain, which created a pool of 56 059 addresses. Each iteration of DNS resolving took from 3 to 7 hours. Both machines sent DNS query 9 times. Configuration of the first private computer (notebook) was as follow: Windows 7 Professional, 32 bits, Service Pack 1; Intel Core 2 Duo P8600, 2.40 GHz processor; 3 GB RAM. Configuration of the second computer: Windows 7 Professional, 64 bits, Service Pack 1; Intel Core 2 Duo E8400, 3GHz processor; 4 GB RAM. First machine in the first test was located in Bolesławiec, Poland and used hosted-by.leaseweb.com as a DNS server. In other tests it used rtr-67.core.lanet.net.pl. Second machine sent all queries to dns.korbank.pl.

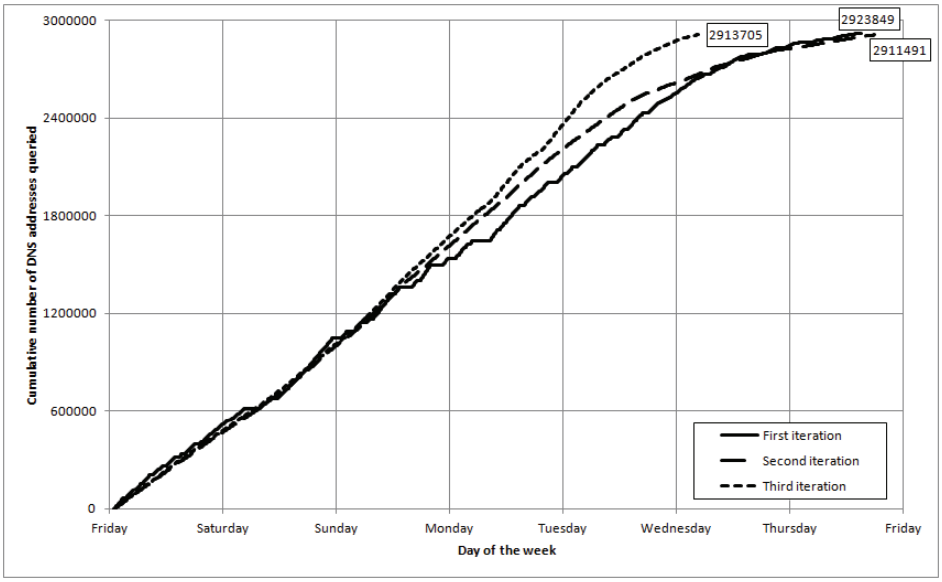


Fig. 1. Cumulative number of DNS addresses queried

2.3. RESULTS

By using 3 different computer workstation and gathering enough data from different location, time and domain names, it was possible to create statistics which can show how many Web sites use load balancing mechanism at the DNS level and in what way they utilize it. Moreover, because additional data concerning addresses in .pl domain was acquired, statistics about polish Web sites were created that are more accurate and can be compared to statistics about almost 3 million addresses that represent whole Internet.

Table 1 summarizes all results. In 3 iterations of queries concerning the pool of 2 943 733 DNS addresses, 117 159 addresses were discovered to use traffic distribution. That gives 3,98% Web sites that can be found under at least 2 different IP addresses. About 45% of them returned constant pool of IP addresses that differed only by order. 39% of these DNS addresses returned different IP address when query about them was sent to DNS server. 16% of these Web sites utilized load balancing in both of the above methods – queries about them returned a pool of IP addresses that may differ every time an answer is presented. These statistics are shown in Fig. 2.

Table 1. Web sites that use load balancing

Load balancing	Amount of DNS addresses		Percent of DNS addresses	
No data	12059		0,41%	
Load balancing undetected	2814515		95,61%	
Constant pool	52651	117159	1,79%	3,98%
Single variable address	45701		1,55%	
Variable pool	18807		0,64%	
Total	2943733		100,00%	

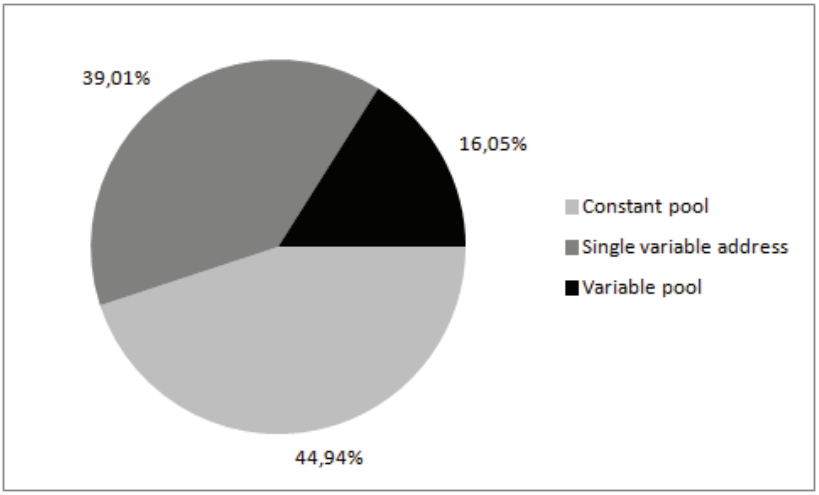


Fig. 2. Method of load balancing

From 56 059 addresses in .pl domain only 945 (1,69% of total) seemed to use load balancing mechanism at the DNS level. This is shown in Table 2. About 28% of these were discovered to return constant pool and 66% to return one address that may differ in subsequent tests. That change, compared to the previous statistics, is most likely caused by a larger number of queries iterations. Addresses in .pl domain were queried 21 times while the other only 3 times, all from one location. Rest of the DNS addresses (6%) have shown both methods of traffic distribution. Comparison of load balancing methods is presented in Fig. 3.

Table 2. Web sites in .pl domain that use load balancing

Load balancing	Amount of DNS addresses		Percent of DNS addresses	
No data	201		0,36%	
Load balancing undetected	54913		97,96%	
Constant pool	262	945	0,47%	1,69%
Single variable address	622		1,11%	
Variable pool	61		0,11%	
Total	56059		100,00%	

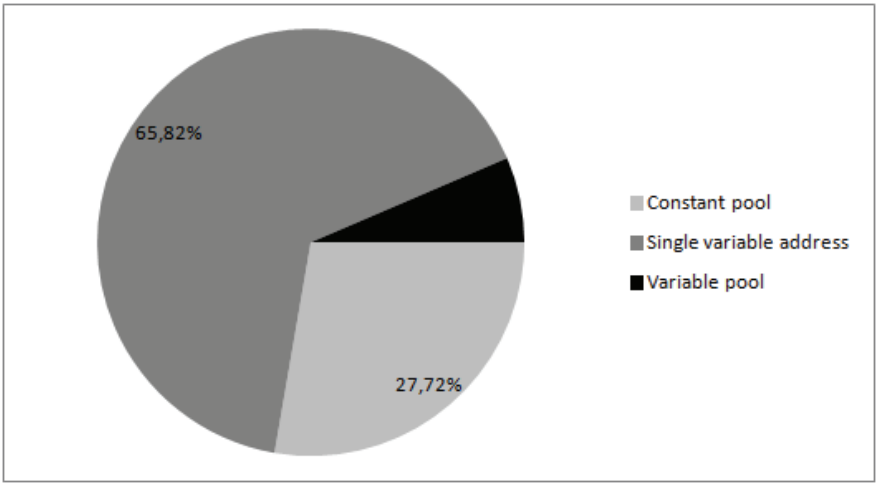


Fig. 3. Method of load balancing of Web sites in .pl domain

When knowing how many Web sites utilize load balancing mechanisms at the DNS level, another question appears: If they use traffic distribution, how many different servers do requests go to? It is possible to answer that in few ways: amount of IP addresses returned in single DNS query, amount of different returned pools of addresses in all queries, and amount of different IP addresses returned in any query. Statistics regarding that subject are shown in Fig. 4 for all 2 943 733 DNS addresses and in Fig. 5 for 56 059 addresses in .pl domain. To better clarify the results, the ordinate is scaled logarithmically and presents the number of DNS addresses that returned exactly the same number of IP addresses that the abscissa shows.

Firstly, it has to be noticed that in Fig. 4 the maximum number of different pools of IP addresses is 3. It cannot be more because only 3 iterations of DNS queries were sent from the main computer workstation which tested all 2 943 733 addresses, as opposed to 21 iterations of queries to addresses in .pl domain from all 3 machines.

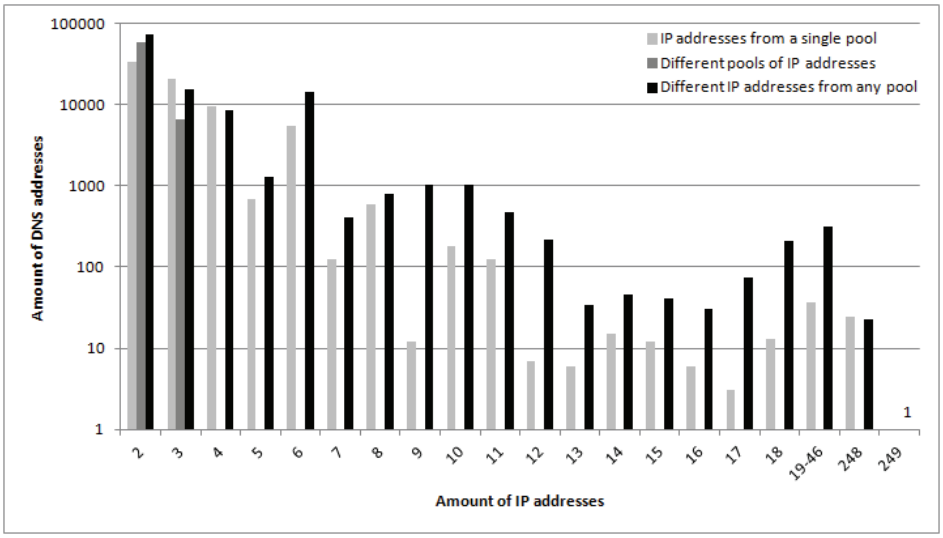


Fig. 4. Amount of IP addresses to which DNS addresses redirect

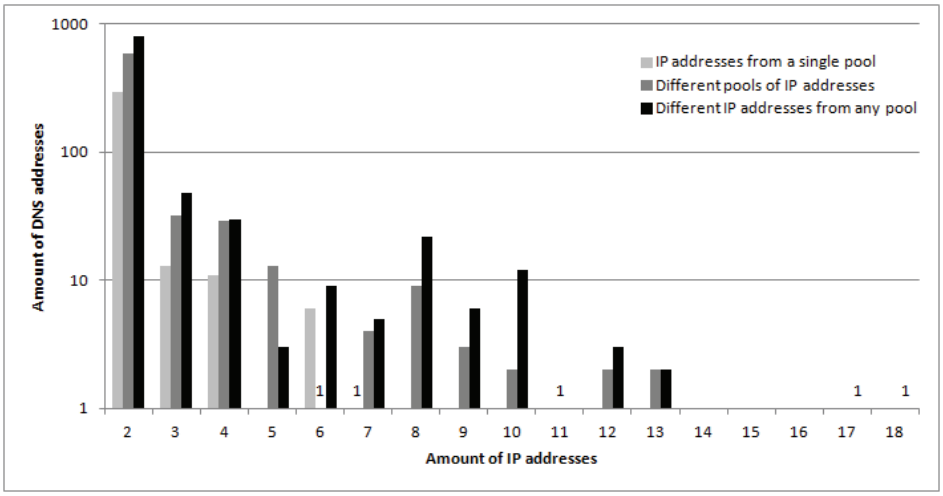


Fig. 5. Amount of IP addresses to which DNS addresses in .pl domain redirect

During tests a phenomenon was discovered related to 24 DNS addresses. Queries to these addresses, all from .netfirms.com domain, often resulted in answers consisting of a pool of 248 IP addresses, which differed only in last octet. Excluding these 24 Web sites, the maximum amount of different IP addresses associated to one DNS address is 46.

Web sites in .pl domain show load balancing using smaller number of servers. One DNS address redirected to 18 different IP addresses. Total of 13 different pools of

addresses were discovered and maximum of 7 IP addresses in one answer to a query. When comparing results shown in Fig. 5 to Fig. 4, it can be seen that DNS addresses in .pl domain usually use less different hosts in traffic distribution.

3. CONLCUSION

3.1. SUMMARY

Research shows that only about 4% of tested Web sites utilizes load balancing mechanisms at the DNS level. Though, that number can actually be higher because some addresses could have not shown signs of traffic distribution during tests. Presumably, if more iterations of DNS queries are done, additional Web sites will turn out to be utilizing load balancing too. Particularly because studies on addresses in .pl domain, which consisted of 21 iterations, shows that there can be very many different answers to a query that depends on the location and time it is sent.

It was proven that DNS addresses can redirect to multiple number of IP addresses. This is the most visible when studying the main pool of 2 943 433 Web sites. Sites, that use traffic distribution, are hosted on many hosts. It is common for a single DNS address to be associated with up to 18 addresses and sometimes even more. That is often done visibly by a single big pool of IP addresses that DNS server returns, but studies on the .pl domain shows that multiple iterations of queries from different location or at different times can result in additional addresses. These answers are controlled by DNS mechanisms, that decide where will content be delivered from, and are sometimes initially invisible.

Comparison of the results of all Web sites and Web sites in .pl domain shows that the first pool of addresses utilizes load balancing more frequently. Only 1,69% of tested DNS addresses in .pl domain use traffic distribution compared to 3,98% of almost 3 million addresses that were discovered to do this. Moreover, Web sites in .pl domain redirect to smaller number of different servers. Only one of them was associated with 18 hosts and usually rest of them were not hosted on more than 10 servers. That can lead to a statement that polish Web sites are not advanced in load balancing or simply it is not yet necessary, because of the lower amount of clients.

Several websites in .netfirms.com domain seem to redirect to 248 different IP addresses that are presented in a single pool as an answer to a query. Question arises if this is done intentionally and is suppose to be visible to every client that uses DNS Resolver. All of IP addresses come from a single network and presumably, routing should not be done at the DNS level. And even if that is the case, these addresses do not have to be presented in a single pool but instead a response could consist of only

one address. Nevertheless, during tests it was presented as a single answer and was classified in such a way.

3.2. FURTHER DEVELOPMENT

This research was done in April and May 2012 and lasted for about three weeks. That allowed to finish only three iterations of queries to all 2 943 433 DNS servers. While pool of 56 059 addresses in .pl domain could be queried multiple times, growing number of tested Web sites resulted in a much longer iteration times. More accurate results can be received by doing more iterations and by choosing more sample Web sites which both need more time. Future researches should last longer to provide more data.

Additionally, method of tests can be discussed. Batch script was most likely not the most optimal and fastest. Besides, it should be checked if using Windows' command *nslookup* is efficient. Perhaps, Linux can deal with resolving DNS addresses more quickly with *host* command and if not, at least a comparison of these two methods could be created.

Optimizing of the research also can be done when dividing address' pool into instances. On the computer workstation placed in the Wrocław University of Technology laboratory DNS addresses were divided into 6 instances of the script, from which 5 consisted of 500 000 addresses each. Even though amount of DNS address to resolve was exactly the same, some instances in all iterations finished their job quicker than others. It was discovered that addresses in some domain (for example .cn and .tw) were resolved significantly slower than addresses in other domains. This seems to show that answer from DNS server about Asian domain will be received after longer time than about European domain. That leads to a conclusion that, when dividing a pool of DNS addresses, it should not be divided randomly or by top level domain, but instead different country code top level domains should be mixed within instances.

Lastly, it could be verified what mostly influences efficiency of the test. Two factors should be considered: hardware of the computer and location (DNS server used). To check that at least two computers running different hardware configurations should run a test through the same DNS server, possibly at the same time. Then the same test, using the same pool of addresses should be run on these computer from different place, through different DNS server; thus, creating enough data to compare how long test lasted in different circumstances.

Such details refining will result not only in more accurate results but also in efficiency. Therefore, future research will either not be so time consuming, or will provide much more data in the same amount of time.

REFERENCES

[1] BORZEMSKI L., NOWAK Z., PORCZYŃSKI R., *Metody, algorytmy i rozwiązania systemowe równoważenia obciążeń serwerów WWW*, In: *Sieci Komputerowe*. VII Konferencja, Zakopane, 14–16 czerwca 2000, PŚl., Gliwice 2000, 335–360.

[2] BORZEMSKI L., NOWAK Z., PORCZYŃSKI R., *The architecture and algorithms for load balancing on web-server systems*, In: *Information Systems Architecture and Technology ISAT 2000*. Proceedings of the 22nd International Scientific School Managing Growth of Organisation Information and Technical Issues, Szklarska Poręba, 21–22 September 2000. Eds Grzech A., Wilimowska Z., Oficyna Wydawnicza. PWr., Wrocław 2000, 247–254.

[3] CARDELLINI V., CASALICCHIO E., COLAJANNI M., *The State of the Art in Locally Distributed Web-server Systems*. "Computer Science", 2001.

[4] CHEN H., ZHAO W., XIE L., *A DNS-pertinent routing algorithm with the maximum network revenue in the content distribution networks*, Proceedings of the IEEE 6th Circuits and Systems Symposium on Emerging Technologies: Frontiers of Mobile and Wireless Communication, 2004, 31 May–2 June 2004.

[5] HONG Y.S., NO J.H., KIM S.Y., *DNS-based load balancing in distributed Web-server systems*, The Fourth IEEE Workshop on Software Technologies for Future Embedded and Ubiquitous Systems, 2006 and the 2006 Second International Workshop on Collaborative Computing, Integration, and Assurance. SEUS 2006/WCCIA 2006, 27–28 April 2006.

[6] JIAO Y., WANG W., *Design and Implementation of Load Balancing of Distributed-system-based Web server*, Third International Symposium on Electronic Commerce and Security (ISECS), 29–31 July 2010.

[7] KHOSLA R., FAHMY S., HU Y.C., *Content retrieval using cloud-based DNS*, 2012 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), 25–30 March 2012.

[8] Network Working Group: *Request for Comments 1794: DNS Support for Load Balancing,* April 1995.

[9] PATHAN M., BUYYA R., *A Taxonomy of CDNs. In: Content Delivery Networks*., Springer, 2008.

[10] XU Z., HUANG R., BHUYAN L.N.: *Load balancing of DNS-based distributed Web server systems with page caching*, Tenth International Conference on Parallel and Distributed Systems, ICPADS 2004, 7–9 July 2004.

[11] ZHANG Y., MA S., HUANG J., *A Simple Approach of Improving DNS based CDN Video Sharing System*, International Conference on Information Networking, ICOIN 2008, 23–25 January 2008.

website, navigation structure,
usability, decision, network, measure,

Dmitrij ŻATUCHIN*

CHANGING THE WEBSITE NAVIGATION STRUCTURE

The navigation structure of the website interface is an essential tool for user interaction with the website. As users interact with the website, the usage statistics is collected with an online service. In existing website usability metrics, the measurement of how usable is the navigation structure was not included. To assess the usability of the website navigation structure a metric, called the energy of a network, has been developed. There were taken into account such characteristics as the availability of every page in the navigation structure, the structure of hyperlinks, and usage data of navigation structure. After valuation of the website structure with the energy of a network metric, it is possible to decide if to maintain or change the website navigation structure. The decision task on changing the website navigation structure is crucial in the task of designing the web interface.

In this paper, there are presented three scenarios after valuation of usability of the website navigation structure: optimization of navigation structure graph, leaving the existing structure and changing the way it is operated, and step changes in the graph navigation structure – leading to an increase in energy of a network. Algorithm for reducing the complexity of the complete search method for finding the optimal solution of the website navigation structure is presented. The stepwise adaptation task is formulated and the algorithm of navigation structure improvement is proposed.

1. INTRODUCTION

1.1. GENERAL DEFINITIONS

Web site is a set of connected pages which may contain: content, multimedia and embedded objects, operating on a local network or the Internet, accessible to users via Web-based User Interface (WUI), communicating to a database or other data set systems on the website's server [5]. WUI is the type of user interface and is a subclass of

* Institute of Informatics, Wrocław University of Technology, Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland.

Graphical User Interface (GUI). Web interface serves as a tool for interaction with the Internet service through a web browser. Interaction is done by inputting requests and outputting results of requests in the form of web pages generated by the website server and viewed by users.

The website users move between hypertext pages. This is achieved through hyperlinks, which are arranged in the website navigation structure [3]. The navigation structure of the website is an essential tool for user interaction within the website. If it is not possible to reach the page with the website navigation structure, which is physically located on the website, then such a page is called an orphaned node and the navigation structure is inconsistent. The way users operate the website and the efficiency of information search and processing depends upon navigation structure [12].

The usability of a website is evaluated to verify the quality of web interface, including a structure of the website, and use the results of this assessment to make changes in the structure of the website to increase efficiency, effectiveness and satisfaction of website's users [15].

There are such behaviours of users on the website, which are hard to register by observation methods [13]. In order to detect them there are used automatic data registration and data mining services [6] [8]. Analysis of website usage data make possible to assess the quality of website interface and, if necessary, adapt the website, including improvement usability of the navigation structure.

1.2. USABILITY VALUATION OF A WEBSITE NAVIGATION STRUCTURE

Evaluation and improvement of usability of navigation structure is essential for evaluating and improving usability of a website interface because the usability of the website interface depends in particular on the navigation structure [1], [4], [6], [7], [9], [11], [12], [13], [16].

There are known methods for estimating the complexity of website navigation structure, but a method for valuation its usability was proposed recently [17]. In the [17] there was proposed a network model $SSN_{t,\tau}$, which is based on usage data and construction data of website navigation structure in the defined interval of time $[t-\tau,t)$. Parameters of a network $SSN_{t,\tau}$ depend on usage data gathered in the interval of time $[t-\tau,t)$ and the structure of connections between pages of the website. To estimate the usability of website navigation structure it was proposed to use the energy of a network measure – $En(SSN_{t,\tau})$ – which makes possible to evaluate a website's navigation conformity to the way of how real users do use the website after its release to the general public. To calculate the value of the energy of a network, the characteristics of $SSN_{t,\tau}$ – impression of nodes and impression of edges – are defined [20].

Estimation of usability of website navigation structure (calculation of $En(SSN_{t,\tau})$) for usage data from the interval of time $[t-\tau,t)$ allows to decide whether to maintain or change the navigation structure of the website.

2. DECIDING TO CHANGE A NAVIGATION STRUCTURE

2.1. SCENARIOS AFTER USABILITY VALUATION OF WEBSITE NAVIGATION STRUCTURE

After certain period of usage of a website, its usability is assessed. It is usual that usability problems, especially within navigation structure, are detected. It is crucial to revise the construction of navigation structure if user goals are not reached. Then, it should be decided whether to change the website navigation structure or to leave it.

As the measure of usability of website ($En(SSN_{t,\tau})$) is calculated the following scenarios of dealing with the website navigation structure are proposed:

– leave the existing navigational structure and change the way that users use the website,
– optimization of graph of navigation structure,
– step changes of website navigation structure leading to an increase in the energy of a network and, in final, the adaptation of website navigation structure to the usage habits of website's users.

In order to change the way that users operate the website structure, the work with users should be done i.e. with the help with marketing (advertisement campaigns), education (online and offline courses) or online help (live chat solutions, recommendation methods). Then the website owner may expect some of the users to change their navigation habits. The second and third scenario will be discussed in detail.

For the second and third scenario, if the calculated value of the energy of a network ($En(SSN_{t,\tau})$) is less than the energy of a network of optimal website navigation structure ($En_{opt}(SSN_{t,\tau})$) or an acceptable value of the energy of a network ($\overline{En}$), the structure of the analysed website should be changed. Otherwise, the navigation structure remains unchanged.

An acceptable the energy of a network value ($\overline{En}$) can be determined arbitrarily (e.g. by an experienced usability specialist), or calculated using the set of measured the energy of a network values of similarly constructed usable website with a same context of use, user population and the way of usage of navigation structure. Another way to determine the value of acceptable energy is to use the statistical quality control i.e. Shewhart control cards [3] [17].

The $\overline{En}$ value, calculated on basis of set of the energy of a network values of usable website, is defined as:

$$\overline{En} = \frac{1}{k}\sum_{i=1}^{k} En(SSN_{t,\tau}(GS_i)), \tag{1}$$

where $En(SSN_{t,\tau}(GS_i))$ is the i-th value of the energy of a network of navigation structure from the set of k websites, which were assessed by usability analysts or users.

2.2. OPTIMIZATION OF WEBSITE NAVIGATION STRUCTURE

Optimization of website navigation structure consists of a series of changes made to a graph structure such, as the optimal navigation structure (maximum $En(SSN_{t,\tau})$ value) will be reached for the given way of usage of the website and given navigation structure.

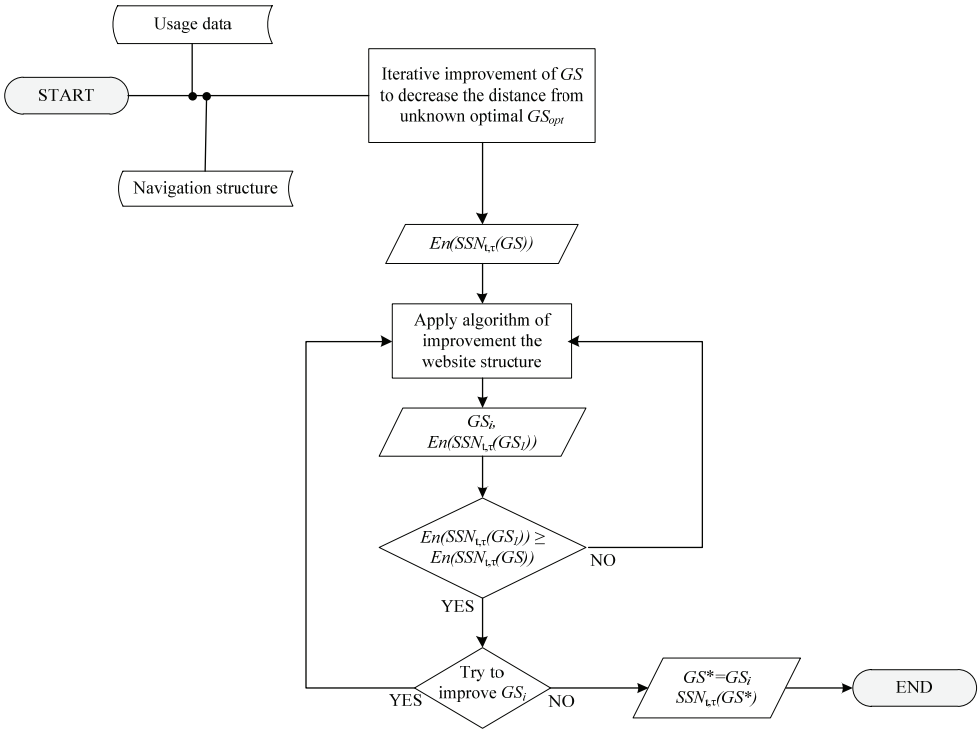


Fig. 1. Improving the usability of the website navigation structure through optimization

After estimating the energy of a network of valuated navigation structure, it should be decided whether a navigation structure needs to be changed or not. It requires knowledge about the optimal navigation structure for the given population of users, usage data, structure, navigation and value of optimal the energy of a network.

Finding the optimal website navigation structure is possible using a complete search, which involves checking all possible combinations of edge connections for a given set of nodes of a graph navigation structure *GS*.

This task is difficult and time-consuming due to the computational complexity of a complete review, which for a directed graph is O (n!). Due to the large computational complexity of a complete search method, for quality valuation of website navigation structure, an acceptable level of energy ($\overline{En}$) or stepwise approach for improvement of the navigation structure may be applied, what should be controlled by the energy of a network, taking into account its general properties. On Fig. 1 the process of optimization is shown.

In order to reduce complexity of complete search method, the Connecting Components of Navigation Structure (CCNS) algorithm is proposed.

The CCNS algorithm it is proposed to reduce analysis of all possible connections to analysis of strongly connected components of the graph structure GS and control their quality with the use of the energy of a network measure. The graph of strongly connected components with the highest value of the energy of a network, and if this is a value greater than the energy of the original network, is a locally optimal solution and proposed as a solution to the problem of adaptation of the website navigation structure.

The algorithm is as follows:

Step 1. Directed graph *GS* contains cycles and though is converted into the acyclic graph, using Gasner's transformation algorithm.

Step 2. The acyclic graph of navigation structure is searched for strongly connected components using Tarjan's algorithm.

Step 3. As the result of the Strongly Connected Components algorithm [16], the graph of strongly connected components (GS_{SCC}) is constructed.

Step 4. On the GS_{SCC} graph there are constructed *i*-th graphs of strongly connected components $GS_{SCC,i}$, which are the combinations of connections between nodes of the GS_{SCC} graph.

Step 5. For each *i*-th $GS_{SCC,i}$ graph the energy of a network of $SSN_{t,\tau}(GS_{SCC,i})$, coherent to this graph, is calculated..

Step 6. Within all possible graphs of strongly connected components the graph with highest value of the energy of a network $En(SSN_{t,\tau}(GS_{SCC,i}))$ is selected for further analysis. If the energy of a network satisfies the condition $En(SSN_{t,\tau}(GS_{SCC,i})) > En(SSN_{t,\tau}(GS))$, then such a graph is proposed as an output of the CCNS algorithm.

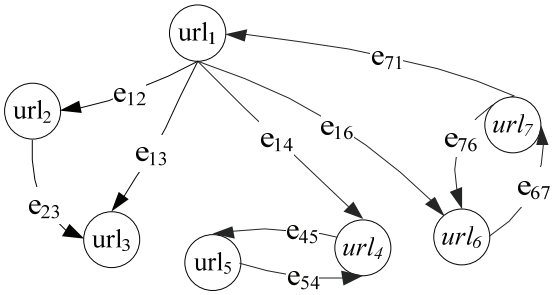


Fig. 2. Example graph of website navigation structure (7 nodes, 10 edges)

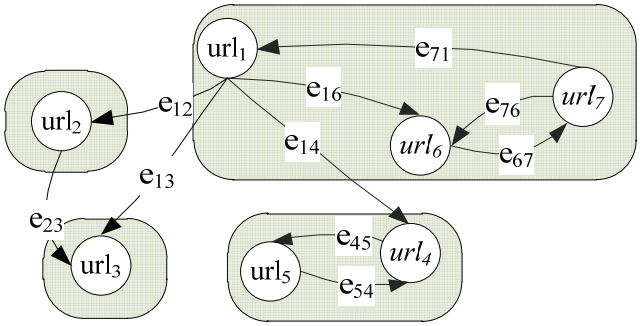


Fig. 3. Example graph of strongly connected components built from example graph (Fig. 2)

Analysis of a sample graph of the website navigation structure *GS* (Fig. 2) requires the analysis of a 5040 possible combinations of nodes in the graph structure with the complete search method, which is 210 times more compared to the number of all possible combinations of strongly connected components in a GS_{SCC} graph built over the *GS* graph (Fig. 3).

The CCNS algorithm results in:

– the reduction of the computational complexity to polynomial of the fourth degree, compared with the complexity of complete search (O(n!)),
– the loss of the guarantee of finding the optimum solution.

2.3. STEPWISE APTATION OF THE WEBSITE NAVIGATION STRUCTURE

The adaptation of the website navigation structure is to make such changes in the graph *GS*, which will result in the graph GS^*, for which the energy of a network will increase compared to the initial value.

Task of adaptation of the website navigation structure with known acceptable value $\overline{En}$ is following

For:

- the graph of the website navigation structure *GS*,
- usage data in the time interval $[t-\tau, t)$,
- the energy of a network $En(SSN_{t,\tau}(GS))$,
- given acceptable value of the energy of a network $\overline{En}$,
- D_{GS} – space of possible graph structures,

Find:

the website navigation structure $GS^* \in D_{GS}$, for which the estimated value of the the energy of a network will satisfy the following condition:

$$En(SSN_{t,\tau}(GS^*)) - \overline{En} \geq 0 \tag{2}$$

In the task of adaptation of the website navigation structure, without the knowledge of "good" navigation structures of reference website, and thus not know acceptable value $\overline{En}$, in order to reach a possible improvement of the website navigation structure the general properties of the usability measure of navigation structure (the energy of a network) can be used.[18]. For this purpose the algorithms of stepwise adaptation may be applied (e.g. algorithm of promotion [19]). In order to increase the value of the energy of a network, thus usability of the website navigation structure, another algorithm is proposed.

The Improvement of Navigation Structure (INS) algorithm results in reduction of number of transit nodes, new paths in a website structure and connections between orphaned nodes with the nodes from connected part of website structure. These results in an increase of the energy of a network value.

The INS algorithm consists of following procedures:

- connection of orphaned nodes,
- detection of transit nodes,
- merge of transit nodes,
- bypass of transit nodes.

In procedure of connection of orphaned nodes such a property of the model of navigation structure (network $SSN_{t,\tau}$) is used – the addition of edge between orphaned node and not orphaned nodes in the graph results in the increase of the energy of a network.

In the procedure of detection of transit nodes a set of transit nodes, which occur in the navigation structure, are processed and the following conditions are checked:

- if the *i*-th transit node is a father of a leaf-node, then it is merged in the merge procedure.
- if the *i*-th transit is on a lower depth in the navigation structure (and belongs to the transit path), then it is omitted in the procedure of bypass of transit nodes.

In the procedure of merge of transit nodes and leaves-nodes, the property of the $SSN_{t,\tau}$ network model is used. If the depth of the node, which is a leaf-node in the website navigation structure, is reduced, then the energy of a network increases. Thus, as the result of merge of the transit nodes with the leaf-node, the path length to the merged node decreases, the depth of the merged nodes is one less than the depth of the merged leaf-node, what results in the increase of the energy of a network.

In the procedure of bypass of transit nodes, the following property of the $SSN_{t,\tau}$ network model is used – if the path to the node in website navigation structure decreases, the energy of a network increases (e.g. Fig. 4).

The computational complexity of the INS algorithm consists of described procedures and equals $O(N^2)$.

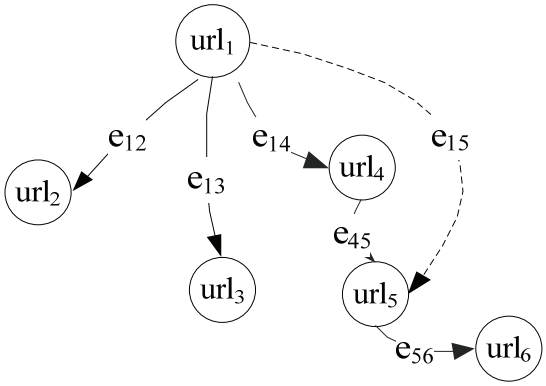


Fig. 4. Example improved graph with omitted transit node (url4) with the new edge (dashed)

The adaptation process of website navigation structure is an iterative process that ends when the condition (2) (for known acceptable value $\overline{En}$) is satisfied or when the growth with subsequent iterations will be less than the cost of improving the navigation structure (for an unknown value $\overline{En}$).

3. SUMMARY

The usage data gathered for a period of time makes possible to evaluate the usability of website navigation structure. In this paper, there were proposed three scenarios after usability valuation. The navigation structure of the analysed website may be left the same if the behaviour of users will be changed. Otherwise, the website navigation structure may be changed if the value of the energy network of navigation structure

$En(SSN_{t,\tau})$ is less than the optimal value of the energy of a network ($En_{opt}(SSN_{t,\tau})$) or an acceptable value of the energy of a network ($\overline{En}$).

It is possible to find the optimal website navigation structure using a complete search method. Due to the large computational complexity of a complete search, it was proposed to use an acceptable value of the energy of a network ($\overline{En}$) and the algorithm of connecting components of navigation structure.

To adapt the website navigation structure, the stepwise improvement using general properties of the energy of a network measure was proposed. Such an improvement may be done with the iterative application of adaptation or improvement algorithms.

REFERENCES

[1] ARNEY J.B., LAZARONY P.J., *An Inclusive Guide To Assessing Web Site Effectiveness*, In: Journal of College Teaching & Learning, Vol. 2, No. 1, 2005, 27–36.

[2] BARESI L., GARZOTTO F., PAOLINI P., *Extending UML for Modeling Web Applications*, 34th Annual Hawaii International Conference on System Sciences, 2001, 3055.

[3] BASSEVILLE M., NIKIFOROV I.V., *Detection of abrupt changes: Theory and Application*, Englewood Cliffs, N.J: Prentice-Hall, 1993, 26–32.

[4] BENBUNAN-FICH R., *Using protocol analysis to evaluate the usability of a commercial web site*, Journal Information and Management Vol. 39 Issue 2, 2001.

[5] ENCYCLOPÆDIA BRITANNICA INC. (2012, July), *Web site*, In: Encyclopædia Britannica, [Online] http://www.britannica.com/EBchecked/topic/690679/Web-site

[6] GEHRKE D., TURBAN E., *Determinants of Successful Website Design: Relative Importance and Recommendations for Effectiveness*, Thirty-second Annual Hawaii International Conference on System Sciences, Volume 5, Maui, Hawaii, 1999, 5042–5050.

[7] HASAN L., ABUELRUB E., *Assessing the Quality of Web Sites*, INFOCOMP Journal of Computer Science Vol. 7, No. 4, 2008.

[8] IVORY M.Y., *An Empirical Foundation for Automated Web Interface Evaluation*, PhD Dissertation 2001.

[9] KIM J., LEE J., HAN K., LEE M., *Businesses as Buildings: Metrics for the Architectural Quality of Internet Businesses*, Information Systems Research Vol. 13 No.3, 2002, 239–254.

[10] LEVI M.D., CONRAD F.G., Bureau of Labor Statistics, July 2008, [Online] http://www.bls.gov/ore/htm_papers/st960150.htm.

[11] LYNCH P.J., HORTON S., *Web style guide: basic design principles for creating Web sites*, NJ: Yale University Press, 2009, [Online] http://info.med.yale.edu/caim/manual.

[12] NIELSEN J., *Designing Web usability: The practice of simplicity*, Indianapolis: New Riders Publishing, 1999.

[13] PALMER J.W., *Web site usability, design, and performance metrics*, Information Systems Research Vol. 13, No.2, June 2002, 151–168.

[14] ROSENFELD L., *Defining Information Architecture*, Designing Large-Scale Web Sites, 2nd ed.: O'Reilly Media, 2002, ch. 1.

[15] SAVIOJA P., NORROS L., *Systems Usability – Promoting Core-Task Oriented Work Practices*, In: Maturing Usability, Springer-Verlan London Limited, 2008, 123–143.

[16] ZHANG P., VON DRAN G. M., *User Expectations and Rankings of Quality Factors*, International Journal of Electronic Commerce, Vol. 6, No. 2, 2002, 9–33.
[17] ŻATUCHIN D., *Problem of website structure discovery and quality valuation*, Computer Science and Information Systems (FedCSIS), Szczecin, 2011, 117–122.
[18] ŻATUCHIN D., *Webgraph: system do analizy i syntezy struktur serwisów www*, In: Interfejs użytkownika: Kansei w praktyce, Warszawa: Wydawnictwo PJWSTK, 2011, 72–87.
[19] ŻATUCHIN D., M*etoda przebudowy interfejsu serwisu internetowego oparta na historii użytkowania*, In: Interfejs użytkownika: Kansei w praktyce, Warszawa: Wydawnictwo PJWSTK, 2010, 98–105.
[20] ŻATUCHIN D., GRZECH A., *Evaluation of website interface quality*, In: Advances in systems science, Warsaw: Exit, 2010, 271–280.

web service, ontology, semantic annotation, Wiki-WS

Andrzej SOBECKI*, Marek DOWNAR*

WEB COMPONENT FOR AUTOMATIC EXTRACTION OF ONTOLOGICAL INFORMATION FROM INFORMAL DESCRIPTION OF WEB SERVICES

This article treats about usage possibilities of Web Services and cooperation in development that leads to constant improvement of these components. It describes the semantic methods which can be used to create the description that is comprehensible for computers. It also presents the two models supporting the automatic generation of the semantic description based on informal description. The paper draws upon the comparison of two languages, which can be used while defining the semantic description of the Web Services. This article presents the way of creating, developing and using the ontology in the Web Services repositories.

1. INTRODUCTION

The variety of methods solving the same problem by many services requires distinguishing them from each other. The great sets of such elements can be searched efficiently only by the systems established for that purpose. Relying on the informal description requires using dictionary and lexical methods and does not guarantee the full understanding of how the Web Services work. The description of the services should be written in the format that is comprehensible for the computer, so it should be derived from the natural language elements, which have to be defined previously. The solution proposed in the paper is based on the semantic description supported by specific domain ontology defined in the system. The main drawback of this solution is the fact that it requires the knowledge of the language used to define the semantic annotations as well as the additional time is needed to create such a description. This article describes the model supporting the semantic description generation while establishing informal description.

* Department of Computer Architecture, Gdansk University of Technology, Narutowicza 11/12, 80-233 Gdańsk.

2. WEB SERVICE USAGE

The evolution of the Internet enables the free communication and the information exchange between the remote centers. It also influences the software development and distribution. The standards which were acceptable a few years ago are continuously evolving and changing the way of cooperation. The great example of the evolution process is the development of the SOA (*Service Oriented Architecture*) that recommends creating applications that are oriented on services. Implementing the Web Services in the distribution systems, such as Wiki-WS platform [1], enables creating, searching and free usage of them. Established applications can use the Web Services to solve the problems instead of the traditional libraries. Software developed with the use of the SOA [2] design patterns have a positive impact on the independence of the components from the application and vice-versa. Consequently, the business logic remains outside the main program. Redeploying the application on other platforms, even slower and with less resources, is much easier than in the traditional solutions. The diagram of using the Web Services repository by the applications in accordance with SOA is presented in figure 1.

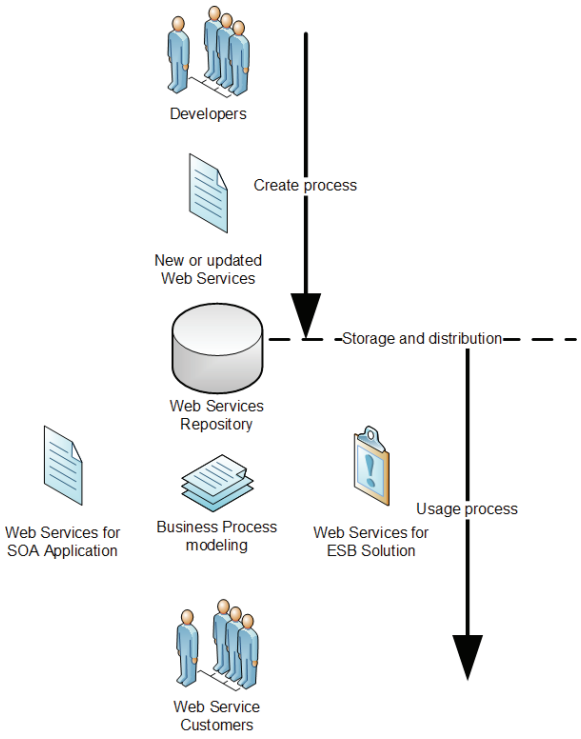


Fig. 1. The use of the Web Services repository in the SOA applications

The main problem of developing applications in SOA paradigm is finding the right service. The lack of unified methods makes it impossible to distinguish and choose the service depending on its context. The traditional methods of describing the Web Services (e.g. WSDL) do not provide to the public any information such as functionality of the service and the way of using it. This kind of information can be attached as a supplement in the Web Services descriptor, however, it remains out of the control.

The solution of the Web Services identification problem in the semi-automatic or automatic way could be the usage of the ontology languages. The development and availability of many standards (e.g. OWL, OWL-S, OWL-WS or WSDL-S) makes it difficult to use them in the majority of solutions. These standards are characterized by the distinctive formal language and the scope of functionality. However, the enforcement of the Web Services description in such format is harder. Additional expectations about the description is also the conceptual cohesion with a specific dictionary, and such knowledge of this dictionary cannot be expected from the customer who proposes the services in the distribution system.

3. DOMAIN ONTOLOGY

The artifact registry, which is too expanded, may cause inaccuracies. The efficiency of generating the semantic description depends on the use of the domain ontology, which should be pulled out from the universe (description of the world). The set of definitions can be written in the database [3].

The cooperation and distribution services system [4], for example Wiki-WS, expanded by the ontological information set can be used by the agents as well as by the human beings. The proper label, comprehensible for computer, which can bind together information about the interface and about the way of implementing the services in the context of executing, enables the distinguishing the identical services in a way of functionality. Currently the formats specifying the services (e.g. WSDL) do not contain the described relations. Adding the semantic description requires the use of distinct formats and the domain ontology sets.

Domain ontology connected to Web Services with the semantic description enables the evolution of the knowledge exchange systems and Web Services such as Wiki-WS to Wiki-SWS [5]. The resources that are available in such a system might be distinguishable and chosen depending on the context of use. Additionally, there might be the possibility of the automatic choice of services to the defined usage scenario e.g. in languages OWL-S, SWSL or OWL-WS. The diagram of the Wiki-SWS system structure supported by the ontological information is presented in figure 2.

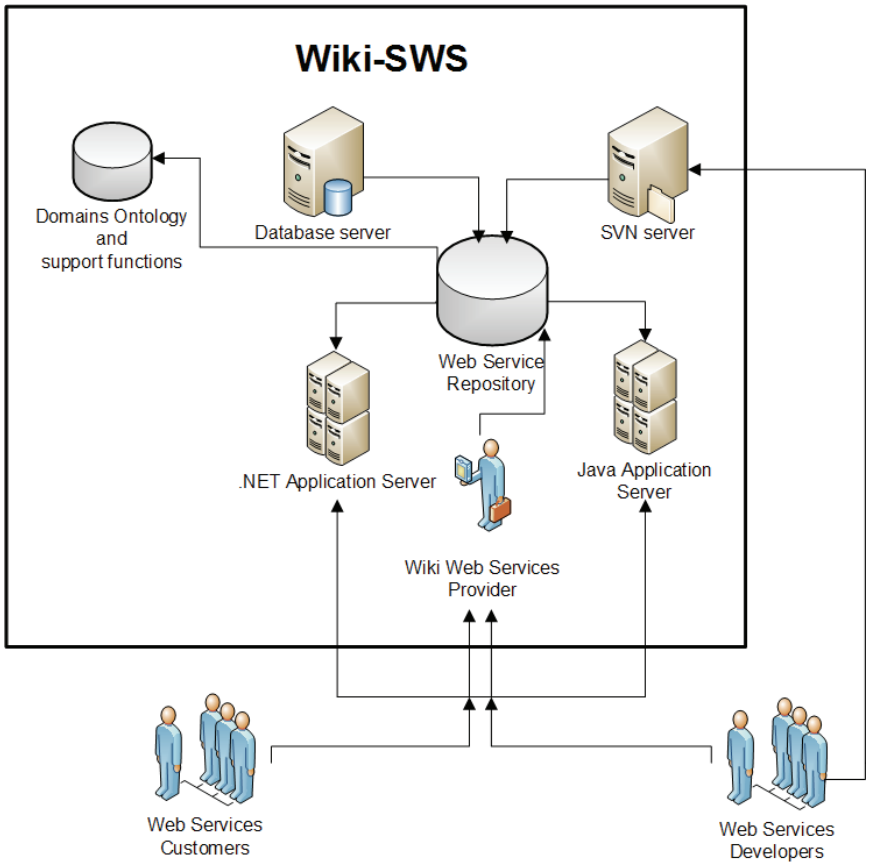


Fig. 2. Wiki-SWS system structure

4. THE PRINCIPLES OF THE PROPOSED MODEL

The main element that distinguishes storing web-services in the Wiki-SWS system from the Wiki-WS system is adding to it the semantic information. The process of adding the services is mainly connected with filling the fields of the form such as:

- Name,
- Description,
- Keywords,
- Source code.

The last one can be added by uploading the group of files. Defining the semantic description shall occur in the place of adding new services and be supported by the web portal.

It should be assumed that the supported language is English and the domain ontology is defined in the system or in the entire ontology there can be distinguished subsets, which are the domain ontology, and the mechanisms of lemmatization using lexicon e.g. WordNet are available [6]. The kind of support will depend on the maturity of the domain ontology available in the system. There can be two supportive solutions:

- The text processing and exchanging it into semantic description with supervision,
- Defining ontological triples *(O,A,R)* on the fly, while writing.

4.1. THE METHOD OF INFORMAL TEXT PROCESSING WITH SUPERVISING

The first solution is possible to use in case when the system does not contain the expanded ontology and cannot fully automate the process of the semantic description. In this method we define the term of the context where the service is used. And based on the gained information the extraction of the triple (O,A,R) is possible. The triple is the value set that is defined as follows: attribute A is in relation with R with object O. In our case the relation will be binary function indicating value 0 or 1.

To determine the context properly, the adequate length of the service description is required. Then it is processed with the use of NLP (Natural Language Processing), in which the consecutive stages, in accordance with [7], are:

- Division into sentences and words,
- Labeling of the parts of the speech,
- Bringing to the basic form (lemmatization).

The words or definitions distinguished by M. Hearst method [10] are then reviewed using the measures [8][9] in accordance with the equations 1 and 2, in which s is a service and c is a concept. The first equation determines the probability of occurring the service s on condition that it is connected with the concept c . The second equation determines the quality of selected terms from the text in relation with all known terms. The vector space of the services description is presented in table 1.

$$Cond(s,c) = P(s|c) = \frac{f(s,c)}{f(c)} \tag{1}$$

$$PMI(s,c) = log_2 \frac{P(s|c)}{P(s)} \tag{2}$$

Table 1. Vector space of the services description

	concept 1	concept 2
Web service 1	1	1
Web service 2	1	1

The information gained about assigning the terms to the services are reviewed with the domain ontology we have. Based both on the relation of the terms found thanks to the text processing method and the relations of those terms in the domains ontology it is possible to propose to the user the additional relations. Thanks to the supervision this method enables to reject or accept the proposition or expand the ontology with the terms unknown so far.

We defined the concept of the formal context as a connection of the objects and the attributes with the incidence relation R – (O, A, R). Additionally, implementing the term of the formal concept (O_i, A_i), in which O_i is the object set, which attributes are included in A_i. And A_i is the set of all attributes connected with R relation with every object O_i. We can establish the taxonomy with the use of the R. Wille method [11] described in [7] based on the acquired set. The taxonomy we have gained, expanded with domain ontological concepts, can create the context of the web services. Finding the concepts as well as the attributes connected with them is possible thanks to the use of the text processing method [13], which aim is to determine the key words or summarizing the text. In many publications [12], [13] it is claimed that most frequently the terms in the text are noun phrases. Using the regular phrase, determined by the formula (3), it is possible to distinguish the following phrase from the text [12]:

$$((Adj|Noun)^{+}|((Adj|Noun)^{*}(NounPrep)^{?})(Adj|Noun)^{*})Noun \tag{3}$$

As reads:

- *Adj* – adjective,
- *Noun* – noun,
- *NounPrep* – noun preposition.

Subsequently using the statistical methods enables the determination of "the quality" of the found words. That shows the significance of their role in the document. Taking this into account, the terms with slight informative value can be rejected [14][15].

4.2. THE METHOD OF CREATING FORMAL DESCRIPTION ON THE FLY

The second method of supporting the creation of the semantic description is suggesting terms, attributes and relations while establishing the description by the user. This attitude requires the ontology integrated with the system to be complete and expanded. In the construction of the Web Services description there can be distinguished three elements:

- The description of the problem it solves,
- Condition (context) in which it works,
- Functions and their features which it provides.

Each element should be supported by the distinct function module so that the support is full. Additionally, it is recommended that the stage describing functions and their features should use the description included in the service descriptor and should enable the connection between implementing information and existing endings.

This kind of support does not require the concept selection method, because the user defining the description chooses the kind of element according to the given word. Complementing the informal description with the relations with domains ontology artifacts isolates the user from the language, in which the semantic description is defined.

The expanding of the proposed model is the extension of the offer list with the concepts consistent not only syntactically but also semantically on the basis of the defined relation. It makes possible to hold the dialogue with the user while creating the description, when question asked by the machine is "If you have defined the attribute A in relation to O_1, in relation R_1, the truth is that O_2 is in relation R_2 with the attribute A?". The user can accept the hint or reject it.

Both the acceptance and the rejection can influence the level of cohesion of the relation proposed in the model. In case of the lack of the proper relation or the concept the system should take the notification about that. If the proposition is consistent with the expectation of the user the relation should be rewarded, if not, it should be penalized. It enables the automatic adjustment of the existing ontologies to the demands of the final users. Such a model of the ontology influenced by the user can support the development of the ontology, similar to the HS model (Helix-Spindle model) described in the paper [16].

5. THE COMPARISION OF THE SEMANTIC DESCRIPTION LANGUAGES

Establishing the ontological system supporting the distribution and managing of the services requires examining the scenarios which can be realized later. Currently the support for generating semantic description only for the Web Services is sufficient from the position of the system, however, in the long time perspective it will be required that the semantic description will be also available for the scenarios of services, that is the complex services.

That is why the choice of language used in the formal description definition is not a trivial undertaking. The variety of solutions and diversification of the offered possibilities requires listing them and choosing the one that meets most of the expectations. Based on article [17] in table 2 there were listed the languages and research projects connected with the semantic annotation and services composition. In the column "semantics" following labels were taken:

- + semantic information directly included,
- – semantic information not included,
- +/– simple reference to existing ontologies.

In the column annotations/composition:

- A available only semantic annotations,
- C available only the services composition,
- AC both available: semantic annotations and services composition,
- – it cannot define semantic annotations and services composition.

Based on the above list it may be assumed that the promising language in the systems creating the semantic annotation will be OWL-WS.

Table 2. The list of the research projects on the semantic annotations and composition web service

Project/Organization	Language	semantics	annotations/ composition
Uni Trento	"semantic BPEL"	+/-	A
OWL-S Coalition	OWL-S	+	A
W3C WG	SAWSDL	+/-	A
WSMO	WSML	+/-	A
DAML.org/SWSI	SWSL	+	A
SHOP2	OWL-S	+	C
NextGrid	OWL-WS	+	AC

6. THE CONCEPT OF USING PROPOSED MODEL

The system managing Web Services is most frequently put on the server accessible from the Internet. The tool that enables the communication with the system is in this case the browser. Portal supporting the users should help them in generating the semantic annotations with the use of the component implemented in the Internet application. The Wiki-WS platform supporting the realization the Web Services built entirely at the Gdansk University of Technology has been realized with the use of ASP.NET MVC 3 technology. This system is the basis to implement the described methods of creating the semantic description.

The component should enable the integration with the existing platform managing the repository. The current condition of the domain ontology does not allow to use the second of the described methods of supporting the formal description. Therefore the first model has been chosen. The component carries out most of the operations synchronously. Only the stage of the interaction with the user during the presentation of the semantic description propositions occurs asynchronously. It is planned to connect the component with the Microsoft Business Intelligence system to bring results on the basis of the data we receive.

Expanding the services distribution system with the domain ontology must combine the searching of the web services with the lemmatization module of the phrases written in by the user. Then such processed text could be passed on directly to the reasoning module. We must consider the additional information set, which is required to reduce the number of results.

7. SUMMARY

The proposed models of expanding the description of the services with the semantic annotations may make easier the searching of the appropriate services both by the user and by the machine in the intelligent space. Putting the services in the context of their operation allows to distinguish the services which work in the specifically defined conditions.

The system which has the semantic description of all services in the repositories should contain the modified broker of the services based on the register UDDI [18]. The results of its work based on the keywords should be expanded with the semantic information. The import of the ontological information to the UDDI registry would enable the distinguishing of the services against each other and the understanding of how the services work by the system searching the services itself [19].

In contrast to the classical method of creating the libraries, the web services performing the same functions can be a autonomous application. The increasing popularity of the SOA solutions will enable the gathering of numerous web services. If the coherent system of their labeling is not established timely, then accurate reasoning in the future may be hindered. In such a situation, soon the searching methods may lead to wrong results or the number of retrieved services will make it impossible for us to achieve the expected solutions. Therefore it is very important to implement the tools supporting the automatic generation of the description in the formal language, without expecting from the user the knowledge of this language. But at the same time providing the possibilities of searching and matching the Web Services automatically.

REFERENCES

[1] KRAWCZYK H., DOWNAR M. *Commonly Accessible Web Service Platform – Wiki-WS*, *Intelligent Tools For Building a Scientific Information Platform*, Studies in Computational Intelligence, 2012, Vol. 390/2012, Springer Berlin/Heidelberg, 251-265.
[2] ERL T., *SOA Design Patterns,* Prentice Hall, 2010.
[3] SZYMAŃSKI J., Portal do kooperacyjnej pracy nad ontologiami dziedzinowymi, KASKBOOK, 2007.
[4] SOBECKI A., DOWNAR M., *Wiki-WS – Repozytorium kodów źródłowych i środowisko wykonawcze usług sieciowych*, Materiały konferencyjne ICT Young 2012, II Krajowa Konferencja Studentów i Doktorantów Elektroniki, Telekomunikacji, Informatyki, Automatyki i Robotyki, Gdańsk 2012, 607-613.
[5] KRAWCZYK H., *Semantyczny model realizacji ludzkich przedsięwzięć*, KASKBOOK, 2007.
[6] PENG-YUAN L., TIE-JUN Z., XIAO-FENG Y., *Application-Oriented Comparison and Evaluation of Six Semantic Similarity Measures Based on Wordnet*, In: Machine Learning and Cybernetics., 2006.
[7] MICHALSKI M., *Automatyczna budowa taksonomii usług w oparciu o ich opisy w języku naturalnym przy użyciu zewnętrznych źródeł wiedzy*, KASKBOOK, 2009.
[8] HINDLE D., *Noun classification from predicate-argument structures*, In: Proc. Annual Meeting of the Association for Computational Linguistics (ACL), 1990.
[9] CIMIANO P., *Ontology Learning and Population from Text: Algorithm, Evaluation and Applications*, Springer, 2006.
[10] HEARST M., *Automatic acquisition of hyponyms from large text corpora*, In: Proc the 14th International Conference of Computational Linguistics (COLING), 1992.
[11] WILLE R., *Restructuring lattice theory: an approach based on hierarchies of concepts*, Ordered Sets, 1982.
[12] JUSTESON J., Katz S., *Technical terminology: some linguistic properties and an algorithm for identification in text*, Natural Language Engineering, 1995.
[13] KOZŁOWSKI M., *Inteligentne metody wykrywania istotnych pojęć w korpusach tekstowych*, In: Zeszyty Naukowe Elektroniki, Telekomunikacji i Informatyki Politechniki Gdańskiej, Tom 1, 2011.
[14] JONES K., *A statistical interpretation of term specificity and its application in retrieval*, In: Journal of Documentation., 1972.
[15] ANDRADE M., VALENCIA A., *Automatic extraction of keyword from scientific text: application to the knowledge domain of protein families*, BioInformatics, 1998.
[16] KISHORE R., ZHANG H., RAMESH R., *A Helix – Spindle Model for Ontological Engineering*, In: Communication of ACM, Vol. 47, No. 2, 2004.
[17] DZIUBICH K., *Semantyczny Workflow jako usługa złożona*, KASKBOOK., 2007.
[18] UDDI: The UDDI Technical White Paper, http://www.uddi.org/, 2011.
[19] SRINIVASAN N., PAOLUCCI M., SYCARA K., *Adding OWL-S to UDDI, implementation and throughput*, Robotics Institute, Carnegie Mellon University, USA, Springer, 2002.

usability testing, human-computer interaction

Piotr CHYNAŁ*

A METHOD FOR COMPARING EFFICIENCY OF THE DIFFERENT USABILITY EVALUATION TECHNIQUES

In this paper I present a method for comparing efficiency of different usability techniques. While performing a thorough usability audit of a particular website we use different usability techniques such as expert evaluation, focus groups, clicktracking, eyetracking and many others [3], [7]. For the certain types of web systems different techniques might be more or less effective. To compare the different usability methods I have created a formal representation of a method's properties. After performing a usability evaluation we can assign the obtained data, such as number of usability problems found on the website, the importance of those problems, cost and time, to the method properties model. After that we can compare those models and see which of the used techniques are more effective for the particular web system.

1. INTRODUCTION

Usability testing is one of the methods of software testing. Such tests are becoming more and more popular in recent years [1], [2], [5], and we can evaluate usability of web systems and desktop and mobile applications. According to the norm ISO 9241 we can define usability as "The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use" [4]. Usability testing allows us to increase lernability, satisfaction and efficiency for the given system and also thanks to such test we can find and eliminate errors.

While performing an usability audit of a particular web system we use many usability testing techniques. We often perform usability audit more than once for a system, for example after introducing some new functionality or after some bigger layout changes. To save time and money on a usability audit we would like to use only the

* Institute of Informatics, Technical University of Wrocław, Wybrzeże Wyspiańskiego 27, 50-370 Wrocław.

techniques that were the most effective for our system. To achieve that we need to compare the results, obtained from using those techniques for the first time, in our system evaluation. To enable such comparison we need a formal verification method. Such method is introduced in this paper. It was used to evaluate which of the usability techniques were more efficient for a SOA-based web application platel.pl (Fig. 1).

Fig. 1. Platel.pl website

We have performed expert evaluation and a focus group evaluation of that system. Than both methods have been compared using the introduced method.

2. EVALUATION METHOD

To compare different usability testing techniques we need to establish that properties of such technique can be written as:

$$W =< D, K, c, t >$$

Where D is the accuracy of the technique (whether only big usability problems have been found, or has the evaluation using this technique showed also smaller usability problems), K is the completeness of the technique (number of usability problems found), c is the cost of such method and t is the time that we needed to perform a test with the given technique. We say that a usability problem has occurred when the usability rules provided in the definition are broken in the evaluated system, for example where the system or its part is not effective, or there are some errors in it.

To all of those parameters we will assign a value from 0 to 1. The 1 value will be assigned to the parameter of the technique that had the better results in the usability test, for example if using a technique we would found more usability problems than with using the other one, the *K* parameter of the first one will be 1, and the *K* parameter of the second one will be calculated in proportion, base on the number of usability problems found.

We can also attach weights to those parameters, depending on which parameter is the most important for us. The weights should sum up to 1. If the cost is the most important we can give it the highest weight, so even though another method turns out to be more accurate and complete the final effectiveness for the cheaper technique will be higher. At the end the effectiveness of a technique can be counted as:

$$E = D*\text{weight}_1 + K*\text{weight}_2 + c*\text{weight}_3 + t*\text{weight}_4$$

To illustrate that I will present a comparison between the two usability techniques used to evaluate the website platel.pl.

3. THE EXPERIMENT

To check the evaluation method we have performed a usability test of a SOA-based web application platel.pl. We used two techniques – expert evaluation [7] and focus group evaluation [6]. In the expert evaluation a group of "experts" evaluates the system in terms of its construction, consistency and aesthetic aspects of website, communication between user and website, errors on website and the effectiveness of the website. In a focus group research a group of users (10–20 people) work with the system and perform some tasks in it. All this time they are communicating with a moderator and they discuss their work with the system. The participants mention the problems and difficulties that they have encountered and their general feelings after working with the system.

The first test that we have conducted was the expert evaluation. For this test we have selected three experts with different usability evaluation experience. During one week time, the experts evaluated the platel.pl system and have created a report describing each usability problem that they have found. Figure 2 shows an example of a found usability problem.

The second test was a focus group evaluation. We have evaluated a group of 14 people. Six of them were the developers of particular parts of the platel.pl system, and eight of them were the IT students from our university. This allowed us to have both,

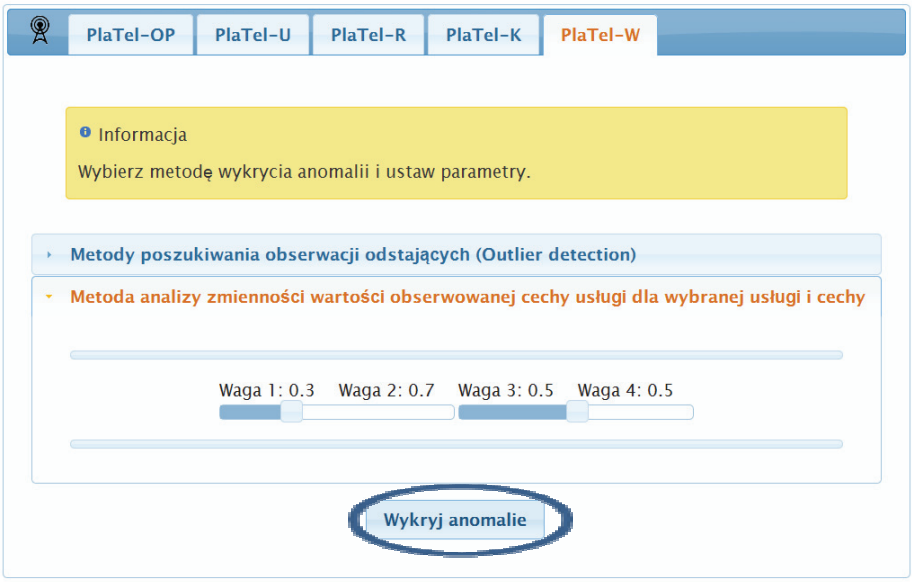


Fig. 2. Example of the usability problem with platel.pl system
– after pressing the button we do not get any feedback from the system,
so we don't know whether the operation was successful or not

users who knew the system and those who used it for the first time. Both groups have been tested separately. During both tests there was one moderator and two assistants that were writing down the remarks made by users. The tests duration was about one hour each. During this time users had some time to browse the system and they also performed tasks in the systems prepared by the moderator. They were encouraged to discuss and mention everything that they liked and disliked about the system. After the focus group evaluation a report was created describing the usability problems that were encountered by the users – analogously as in the expert evaluation.

4. RESULTS OF THE EXPERIMENT

After performing usability tests of the platel.pl system with both techniques and analyzing the created reports we can assign the obtained results in a table as shown below:

Table 1. Results from the both tests

	Expert evaluation	Focus group evaluation
Accuracy	8 critical problems, 7 medium and 4 minor	11 critical problems, 7 medium and 7 minor
Completeness	19 problems	25 problems (11 same as in the expert evaluation)
Cost	3 experts each working for 20 hours	1 moderator working for 40 hour, 2 assistants working for 4 hours each
Time	20 working hours	40 working hours

In expert evaluation we have found 19 usability problems. Eight of them were critical, 7 medium and 4 minor. During focus group evaluation users found 25 usability problems. Eleven of them were critical, 7 medium and 7 minor. Eleven same problems have been found by both tests. For both techniques the cost was only the payment for the people that concluded the tests, but because the respondents were all students and employees of our university, we did not have to pay them for the participation, but in a normal test, participants fee would be around 40 zł which multiplied by 14 participants would be 560 zł. Expert evaluation has lasted only 20 working hours, whether focus group evaluation 40 working hours. Comparing the both techniques properties we can see that:

$$D_e < D_f$$
$$K_e < K_f$$
$$c_e < c_f$$
$$t_e < t_f$$

where index e – expert evaluation technique and f – focus group evaluation.

During the focus group evaluation more usability problems has been found and that method accuracy was better. Also focus group technique required more time and consumed more money.

The next step is to assign values to the parameters. The parameters D and K for the focus group (with f index) will all be 1. The cost and time parameters will be 1 for the expert evaluation method (e index). We can calculate the D_e parameter by comparing the minor and medium parameters found. With 11 problems found during expert evaluation and 14 with focus group evaluation we can assign to D_e a value of 0.78 (11/14). K_e value can be calculated by comparing the number of different usability problems found (8 and 14), so K_e will be 0.57. For the cost let us assume that each

working hour cots 15 zł. For the expert evaluation cost would be 900 zł. For the focus group evaluation 1280 zł (with the addition of the payment for the participants). The parameter c_f value would be 0.7 in this example. Finally the last parameter is t_f, which is 0.5 (twice as much time).

Now we can add weights to the parameters. For this system and this case study we can assume the *D* and *K* parameters are the most important so I will assign to them weights 0.35 and 0.35. The cost is more important than time, so the cost will be 0.2 and the time 0.1, so the all weights would sum up to 1. Having everything ready we can finally calculate the efficiency of the both methods:

$$E_e = 0.78*0.35+0.57*0.35+1*0.2+1*0.1=0.7725$$

$$E_f = 1*0.35+1*0.35+0.7*0.2+0.5*0.1=0.89$$

Base on that we can say that that the focus group technique effectiveness was higher that the effectiveness of the expert evaluation for the platel.pl system. If the values of weights would be different, for example the most important aspect would be the cost, than the expert evaluation might turn out to be more effective for our needs.

5. SUMMARY AND FUTURE WORK

Presented method is very simple and it allows to easily compare different usability techniques. It can be used to check the effectiveness of many different usability testing techniques, such as eyetracking, remote tests etc. It can also be extended with some additional parameters or measures [9]. For example we can compare two techniques which both require user's participation, so we can add other parameters to the method such as number of tested users etc. The introduction of weights for particular parameters enables to customize which of them are the most important in particular situation. We can of course just compare both techniques without using the weights by comparing every parameter with each other.

Moreover we are planning to introduce the evaluation of the statistical importance of the found usability problems, while comparing different techniques. For such evaluation we can use for example the Fisher's exact test [8]. It would allow to have a better comparison of the usability problems found with different techniques.

Presented work is just a first step in usability techniques effectiveness evaluation, but such research can be very useful for developers working on a particular system. After checking various methods of usability testing they can choose the most effective for their system and use only them in the future tests, thus saving money and time.

REFERENCES

[1] BARNUM C.M., *Usability testing and research*, Longman, 2002.
[2] BARNUM C.M., *Usability Testing Essentials: Ready, Set ...Test!*, Elsevier, 2010.
[3] DUCHOWSKI A. T., *Eye tracking methodology: Theory and practice*, London, Springer-Verlag Ltd.
[4] International Standard ISO 9241-11. *Ergonomic requirements for office work with visual display terminals (VDTs) – Part 11: Guidance on Usability*. ISO 1997.
[5] KASPERSKI M., BOGUSKA-TORBICZ A., *Projektowanie stron WWW. Użyteczność w praktyce*, Helion 2008.
[6] KRUGER, R.A., CASEY, M.A., *Focus groups: A practical guide for applied research*, Thousand Oaks, CA: Sage, 1999, 107–113
[7] PEARROW M., *Funkcjonalność stron internetowych*, Helion, 2003.
[8] FISHER R.,A., *Statistical Methods for Research Workers*, 14th edition, Hafner Publishing, 1970, 96.
[9] TULLIS T., ALBERT B., *Measuring the user experience*, Morgan Kaufmann, 2008.

virtualization, private cloud,
cloud service orchestration

Krzysztof BIELAWSKI*, Mariusz PRÓSZYŃSKI**

AUTOMATING THE VIRTUAL PRIVATE CLOUD CREATION WITH USE OF WEB SERVICES AND WORKFLOWS

This chapter presents method for cloud service orchestration with used of workflows, which efficiently scale out administrative workload of private cloud creation. Presented solution utilize the VMware API orchestrator's workflows and web services in order to provide the interface to self-service environment of business application systems. Chapter concept of automated orchestrating of storage, network, and virtualization technology in order to enable the dynamic placement of multi-tier services on public or private cloud infrastructure.

1. THE PRIVATE CLOUD

Cloud computing is an Internet based service which delivers network, computing, storage capacity, security provisioning, maintenance etc. to the organization. This approach is possible by sophisticated automation, provisioning, management, and virtualization technology which is a fundamental component of the cloud computing architecture stack and differs dramatically from the "old school" IT model, because it decouples data and software from the physical infrastructure that runs them. For cloud computing, commonly accepted definitions are defined [1-3] for the deployment models and there are several generally accepted service models. Figure 1 illustrates models on which this chapter focus. The cloud infrastructure operates solely for an organization and it is managed by the organization or a third party. This infrastructure may be on-premise or off-premise.

* Białystok University of Technology, Faculty of Computer Science, Wiejska 45A, 15-351 Białystok.
** Intratel Sp. z o.o., 1000-lecia Państwa Polskiego 39A, 15-111 Białystok.

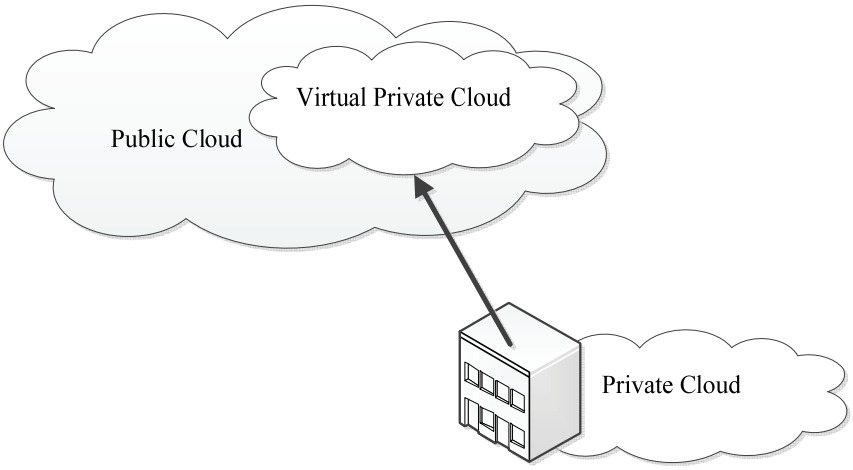


Fig. 1. Private cloud computing ecosystem

However, it can also be created a Hardware Virtual Private Network (VPN) connection between organisation datacenter and its Virtual Private Cloud (VPC) as an extension of corporate datacenter. In such a case it allows a number of physical servers to be pooled into a large computing resource that can be used to run as many virtual machines of almost any size as are needed at any given time.

Virtual environments become increasingly complex, most companies believe that automating the virtual data center to deliver private clouds is complex and time consuming [2, 3]. However they also knew that costumer expectation lays in simplicity of business IT environment creation, which is served by automation and it is an absolute requirement for an efficient and effective data center. Whereby, resources are dynamically provisioned via publicly accessible Web application/Web services (SOAP or RESTful interfaces) from off-site provider.

There are many examples for vendors who publicly provide infrastructure as a service within automation, like: Amazon Elastic Compute Cloud (EC2) [4], GoGrid [5], Joyent Accelerator [6], Rackspace [7], AppNexus [8], FlexiScale [9], and OVH [10]. This portals provide a control of your computing resources and lets you run computing and infrastructure environment easily. It reduces the time required for obtaining and booting a new server's instances to minutes, thereby allowing a quick scalable capacity and resources, up and down, as the computing requirements change. Service offers different instances size according to the resources needs, the CPU's needs it provides, and high-memory instances.

Additionally, it lets provisioning of a private, isolated section of the cloud where you can define a virtual network topology that closely resembles a traditional network, and

complete control over virtual networking environment, including selection of your own IP address range, creation of subnets, and configuration of route tables and network gateways.

2. AUTOMATED IT SERVICE DELIVERY AND MANAGEMENT

In a non-automated world, an IT consumer, requests server or set of servers, which requires resource: network, storage, compute and security provisioning, etc and have been delivered by supplier. In virtual environment there are many steps to process such request, but they are very static in the way, they are carried out. These generally include automated provisioning and lifecycle management, with crucial elements such as: the initial request for service, deployment of virtual machines (VMs), IT services, approving and rejecting requests, and fulfilling requests for change including decommissioning and archiving from a user-appropriate catalog.

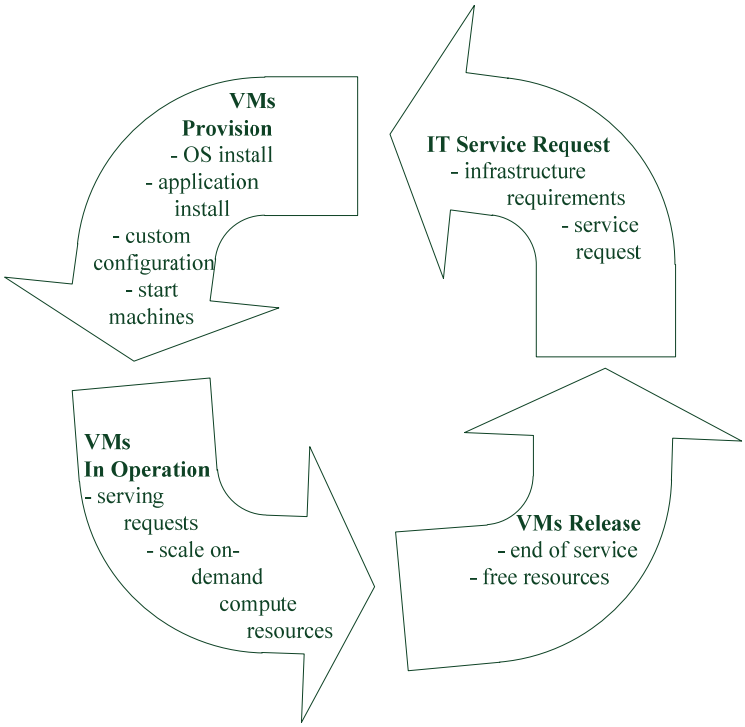


Fig. 2. Virtual machine life cycle

This lifecycle of virtual machine is depicted at fig. 2. and it is first step of automation in virtual environment. However, such approach delivers customer the stable environment of deployed machines in predefined environment, but further customization and additional configuration still must be done. Customer experience has shown that the best approach is to implement second stage of gradual automation, which introduce operating system further customization of defined resource. This step introduce private cloud to firms such as small and medium enterprises (SME), which enable them to deploy all application that are critical for the business daily operations. Many project of virtualization implementation fail in case of business requirements lead to the insufficient hardware resources, improper storage configuration for specific application type or even network interconnection of system elements. Additionally, its require plenty of IT specialist working hour to set up environment, even when the first step of automation is used.

However, cloud it is an orchestrated environment, the IT costumer would log into a portal and input the desired specification for the compute environment required. Assumed orchestration involves tying compute resources, application and processes together, and after defining the various field, the system of the cloud would then deliver the business ready environment – this define the second step of automation presented in this chapter.

3. PROOF OF CONCEPT

To demonstrate the feasibility of presented idea the Microsoft tier-1 applications has been virtualized in vSphere 5.0 cluster. Due to the experience of this type of business environment, the scale-out approach has been used for deployment strategies consist in multiple small virtual machines. This strategies is more appropriate for virtual VMware vSphere environment and provide easier customization of virtual machines and application configuration as needed. Moreover, it provides better workload and security isolation and more granular change management, also works very well for horizontal scalability, load balancing, and high availability [11] (DRS[1] and vMotion migration). However, with one exception, a scale-up approach was applied to SQL Server, that can utilize the resources provided in this approach.

Presented solution incorporate VMware vCenter Orchestrator [12], which provides out of the box workflows to automate vSphere environment manual tasks. Then with

[1] VMotion™, VMware DRS (Distributed Resource Scheduler) dynamically allocates and balances computing capacity and virtual machine placement with resources pooled from multiple ESX Server hosts.

used of Orchestrator's library, additional workflows has been proposed for second stage of automation – fig. 3 illustrates small part of them.

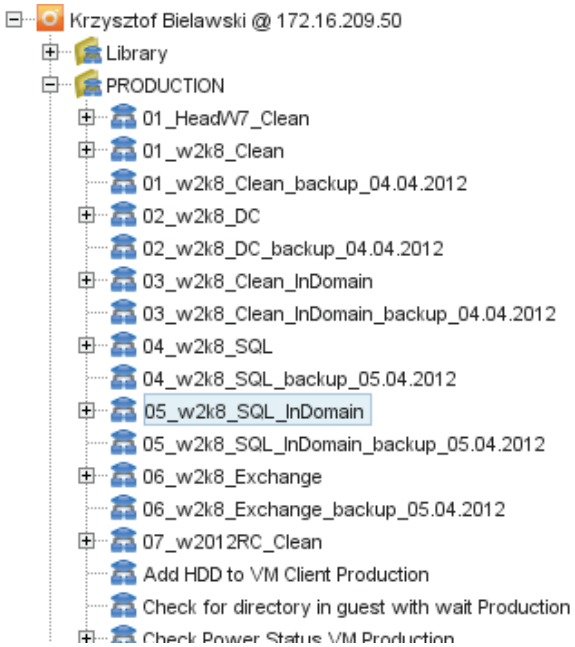


Fig. 3. Workflow library created for second step automation

The Active directory and the domain controller has been assumed for environment so the main workflow `02_w2k8_DC` is always perform, even when it is not pointed directly by the user's order.

Then all other ordered machines are edified with use of listed workflows:

03_w2k8_Clean_InDomain,
05_w2k8_SQL_InDomain,
06_w2k8_Exchange.

The assistance work is done by the following workflows:

Deploy-CLEAN-W2K8-ENT-x64-ENG,
Deploy-CLEAN-Win7-ENT-x64-ENG,
Add HDD to VM Client Production,
Initialize HDD in Guest Production,
Join to Domain PRODUCTION - registering virtual machine in Domain,
Set IP on VM Production,
Copy file from vCO to guest - coping Power CLI scripts to run in VMs,
Run program in guest,

Final Clean VM - finalizing initiation and registering VMs in environment,

Delete VirtualMachine Production -removing machine from environment and deleting it form storage.

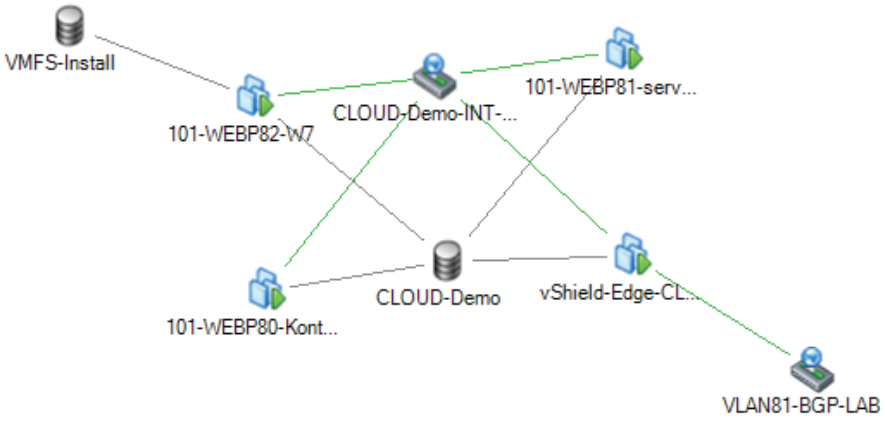


Fig. 4. Example client's network layout

To construct the working and stable environment, which reconfigures on demand in self-service setup and delivers the high level security till setup time, the architecture depicted in fig. 4. has been used. The cloud's private virtual network with ingoing flow separation is used till one initial access point has been setup with Remote Desktop Protocol to first crated machine. Communication in clients LAN of machines are not restricted, however all machines are started with predefine policy of firewall, which is running.

Figure 4 illustrate the initial setup in case of three machines in environment: `101-webp80`-domain controller, `101-webp81`-MS SQL server, `101-webp82w7` - Windows 7 desktop. Where VLAN (Virtual LAN) `Cloud-Demo_INT-101` is a clients no. 101 private VLAN and portgroup `VLAN81-BGP-LAB` represent infrastructure VLAN to the internet. When order is placed for the environment the virtual machine `vShield-Edge-CLOUD-INT-VLAN101` is constructed automatically. This machine plays a gateway role with NAT, firewall and site-to-site VPN and DHCP services to the clients network and is setup through RESTful interface. Due to the fact that vSphere 5.0 distributed switch acts as a single switch across all associated hosts on a datacenter, each VLANs enable a single physical LAN segment to be further segmented, so that groups of ports are isolated from one another (standard 802.1Q). Assumption was taken that this features of virtualization belongs to basic automation workflows, so it is not discussed in this chapter.

Created workflows are divided in two groups: one for virtual machine provisioning and operating system setup, second for operating system configuration, environment creation, registering and connecting workflows. To demonstrate the automation done and to show differences between the out-of-box orchestration of virtual environment and business working environment creation, the Exchange application have been chosen, as a most representative, there is no available out-of-box process to setup an Exchange. Applications such as Microsoft Exchange is one offered applica-

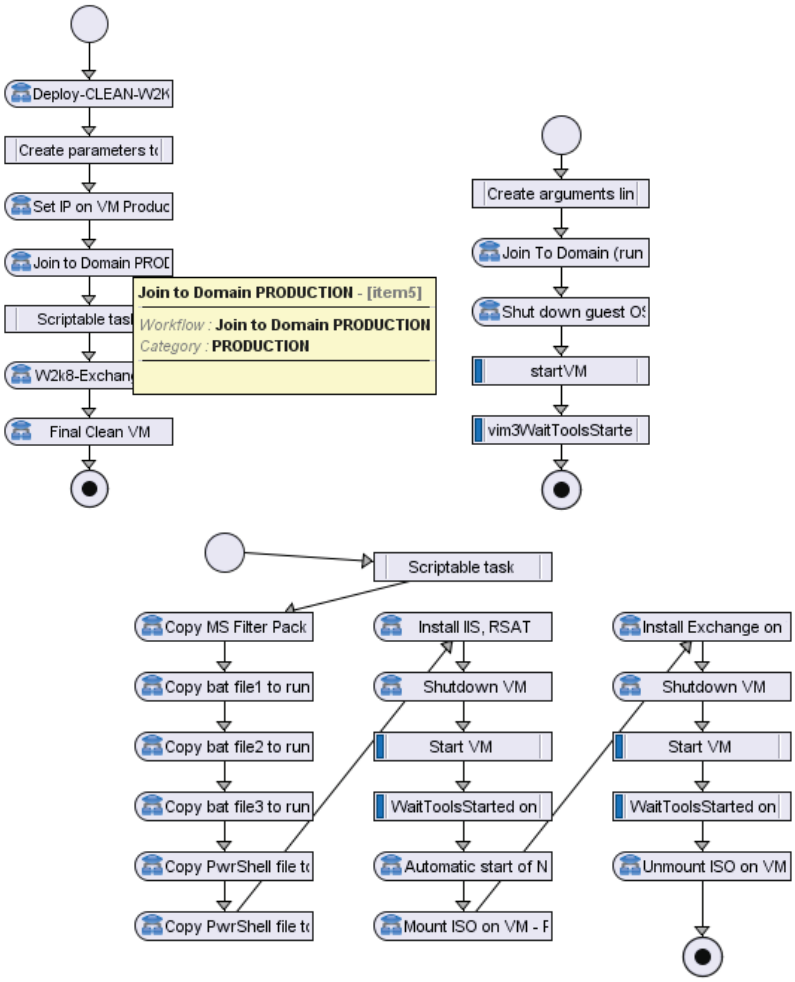


Fig. 5. Workflow to deploy the Exchange server

tion that most modifies the Active Directory and makes more than 100 changes/additions to the Active Directory schema to prepare for the Exchange messaging system [13]. Although these the workflow schema updates and modifications have been made possible on the production environment without user intervention. The fig. 5 illustrate constructed workflow for Exchange setup, where on the left we have main process and on the right the registering Exchange server in domain. Initialize block takes a resource pool element delivered by user through simple web form and by constructed environment by itself. Fig. 6 depict the binding input variables used in workflow.

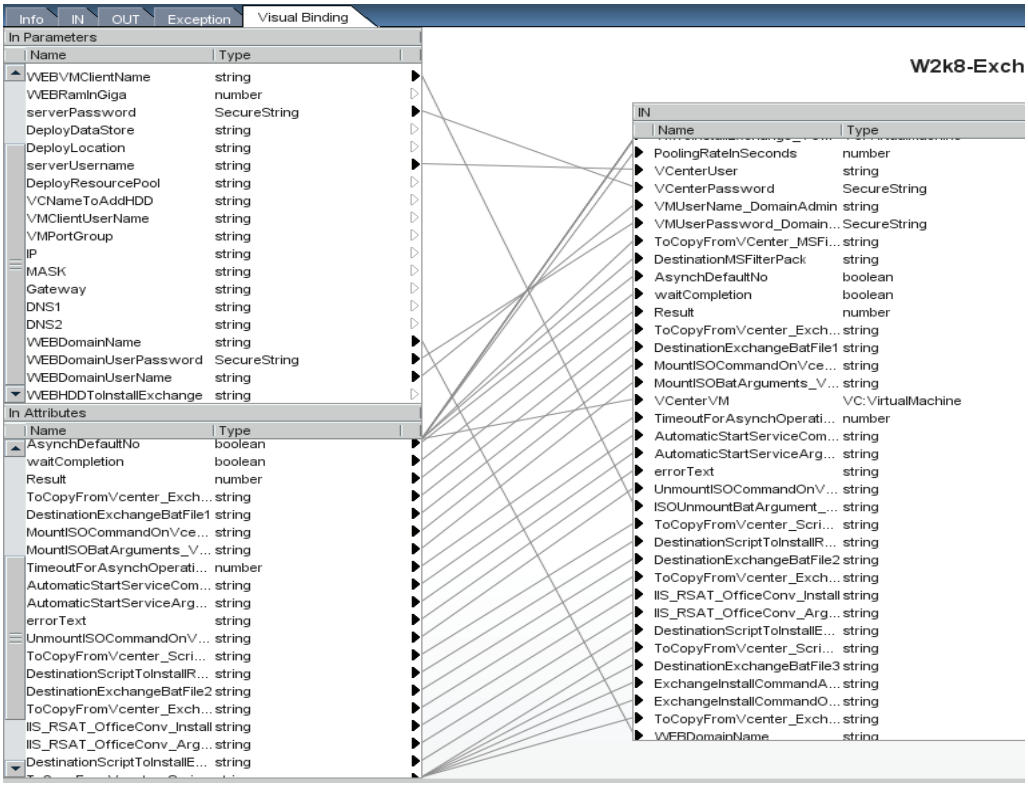


Fig. 6. Input banding of Exchange setup workflow

Presented solution for future automation of business IT environment construction allows to build the web service catalogue for virtual private cloud solution illustrated at fig. 7.

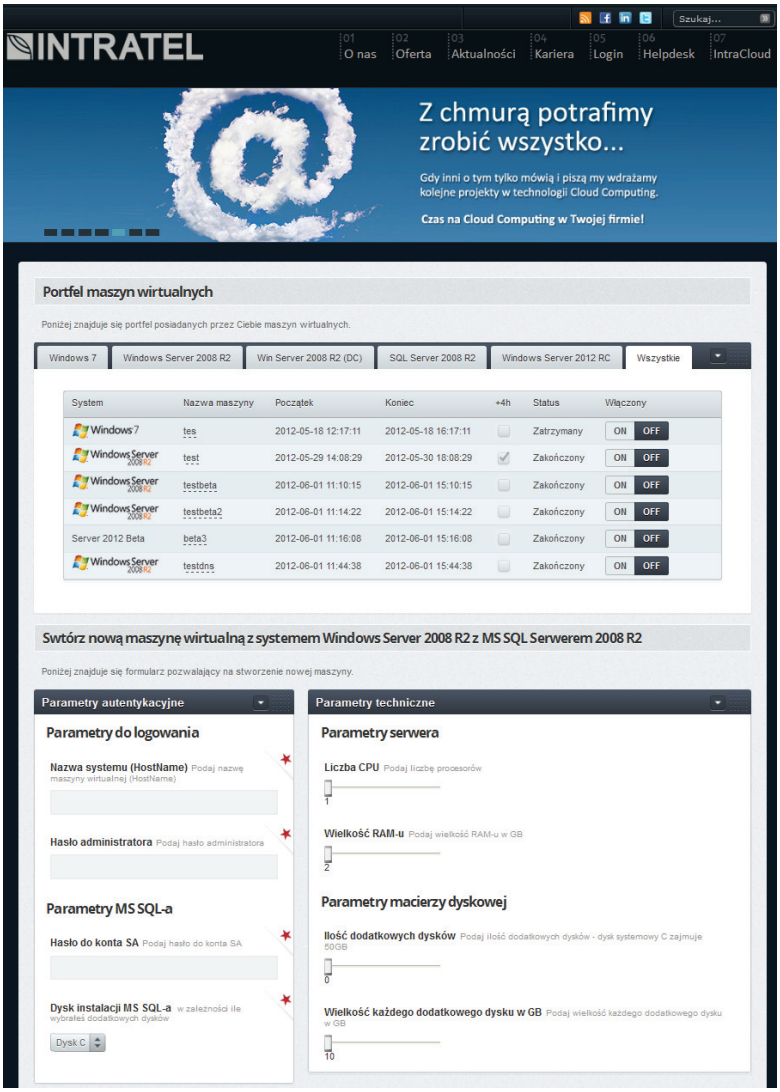


Fig. 7. Service catalogue for cloud computing solution
with use of automation provided by Intratel

CONCLUSION

This chapter propose next step of automation of cloud services provided in virtual cloud model, which trying to overcome the usage barrier caused by the technology

mixed with self-service idea in cloud computing for SME. Therefore presented approach is taken to overcome this difficulties. This model of second stage automation proving the availability and usefulness of such concept to orchestrating of vSphere 5.0 with use of orchestrator's tool, RESTfull interface and VIX API provided by VMware virtualized environment. At the time of writing, presented solution as a service catalog has been prepared for publishing to be public[2] in form of time limited demo.

ACKNOWLEGEMENT

This work was supported under the grant UDA-POIG.01.04.00-20-002/11.

REFERENCES

[1] NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY, *Working Definition of Cloud Computing*, 2009.
[2] ARMBRUST M., FOX A., GRIFFITH R., at all, *Above the Clouds: A Berkeley View of Cloud Computting*, Technical Report No. UCB/EECS 2009 28, University of California at Berkley, USA, Feb. 10, 2009.
[3] GILLET F.E., *Conventional Wisdom is Wrong About Cloud IaaS*, 2009.
[4] *Amazon Elastic Compute Cloud (Amazon EC2)*, http://aws.amazon.com/ec2/, June 5, 2012.
[5] *Cloud Hosting, Cloud Computing, Hybrid Infrastructure from GoGrid*, http://www.gogrid.com/, June 5, 2012.
[6] *Joyent Cloud Computing Companies: Domain, Application & Web Hosting Services*, http://www.joyent.com/, June 5, 2012.
[7] *Rackspace hosting*, http://www.rackspace.com/index.php, June 5, 2012.
[8] *AppNexus Home*, http://www.appnexus.com/, June 5, 2012.
[9] *FlexiScale cloud computing and hosting: instant Windows and Linux cloud servers on demand*, http://www.flexiscale.com/, June 5, 2012.
[10] *OVH Cloud Computing*, http://Ovh.pl, June 5, 2012.
[11] *VMware Infrastructure Resource Management with VMware DRS*, http://www.vmware.com/pdf/vmware_drs_wp.pdf, June 5, 2012.
[12] BUNCH C., *Automating vSphere with VMware vCenter Orchestrator*, VMware Press, Pearson Education Inc., 2012.
[13] WINDOM C.A., GAIDHAIN H., *Virtualizing Microsoft Tier 1 Applications & VMware vSphere 4*, Wiley Publishing, 2010.

[2] For more information please visit - http://onestepcloud.com

rapid development, database system
UML, Spring Roo

Bogumiła HNATKOWSKA, Sebastian BIEŃ, Maciej CEŃKAR*

RAPID APPLICATION DEVELOPMENT WITH UML AND SPRING ROO

Model-Driven Development (MDD) and Domain Specific Languages (DSLs) are becoming more popular last years. These techniques try to maximize the benefits of modelling in many ways, e.g. by eliminating the gap between analytic and design models, and by producing working code directly from models. In the paper an approach to combine classical, visual modelling with UML (preferred by system analysts) with the textual Spring Roo DSL (used by developers) is proposed. The approach aims at rapid development of data-oriented web applications, in which the main functionalities allow to create, delete, update, and retrieve both objects, and links between them. The aspect of user authentication and authorization is also taken into account.

1. INTRODUCTION

Rapid application development is a software development methodology that involves methods like iterative development and software prototyping [1]. There are two types of software prototyping, i.e., rapid prototyping (throwaway) and evolutionary prototyping [1, 2]. The paper deals with evolutionary prototyping, which is developed as a part of the actual system. New features are implemented as the development proceeds in an iterative manner. The prototype is developed to be of production quality [3].

Rapid prototyping is typically supported with specific tools and frameworks. They allow reducing the number of decisions that a developer must make in building an application [4]. Spring Roo belongs to them.

Spring Roo is a rapid application development tool for Java developers. With Roo you can easily build full Java applications in minutes [5]. Spring Roo focuses mainly

* Wroclaw University of Technology, Wybrzeże Wyspiańskiego 27, 50-370 Wrocław.

on entity layer [4] thus it fits in natural way for development of data-oriented web applications.

The main element of data-oriented Web applications is the effective access and management of the data. Such kind applications usually have simpler business logic, and for them the realizations of data input, processing, and data presentation consumes a considerable amount of production time [6].

To support the development of data-oriented web applications Spring Roo offers a set of textual domain specific languages (DSLs). It is enough to write a short textual command to generate substantial amount of source code. However, the Spring Roo lacks in providing readable tools for see the system from bird perspective; to organize classes and relationships between them.

UML [7, 8] is a commonly accepted language with plenty of tools used for application specification and design. Very often, especially if the number of classes increases it is selected as a first option choice to model both: the requirement and the domain. The UML models can be written in XMI notation, which enables the model transformation to any implementation language and/or platform.

The paper shows how to join the benefits of graphical modeling (especially when the model is not a trivial one) with the generation possibilities of Spring Roo framework. The solution is dedicated primarily for intensive data-base systems for which CRUD use-cases are the necessary elements. A developer can easily model basic functionalities, provide data model, and – at the end – obtain working solution, even deployed in the target environment.

The paper is structured as follows. Section 2 presents a general idea of proposed solution to rapid development of Web applications. Section 3 describes a UML profile, prepared by us, being an important part of the solution. Section 4 brings short but representative case-study. Related works are discussed in Section 5. The last Section 6 concludes the paper.

2. INTEGRATION OF UML AND SPRING ROO

A model is a representation of a part of reality. It points out the important aspects of the thing being modeled and simplifies or omits the rest. The model is intended to be easier to use for certain purposes than the final system [8]. A model of a software system is made in a modeling language, such as UML. UML belongs to the most popular modeling languages used today, however, domain specific languages also become more and more popular. In software development and domain engineering, a domain-specific language (DSL) is a programming language or specification language dedicated to a particular problem domain, a particular problem representation technique, and/or a particular solution technique [9].

DSLs are very common in computing: examples include CSS, regular expressions, make, rake, ant, SQL, HQL, many bits of Rails, expectations in JMock, FIT, and strut's configuration file [10].

The main classification divides DSLs into internal languages (built into so called hosting language), and external languages, which have their own custom syntax and need a full parser to process them. The other classification divides DSLs into graphical and textual depending on the notation used.

The paper deals with two types of DSLs:

- External, graphical, elaborated by authors of this paper, used for modeling purposes (specification language) – it is represented as a UML profile, see Section III;
- External, textual used for rapid development, elaborated by Roo developers.

Spring Roo is latest ambitious attempt to bring rapid application development (RAD) to Java developers. Developers are equipped with plenty useful commands that allow them e.g. generate the project, generate the entity classes together with data-base support, and generate the user-interface pages to retrieve or input data [4].

Roo is a development-time only framework which is used for obtaining best-practice-oriented code. It delegates all the runtime handling to Spring and other frameworks, including JPA or Hibernate, JMS, Spring MVC, Spring Web Flow, and GWT. But, when it is necessary, the dependencies to Roo could be easily removed, leaving the application still working [11].

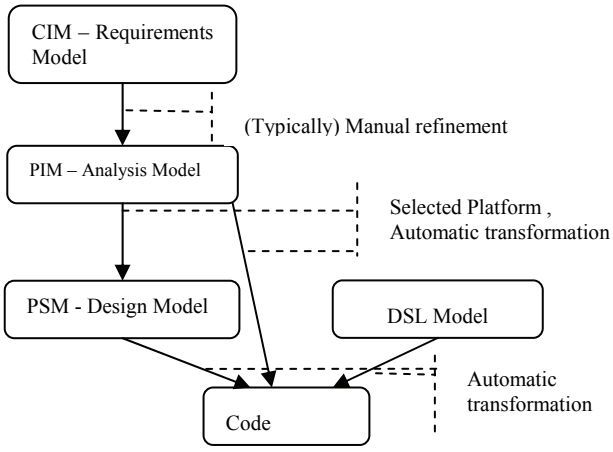


Fig. 1. Alternative approaches to model – source code generations

Regardless of the modelling language used, a model could be transformed either into another model (maybe platform specific) or into source code. Figure 1 presents the possible approaches to source code generation. One of them is MDA [12], and the second comes from MDE (Model Driven Engineering) [13].

MDA introduces tree kind of models, i.e. Computational Independent Model (CIM), Platform Independent Model (PIM), and Platform Specific Model (PSM). They represent the same system from different perspectives and on different abstraction levels (from the most general to the most specific). PSM is expressed in terms for targeted platform, e.g. Spring, or JPA, and is detailed enough to generate significant part of source code. Subsequent models are obtained manually (as the result of refinement) or automatically.

On the other side, DSL could be considered as PSM equivalent as it is also expressed in platform specific way (easy to understand to domain experts).

In proposed method to rapid development we recommend to join both approaches to code generation, presented in Fig. 1. The method itself consists of following steps:

1. Class diagram and use-case diagram definition
2. Applying selected stereotypes to UML elements (if necessary) (DSL 1)
3. Spring Roo code generation (DSL2)
4. Java code generation
5. Runnable version building and deployment

A developer starts with typical requirement specification and analysis. The outcomes of these activities are Use-Case Model and Analysis Model prepared in UML Case Tool, e.g. Visual Paradigm [14]. Next, he needs to decide about targeted platform. If Spring is selected, the developer can use our UML profile (see next section) to define all important aspects of data-oriented Web application. After that the model needs to be exported to XMI format. The XMI to Roo transformation produces a set of Roo commands which further is transformed to java code. Java code is compiled, and packed to war format deployable on www server.

The steps from 3 (XMI version of UML model) are fully automated by tools prepared by us. So, at once, providing that www server is running, the prototype is ready to work.

The proposed solution has following benefits:

- Small changes in the Spring framework don't influence the rest of tools providing that the Roo command syntax is the same or extended with the backward compatibility.
- It is easier to write PSM–Roo Code transformation than PSM–Spring Code transformation because much less lines are to be generated.
- Roo Code – Spring Code transformation is of very good quality, confirmed by lots of Roo users.

3. UML PROFILE FOR WEB DATABASE APPLICATIONS

Profile is a commonly used UML extension mechanism which enables the language adaptation for a specific purpose, e.g. modeling web database applications.

"Profile was defined in order to provide more structure and precision to the definition of stereotypes and tagged values" [7], other UML extension mechanisms of smaller granularity.

To allow efficient utilization of Spring Roo possibilities, a new UML profile for Web database applications was elaborated. The profile consists of three parts – the first serves for model organizing, the second addresses the problem of requirement specification, while the third is used for marking an analysis model with platform details.

The UML diagrams for which the profile can be applied are as follows: package diagram (presenting general structure of the solution), use-case diagram (presenting the actors and use-cases), class diagram (presenting the main domain entities).

3.1. GENERAL PART

The stereotypes and tagged values defined here address the architecture of the system, and are used on package diagram.

All modelling elements should be placed into a subsystem with *<<PrototypedProject>>* stereotype. Additionally, several tagged values for that subsystem can be defined, e.g. topLevelPackage or java version – see Fig. 2.

The *<<PrototypedProject>>* stereotype is transformed into following Spring Roo command:

```
project --topLevelPackage pl.wroc.pwr.wiz.example --projectName prototype
--java 7
```

In a case when a developer is interested in storing the data into a database with the use of Java Persistence API implementations (e.g. Hibernate), one needs to provide a subsystem with *<<PrototypedPersistence>>*. He/she has the possibility to define all important elements of database connection string as tagged values.

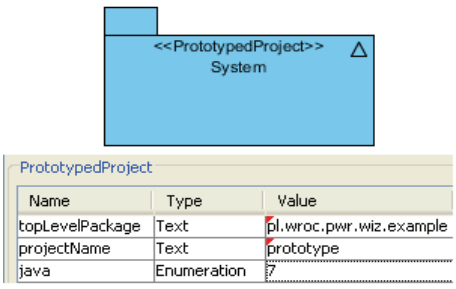


Fig. 2. *<<PrototypedProject>>* stereotype with its tagged values

3.2. REQUIREMENTS PART

The stereotypes and tagged values defined here address functional requirements for the system, and are used on use-case diagrams.

Create/Retrieve/Update/Delete (CRUD) operations are typical ones in data-oriented Web applications. CRUD operations could relate to both: entities (objects) and links between them. Thus, to represent CRUD operations for entities following stereotypes can be applied to any use-case: *<<CreateEntity>>*, *<<ListEntity>>*, *<<EditEntity>>*, *<<DeleteEntity>>*. The stereotypes are used together with obligatory tagged value, pointing to the related class from analysis model. To represent CRUD operations for links we propose to use following stereotypes:

- *<<AssignReference>>* – with *SourceEntity*, and *TargetEntity* as tagged values; the stereotype describes a possibility to create, update and delete links between related entities
- *<<ListReference>>* – with the same tagged values as the previous stereotype, used to define the possibility to show target entities related to the source entity.

We treat an actor as a role which has some accessibility grants to perform specific functions. When a generalization between actors is defined, the child actor inherits associated use-cases (grants). This feature is included in the generated prototype.

3.3. ANALYSIS PART

The stereotypes and tagged values defined here address analysis (only domain entities), and are used on class diagrams.

The stereotypes can be applied to classes (e.g. *<<PrototypedClass>>*), fields of simple types (e.g. *<<PrototypedBoolean>>*, *<<PrototypedDate>>*, *<<PrototypedNumber>>*), and association ends (*<<PrototypedAssociationEnd>>*).

Together with stereotypes several tagged values were defined that could be mapped to Roo commands, e.g. for *<<PrototypedNumber>>* that are: *notNull = false|true*, *column = text* (the column name in a database), *min = value*, *max = value* (definition of the range of correct values), *digitsInteger = value* (number of digits before a dot), *digitsFraction = value* (number of digits after a dot). Additionally, some UML specification elements are also taken into account, e.g. *unique* property defined for the attribute.

Visualisation of *<<PrototypedAssociationEnd>>* stereotype together with its tagged value is shown in Fig. 3.

Product class will remember its supplier reference, and Supplier class will remember a set of delivered products. Products will be read either together with supplier instance (*fetch = EAGER*) or on demand (*fetch = LAZY*).

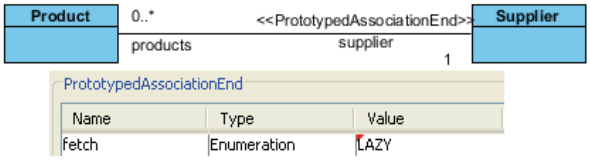


Fig. 3. <<*PrototypedAssociationEnd*>> stereotype with its tagged value

The following elements of UML class diagrams are in further translated to Roo DSL language: classes, binary associations (one to one, one to many, many to many), association classes, and single inheritance. Compositions, and aggregations are treated as ordinary associations. The more advanced UML elements like n-ary associations, qualifiers, multi-sets are not supported.

What is interesting, the proposed stereotypes for a system and classes are assumed implicitly with default values for all defined tagged values. It means that the analysts is forced to used them only if he/she needs to replace default values for tagged values with new ones. Such approach to model definition allows sparing time, and reducing cost of model development.

4. CASE STUDY

This section presents the properties of proposed solution for easy to understand but meaningful example – Conference Management System. The main domain classes used within the system are presented in Fig. 4.

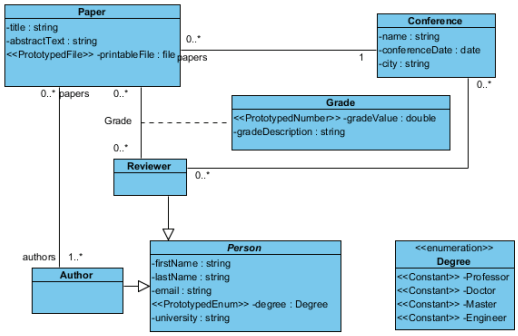


Fig. 4. Class diagram for Conference Management System

Only two of proposed stereotypes (see Section III) were used here explicitly in order to define tagged values. For example, for <<*PrototypedFile*>> stereotype the *notNull* tagged value was set to *true*, and the *contentType* tagged value to *PDF*.

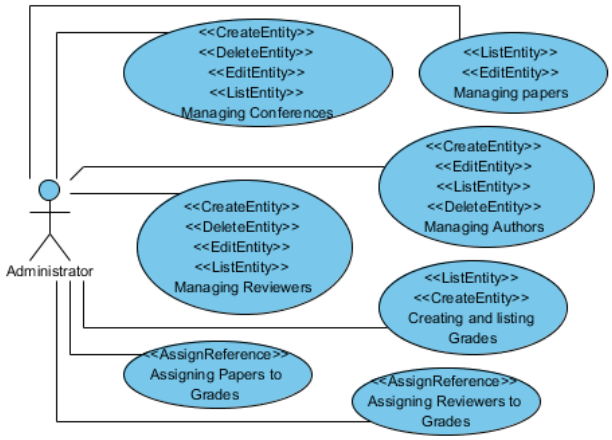


Fig. 5. Administrator use-cases together with stereotypes applied

The conference could have many papers submitted. Each paper is written by one or more authors. The conference has some reviewers involved; each one could have assigned many papers. One paper can be reviewed by many reviewers. The review is represented as Grade association class.

We would like to have 3 actors interacting with the system:

- *Author* – who is interested in submitting papers to the conference and reading the reviews when they are completed
- *Reviewer* – who checks papers assigned to him/her and provides reviews for them
- *Administrator* – who is responsible for managing all important entities – see Fig. 5, e.g. conference data (*Managing Conference* use-case), authors and reviewers data (*Managing Authors*, *Managing Reviewers*). Additionally, administrator prepares blank reviews (*Creating and Listing Grades*), assigns papers to reviewers (this functionality is split into 2 use-cases: *Assign Reviewer to Grade*, and *Assign Paper to Grade*).

It must be mentioned that the model is partially inconsistent with the target version of the system requirements. We would like the authors to register themselves on the conference site. At that moment it is impossible – administrator is the only role who can create an author's account.

On the base of the class and use-case diagrams Spring commands are generated and further translated into java code. We automatically obtained several files including 22 java classes – see statistics presented in table 1.

Figure 6 presents a form allowing an author to submit a paper in pdf format.

Table 1. Basic statistics of the generated files

Programming language	File number	Lines of code
Java	22	3429
HTML	14	1096
XML	7	653
CSS	1	72
Javascript	1	54
Total	45	5304

Fig. 6. Author's form allowing him/her to submit a paper

We assess that to make the application fully functional we need to extend the code generated up to 10%. Registration process is the only one that needs to be added from scratch. Other extensions relate to implementation of business rules, e.g. an author can't be deleted when he/she has some papers submitted.

5. RELATED WORKS

The model-driven approach to software development is addressed in many papers. In this section we would like to present the works most bound up to our approach, from which some inspired us.

The authors of the paper [6] "propose a systematic design method for Web applications which takes into account the data-oriented aspects of the application". The method is based on a UML profile (set of stereotypes) which allows to define so called

conceptual model as well as to show the navigations between the screens presenting the classes' instances. The UML model is transformed into XMI format and further translated with the set of other transformations into source code, e.g. data-base scripts. However, the paper presents only the method (idea) without any tool support.

Authors of [15] defined and implemented a domain-specific language called WebDSL for the definition of web applications. The DSL includes data-models, user interfaces, and actions. Actions present the application reaction on invalid data. The WebDSL compiler generates a complete implementation in java. The proposed DSL is a textual language, very similar to OCL language.

Another attempt to rapid application prototyping is presented in [16]. The code in java (java applet) is generated on the base of use-case model and analysis model, without a use of any intermediate stage.

The most similar to our approach is proposed in [17]. Model2Roo presents a method to model driven development of Web applications built upon the Eclipse Modelling Framework and Spring Roo. The developer starts with modeling activities, next he/she transforms the model into Spring Roo commands. In further he/she uses another transformation to obtain the source code. The main differences between Model2Roo and our approach are listed below:

- Model2Roo generates the source code only for class diagrams, while our solution also for use-case diagrams
- Model2Roo doesn't support association classes nor CRUD for associations
- Model2Roo supports JavaServer Pages (JSP) standard for dynamic pages representation, our solution supports Java Server Faces (JSF) Model2Roo supports only diagrams prepared in Eclipse, while our approach allows to use any UML tool that exports the UML model to XMI format
- Model2Roo doesn't support authorization and authentication mechanism.

6. CONCLUSIONS

Rapid development serves in obtaining valuable results that can be presented to the end-user very fast. In the case of evolutionary approach to rapid prototyping additionally the obtained source code is a part of a target solution. The paper presents an approach to evolutionary rapid prototyping of data-intensive web applications. The main idea behind the approach is to combine the benefits of UML modeling with fast source code generation for specific platform.

The main elements of data-base applications are entities and relationships between them, while the majority of functions deal with so called CRUD operations. To enable a system analyst to visually present the elements one is interested in, a UML profile with several useful stereotypes and tagged values is provided. The profile is generally described and was implemented as a part of very popular Visual Paradigm CASE tool.

The number of diagrams that must be prepared to generate a running prototype is limited to two: use-case diagram, and class diagram. The analyst doesn't need to be a programmer nor a specialist in databases. Once diagrams are prepared, the model of a system is exported to the XMI format, and transformed by XSLT transformation first into a set of Spring Roo commands, and next into a 'war' file, which can be deployed into a www server. The process of both transformations is fully automated by prepared batch files. It must be mentioned, that – on the base of analytic models – also the mechanisms for user authentication and authorization are generated.

Because the process of source code generation is split into two parts (UML to Spring Roo DSL, and Spring Roo to Java Spring) it is resistant to possible changes in Java Spring provided that DSL part becomes unchanged, and can be adapted to other DSLs. The complexity of UML to DSL transformation for sure is much less than the transformation complexity of UML to a source code native for a selected platform.

The tool usability was evaluated by a questionnaire. The potential users found the tool attractive, and would be interested in using it in everyday routine. We plan to encourage students of our faculty to use the tool within Software Engineering course.

REFERENCES

[1] *Rapid application development*, http://en.wikipedia.org/wiki/Rapid_application_development, 2012.
[2] FORWARD A., BADREDDIN O.B., AND LETHBRIDGE T.C., *Umple: Towards combining model driven with prototype driven system development*, in Proc. International Symposium on Rapid System Prototyping, 2010, 1–7.
[3] FU J., BASTANI F. B., YEN I. L., *Model-Driven Prototyping Based Requirements Elicitation*, B. Paech, C. Martell (Eds.): MontereyWorkshop 2007, LNCS 5320, 2008, 43–61.
[4] FISHER P. T., MURPHY B. D., *Spring Persistence with Hibernate*, Apress, 2010.
[5] *Spring projects, Spring Roo site*, http://www.springsource.org/spring-roo, 2012.
[6] ADAMKO A., *Modeling Data-Oriented Web Applications using UML*, in Proc. of EUROCON 2005, Belgrade, 2005, 752–755.
[7] OMG Unified Modeling Language (OMG UML), Superstructure Version 2.3, http://www.omg.org/spec/, 2010.
[8] RUMBAUGH J., JACOBSON I., BOOCH G., *The Unified Modeling Language Reference Manual, Second Edition*, Addison-Wesley, 2004.
[9] *Domain specific language*, http://en.wikipedia.org/wiki/Domain-specific_language, 2012
[10] FOWLER M., *Domain Specific Language*, http://martinfowler.com/bliki/DomainSpecificLanguage.html, May 2008.
[11] MAK G., LONG J., RUBIO D., *Spring Recipes*, Apress, 2010.
[12] MDA Guide v. 1.0.1, http://www.omg.org/mda, 2003.
[13] SCHMIDT D.C., "Model Driven Engineering", IEEE Computer, vol. 39, no 2, 2006, 25–31.
[14] Visual Paradigm site, http://www.visual-paradigm.com/, 2012.

[15] GROENEWEGEN D.M., VISSER E., *Integration of data validation and user interface concerns in a DSL for web applications*, M. van den Brand, D. Gašević, J. Gray (Eds.): SLE 2009, LNCS 5969, 2010, 164–173.
[16] LI X., LIU Z., HE J., LONG Q., *Generating a Prototype from a UML Model of System Requirements*, R.K. Ghosh and H. Mohanty (Eds.): ICDCIT 2004, LNCS 3347, 2004, 255–265.
[17] CASTREJÓN J., LÓPEZ-LANDA R., LOZANO, R., *Model2Roo: A Model Driven Approach for Web Application Development based on the Eclipse Modeling Framework and Spring Roo*, Electrical Communications and Computers (CONIELECOMP), 2011, 82–87.

architecture analysis, quality scenarios,
architecture trade-off analysis method

Andrzej ZALEWSKI, Szymon KIJAS*

FEATURE-BASED ARCHITECTURE REVIEWS

The Architecture Trade-off Analysis Method and other scenario-oriented architecture assessment methods have not become common everyday industrial practice, as industrial surveys show. The high cost has been indicated as causes for this condition, together with the lack of a convincing business case and difficulties in knowledge transfer. We argue that all these factors result from the intrinsic complexities of the scenario-oriented architecture assessment paradigm, their limited scalability and from problems with integrating architecture assessment into well-established software development practices. The Feature-Based Architecture Reviews Method has been elaborated to overcome these problems. The scope of the analysis is defined by a set of architecturally-relevant software features. Each of these features is addressed with architectural decisions. These decisions, in turn, may cause risks concerning the system's quality attributes. Various techniques can support the risk assessment process, including risks connected with certain architectural patterns, architectural tactics or comparisons with reference architecture or industrial baselines. The method scales very well, as any set of software features can be assessed, and so it scales from assessing just a single feature to a fully comprehensive architecture review. The method integrates naturally with RUP or agile methodologies. A number of examples illustrate the proposed method and its advantages.

1. INTRODUCTION

Numerous architecture analysis methods have been developed since the advent of research in this field – compare [1], [3] for a survey. However, an industrial survey carried out in 2009 by Ali Babar and Gorton [4] revealed a kind of architecture analysis crisis. They found that scenario-oriented architecture analysis methods, which had dominated the research carried out until then, were still far away from becoming an everyday practice in software industry.

* Institute of Control and Computation Engineering, Warsaw University of Technology, Nowowiejska 15/19, 00-665 Warsaw, Poland.

There are numerous factors contributing to this condition [4], [5], [12]: knowledge transfer problems, a high level of effort and expense, together with the lack of a convincing business case for investment in architecture assessment, as well as the conditions of many real-world projects, all of which hinder a fully-blown architecture analysis, not to mention the lack of documentation necessary to perform the evaluation, and unstable and undocumented requirements.

In section 2, we look deeper into the reasons behind the architecture assessment crisis, and add the following factors to the above list: difficulties in the integration of architecture assessment methods with the development and project management processes, the complexity of quality scenario-oriented architecture analysis methods, and the inherent limitations resulting from the assumptions underlying these methods.

Feature-Based Architecture Reviews (FBAR), as introduced in section 3, have been designed to addresses the above issues. They integrate easily both with traditional waterfall, as well as with modern incremental, iterative development processes (such as RUP and even agile methods). This method also complies with risk management practices present in every project management methodology (e.g. PMBoK, Prince2). The central assumption of the proposed method is that architecture design is crafted in order to implement software features. The architecture decisions may cause risks, and so the architecture assessment should be aimed at identifying and assessing those risks.

The FBAR concepts have been illustrated on a number of real-world examples, and a comprehensive evaluation has been presented in section 4. The contribution of the paper is discussed against related work in section 5, while section 6 summarises the paper and presents the research outlook.

2. RELATED WORK

The shortcomings of existing scenario-oriented architecture analysis methods have already been investigated in a number of papers. The identified deficiencies include:

- High level of effort (32–70 man days, 2–6 weeks) and cost needed to perform a fully-blown architecture assessment with ATAM – compare [5];
- Mismatches with the conditions of many real-world projects, as a lack of architectural documentation, a lack of quality requirements specifications, and the instability of the quality requirements make ATAM-based assessments impossible – compare [5];
- Traceability issues – it is generally difficult to trace how business goals are reflected within the software architecture, i.e. which architecture decision addresses which business goals. As a result, the assessment of whether the architecture meets

business goals, or supports them sufficiently, is a matter of expert judgment not provided by the method itself – compare [12].

- To the above list, we would add another two issues:
- Difficulties in integrating architecture analysis methods with established development and project management methodologies;
- Intrinsic complexity of scenario-based architecture analysis methods.

The problem of integrating architecture-centric methods [10] (including architecture analysis) into development processes has been investigated in a number of papers – compare [7], [9] for integrating with agile methods, and [8] for RUP.

Architecture analysis methods can contribute to agile development by providing for better understanding of business drivers and quality requirements, enhancing communication between stakeholders, delivering a "big picture" view, improving documentation and ensuring a stable platform for agile iterations. Therefore, both approaches can co-exist [6], if project stakeholders want them to.

However, mismatches with the real-world conditions [5] will not be overcome this way. The agile methods become "less agile" when supplemented with architecture-centric methods (e.g. the architecture's description has to be retrieved somehow). The alignment between these approaches remains rather limited – compare the iteration time of agile methods and the time needed for ATAM [5]: in such conditions ATAM's can be performed at a specific milestones only, and require that agile iterations be temporarily suspended.

ATAM integrates much better with Rational Unified Process. It has been proposed to include ATAM activities in the task "Refine the Architecture" of the "Analysis and design" domain [8], performed at least once per iteration, especially during the "Elaboration" phase. At the same time, a Quality Attribute Workshop could supplement the "Requirements" discipline during the Inception and Elaboration Phase, as a requirements elicitation technique.

However, initial ATAM activities aimed at identifying and prioritising quality requirements generally overlap with the activities of the "Requirements" discipline of RUP. This may confuse project team members and undermine faith in the value added by these overlapping activities. Repeating them in the "Elaboration" phase could play a role of secondary validation of the requirements analysis, which can contribute significantly in the case of larger projects, while being perceived as redundant in the case of medium and small projects.

The requirements prioritisation technique based on cumulative voting [18] and the utility tree is just another problematic component of ATAM. It requires the engagement of all the stakeholders, which increases both the overall effort and the cost of the assessment. The voting can give rise to numerous games played between the stakeholders in order to obtain preferred requirements priorities. The method assumes that all the stakeholders act rationally and thoughtfully towards the project's success,

which might not hold in real life conditions. It is observed [18] that, in most cases, the method works well only once per project.

Quality scenarios are both complex and not fully compatible with the approaches adopted by the industry – compare, for example, the Volere template [17] for requirements specification or popular requirements management software, like RequistePro.

They effectively represent detailed requirements, but are far more ineffective when defining more general properties. Let us consider the typical requirement that the generation of reports should not interfere with transaction processing. This can be perceived both as a requirement and as a design constraint (the separation of analytical and transactional processing). However, quality scenarios are too detailed and the above property cannot be expressed directly. A concrete situation has to be specified using a kind-of workaround: Stimulus: transaction arrives; Artefact stimulated: system; Source of stimulus: user; Response: system processes the transaction in no more than 2 seconds; Environment: system is processing a management report when the transaction arrives; Response measure: maximum transaction processing time. It is certainly not a straightforward way to put across the requirement to separate OLTP from OLAP.

Expressing requirements in terms of solutions that should be included into system design is a common practice. They may come for example from the corporate baselines and best practices defined by appropriate standards (e.g. security, safety). These solutions naturally address concrete design concerns, though they are not included into software architecture as a result of a rational decision-making process starting from quality requirements and ending with the architecture decisions that ensure these requirements are met. The direct reason is that they belong to the set of best practices or corporate design baseline. Such rational decision-making must have taken place when including them into a set of best practices or a design baseline.

In this way, we come back to the issue of tracing the relations between the architecture and the business goals. There are two main factors that amplify this problem:

- Architecting is not a fully rational process driven solely by business drivers and reflected by the quality requirements. Many architecture decisions are made not because they ensure that certain quality requirements are met, but because they are well-known to the architect – they are simply the way that architect is used to resolving certain kinds of problems etc.;
- The relationships between architecture decisions and the properties of the resulting system are complex, ambiguous and, to a large extent, intangible. In most cases one can only identify and assess risks posed by certain architectural solutions. However, we are usually unable to identify concrete values of quality attributes of the resulting system at the level of software architecture only. For example, if we require that it should be possible to implement certain kinds of changes in two days, we can only assess the risk that a given feature will finally be achieved. The way up, i.e. from architecture decisions to quality attributes, and further to the business drivers, is also

complex and ambiguous. For example: including, a message broker within the architecture may or may not cause performance problems, but at the same time it improves interoperability.

Pattern-Based Assessment Reviews [5] depart from the scenario-based assessment paradigm. PBAR is a light-weight architecture assessment method in which the architecture assessment is based on the relationships between architecture patterns and quality attributes. The method is light-weight with regard to the effort, simplicity and limited prerequisites needed to start the evaluation procedure, and so the method is supposed to be more compatible with the conditions of real-world projects than ATAM and other scenario-based architecture assessment methods.

We believe that the future of architecture assessment belongs to light-weight methods, which easily and naturally integrate into existing, established development practices.

3. FEATURE-BASED ARCHITECTURE REVIEWS

The Feature-Based Architecture Reviews (FBAR) has been designed to be a scalable, light-weight architecture assessment method, suitable for a wide range of applications, i.e. from short reviews limited to just several architectural decisions, to comprehensive reviews encompassing the entire system's architecture. The method assumes that an architecture assessment will focus on risk identification and assessment. At the same time, the method was designed to foster integration with existing development and project management practices, while avoiding the difficulties of tracing business drivers and quality requirements throughout software architecture. The heart of the proposed approach is a feature-based assessment, as presented in sections 3.1–3.4, while risk identification techniques are shown in section 3.5, and integration with software development processes and software evolution are developed in sections 3.6–3.7.

3.1. ARCHITECTURALLY SIGNIFICANT FEATURES

The notion of a "feature" has been defined in IEEE 829 standard [11] as "a distinguishing characteristic of a system item (including both functional and non-functional attributes such as performance and reusability)".

With respect to software architecture, features can either be architecturally significant or insignificant. In order to achieve characteristics specified by architecturally significant feature, concrete architectural decisions have to be made. Those features that are not directly addressed by architectural decisions belong to the "architecturally insignificant" class. This will often include general non-functional requirements, such

as “the system must be able to serve 50 simultaneous users”, “the system should be available 24/7”, which are usually not directly achieved by concrete architectural decisions, but result from the overall design. We will refer to such features simply as “requirements” and keep them in a separate set for reference, if their specification has been elaborated during the development process. Non-functional requirements can belong to both “requirements” set and architecturally significant features, if such a feature was directly addressed by some architecture decisions.

Let us consider a couple of examples of architecturally significant features:

- “The product list is available on salesmen’s terminals” feature is architecturally significant as, in order to achieve the designated characteristic, it has to be decided how the product list will be made available (example options are: salesmen use internet browser to access the system via GSM VPN, or salesmen replicate the product list when synchronising with central system’s database);
- “The reports from the corporate systems are generated by the data warehouse at the user’s demand” feature requires that the data warehouse subsystem and mechanisms to upload data from the corporate systems have to be included into the system design;
- “The application authorises users via single sign-on service” – it means that there must be a single log-on subsystem providing a sign-on service within the system structure.

In the above context, concrete reports implemented to be generated from the data warehouse, as well as certain methods of navigation through the product list, belong to architecturally insignificant features.

3.2. FEATURE-BASED ARCHITECTURE REVIEWS MODES

Feature-Based Architecture Reviews can be used in two basic modes:

- Partial reviews – the review encompasses a subset of software’s features;
- Comprehensive reviews – the review embraces all software’s features.

Partial reviews can be performed whenever there is a need to assess architecture implementing some set of features. Comprehensive reviews are supposed to be performed at major mile-stones of a project, when entire or substantial parts for software architecture are ready.

3.3. FEATURE-BASED ASSESSMENT SCHEME

Decisions made to implement certain features may cause risks affecting the system’s quality attributes – for example:

- a choice to access the central system via GSM VPN causes a risk to the system’s availability, which may result, for example, from the break-down of connection to the

central system (e.g. the GSM signal may be too weak in underground locations like cellars causing connection cut-off),

- the choice to replicate the product list on a salesman's terminal poses a risk to system availability and buildability (synchronisation solutions are complex, error-prone and have to be carefully designed and/or based on a proven solution),
- the data warehouse can be a single point of failure – risks concerning availability and performance should be properly managed.

The main purpose of architecture assessment is to identify risks posed by the architectural design. Therefore, identification should follow a causal chain: architecturally significant feature – architecture decisions (addressing that feature) – identified risks (posed by architectural decisions) affected quality attributes – assessed risks (in terms of probability and impact).

3.4. REVIEWING ARCHITECTURE ON THE BASIS OF FEATURES

The assessment process presented in Fig. 1 starts from eliciting architecturally significant features from the entire set of software features. This set of features delimits the scope of the review. Then the analysis follows the chain described in section 3.3: for each elicited architecturally significant feature, relevant architectural decisions are identified, and then the risks caused by each of these decisions are identified and assessed. Identified risks, along with their preliminary assessed priorities, should subsequently be fed into risk management procedures, or provide feedback for the developers or architects.

All the information gathered during the review process can be stored using the data model of Fig. 2.

It is worth noting that "Quality Requirements Register" represents Quality Requirements Specification, if such was developed. It enables to refer identified risks to specific quality requirements. In the case of comprehensive assessment it may help to assess, how well the analysis covers identified quality requirements. The ratio of quality requirements affected by risks to the overall number of quality requirements can be used as coverage metric.

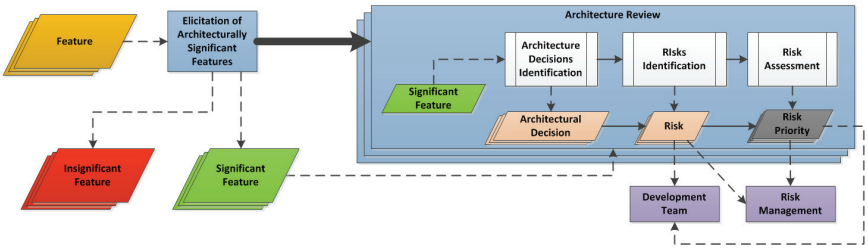


Fig. 1. The Feature-Based Architecture Review Process

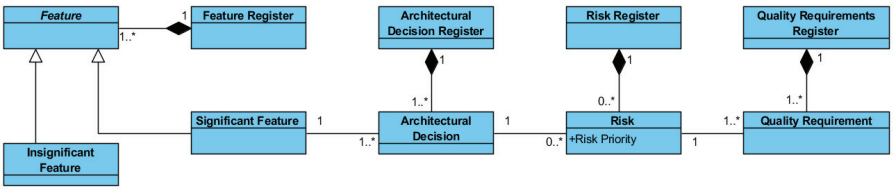


Fig. 2. The Data Model Supporting Feature-Based Architecture Reviews

3.5. RISK IDENTIFICATION AND ASSESSMENT

Identifying the risks posed by certain architectural decisions is at the core of any architecture assessment. However, the success of architecture analysis still relies heavily on the knowledge and experience of the surveyors, because there is no general method addressing this issue, only a number of techniques supporting risk identification:

- risk themes discovered during ATAM evaluations – presented in [13];
- exploiting the relationship between architectural patterns and their influence on the quality requirements – compare [5];
- a comparative analysis of the developed architecture against industrial baselines (e.g. reference architectures), applicable standards (e.g. ISO 17799 for security), architectural tactics [2], and design rules.

Identified risks should be subsequently assessed in terms of the probability of occurrence and impact, which will become a basis for prioritising them according to the well-known equation: risk importance = probability of occurrence x impact. Taking into account the approximate and expert nature of the assessment, we suggest using a three-grade scale for both impact and probability – small, medium and large:

- small (S) – only selected features may be affected by the risk (impact); there is rather small probability that certain risk will occur (probability);
- medium (M) – impact and probability, which can neither be classified as small nor as large;
- large (L) – all or a large number of features may be affected by the risk (impact); it is very likely that certain risk will really happen (probability).

The resulting risk importance assessment can be treated as preliminary. It is supposed to be reviewed by risk management procedures (if any are in place). Hence, the preliminary assessment provided by FBAR could be adjusted with regard to business priorities. The decision how to cope with the identified risks can be done at the project management level.

3.6. INTEGRATING FEATURE-BASED REVIEWS WITH DEVELOPMENT PROCESSES

Feature-Based Architecture Reviews can be performed on a feature-by-feature, stage-by-stage basis, or as a comprehensive architecture review, depending on how many features the review embraces. Architectural decisions can represent either a chosen design option, or the alternatives being considered as a solution to implementing a certain feature.

This facilitates the integration of FBAR with various development processes. In general, Feature-Based Architecture Reviews can be used both for a comprehensive audit-like review, and as a component of architects' services, [14] providing a kind of architectural assistance to the developers.

FBAR and Agile Development Processes

Agile software development methods are iterative and incremental processes. The software is developed gradually, increment-by-increment. Each increment encompasses a subset of features. The priority of implementing the features is discussed and agreed with the client in order to reflect the business priorities and to maximise his value. The features are defined by user stories.

This makes the integration of a feature-based assessment with agile methods quite straightforward:

- In the case of Scrum: FBAR could be performed during the sprints (iterations), either as a comprehensive assessment (Fig. 3) of the design of all the features (user stories), or as a kind of architectural assistance service for selected, architecturally significant features;
- In the case of Feature-Driven Development: FBAR could be performed between the "Design by Features" and "Build-by-Features" phases of each iteration – Fig. 4 – or as an architectural assistance service as described above for Scrum;
- In the case of XP: FBAR could be integrated into XP in the same way as in the case of Scrum.

FBAR and the Rational Unified Process

The Rational Unified Process is an iterative development process. Unlike agile methods, it assumes thorough planning based on an intensive requirements analysis at early phases of the project (Inception, Elaboration). However, the process realistically allows for new requirements to emerge during the entire course of the project, hence the activities of Requirements and Analysis and Design disciplines can take place during the entire project lifecycle. This needs to be accounted for by the integration of FBAR into RUP.

The integration of the FBAR method into RUP has been achieved with the following activities:

a) “Selection of Architecturally Significant Features” (in the “Requirements” discipline);

b) “Architecture Review” Activity (in the Analysis & Design discipline).

Activity “a” should be performed at the end of the Inception and Elaboration phases. It may optionally be repeated if new requirements emerge later during the project.

Activity “b” should be performed during all RUP phases – especially at the end of the Inception and Elaboration phases, as well as during all the iterations of the Construction phase and even during Transition phases.

3.7. FEATURE-BASED ARCHITECTURE REVIEWS AND EVOLUTION

Software evolution can easily be expressed in terms of software features being added, removed or modified. The proposed method can be used both for an assessment of the evolution step’s increments on a feature-by-feature basis, or for a comprehensive assessment of software architecture resulting from the evolution step.

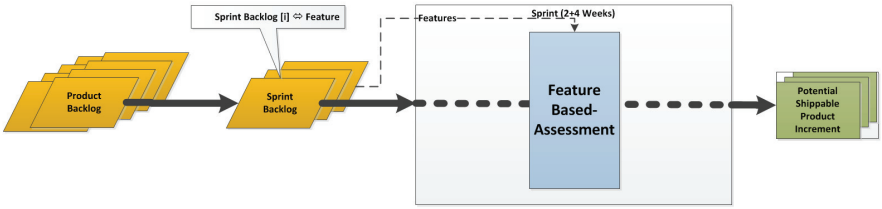


Fig. 3. Integration of Feature-Based Reviews with Scrum

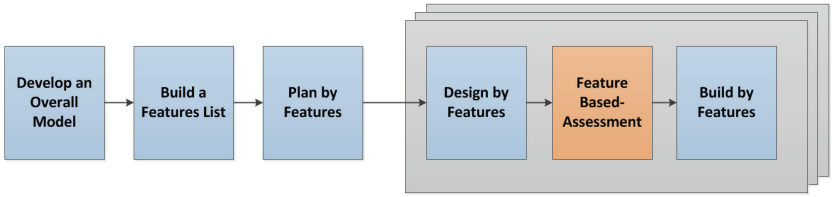


Fig. 4. Integration of Feature-Based Reviews with Feature-Driven Design

4. CASE STUDY

The concepts of Feature-Based Architecture Reviews will be illustrated with an example of a real world system used in the banking sector. The system supports the exchange of various kinds of information and documents (claims, direct debits, information concerning accounts moved from one bank to another, etc.) between banks and other institutions (e.g. bailiff offices, social security agencies). Additionally, it pro-

vides a fail-over communications channel in the event of a failure of the main clearing system. The system generally follows the service-oriented architecture scheme providing both www and web service interfaces to its functionality. Table 1 presents a sample of data gathered during the FBAR assessment: identified architecturally significant features, architectural decisions and risks.

Table. 1. Feature-Based Architecture Reviews – case study

Feature	Architectural decision	Risk	Quality attributes	Priority	
				Impact	Probability
System shall be accessed via secured communication channels only	Encrypted internet communication transmission with SSL using 128-bit keys	1. Slowdown of data transmission	Performance Availability Reliability	M L M	M S S
	Authentication with client's certificate	2. Problems with transport of client's certificate through a proxy servers	Availability Reliability	L M	M M
		3. Problems with getting access to the website using client's certificate	Usability Availability	M M	M M
Access to the system by website	3-tier web application	4. System performance degradation	Performance Availability	L M	M S
Access to the system by webservice	3-tier application	5. System performance degradation	Performance Availability	M L	M S
Scalable data repository	Relational database running on a server cluster	6. Inefficient and error prone administration of the server	Scalability Performance Availability Reliability	L M L M	S S S S
	Database in the same data centre as the business logic layer of the system	7. Breakdown or disaster affecting the data centre making the whole system not available	Availability	L	S
Daily update of client's application dictionaries	Update of dictionaries via webservice	8. Connection lost while downloading updated dictionary	Reliability	M	M
Enable of quick addition of new features	The system has been split into following independent modules using a common database: claims, direct debits, informa-	9. Failure of the common database (repository) causing the downtime of the whole system	Availability	M	S

	tion concerning accounts moved from one bank to another, requests for debtors accounts and balances monitoring				
User management shared with other corporate systems	System shall contain user management module to be used by all the corporate systems	10. User management module failure will result in the unavailability of the whole systems and other company's systems	Availability	L	S
		11. User configuration becomes too complicated and time consuming	Usability	S	M

The above analysis has been explained in more detail below (applied risk identification technique has been indicated in brackets):

Ad 1. Data encryption necessary to secure communication channel may slow down the transmission. (evaluator's knowledge)

Ad 2. In many cases (especially in large companies) firewalls and proxy servers exists on system borders. Transmission of the client certificate to the server via proxy servers is often a source of problems. (evaluator's knowledge)

Ad 3. Installation of client's certificate, root certificates and chip card reader is known as a problem for many users. (evaluator's experience)

Ad 4 and 5. Multi-tier architecture follows a layers architectural pattern, which has a negative influence on system performance. (mismatch between architecture pattern and quality attributes)

Ad 6. Administration of database in a cluster could be difficult for a novice or inexperienced administrator. (evaluator's knowledge)

Ad 7. Breakdown or disaster affecting the data center would make the whole system unavailable. (single point of failure – a risk theme)

Ad 8. Overloaded internet links may cause a connection loss while downloading the dictionaries. This can make system not available, as current dictionaries are necessary for client's software to be used. (evaluator's knowledge)

Ad 9. Failure of the common database will cause failure of all the modules. (single point of failure – a risk theme)

Ad 10. This was supposed to provide centralized access control mechanism for all the systems. However, this makes the operation of all the systems depending on single sign-on service provided by the evaluated system. (single point of failure – a risk theme)

Ad 11. Configuration of users and users' permissions for many system's at once may become very complicated making such a system difficult to use. (evaluator's knowledge)

It is also notable that most of the above risks have been identified on the basis of expert's knowledge and by referring certain solution to known risk themes. The assessment requires not only knowledge on architectural patterns but also a lot of context information specific to the evaluated system and its owner's organization.

5. CONTRIBUTION AND DISCUSSION

The Feature-Based Assessment Reviews method is the main contribution of this paper. The main advantages of the method are:

- Easy integration with existing development processes, including agile and non-agile ones. The method interfaces with project management methodologies – its outcomes should be fed into the risk management process;
- Scalability – the method scales well – from the assessment of an architecture implementing just a single feature, to an analysis of an entire system's architecture; it can be both a light-weight architecture assessment method used by small development teams, as well as a fully-blown architectural analysis. Therefore, the method can be used both to assist architectural decision-making while developing software architecture, or for a comprehensive assessment of already developed architecture or its substantial parts, as well as to assist software evolution;
- Limited prerequisites needed to start the assessment – no architecture documentation is needed to start the assessment, although a person skilled in software architecture is needed to perform the assessment;
- Limited amount of information needed and gathered during the architecture assessment – architectural decisions comprising architectural design have to be retrieved from the development team or from the architectural documentation (if available) or from both. Not all the information contained in the quality scenarios is necessary for the analysis;
- Simple assessment process – the logical sequence of the assessment process seems to be intuitive – software design is done in order to achieve certain features. These features may concern functional and non-functional requirements. However, architectural decisions made to implement a certain feature may pose risks to the non-functional properties of the system. These risks, after the assessment, are the final result of the analysis;
- Completeness of the analysis – completeness is ensured by enforcing that all the features subject to the assessment have been reviewed.

The two last issues deserve a more in-depth discussion. Feature-Based Architecture Reviews assume quite a different logic of the assessment process than ATAM or even Pattern-Based Assessment Reviews [5], logic of both has been presented and compared in Fig. 5.

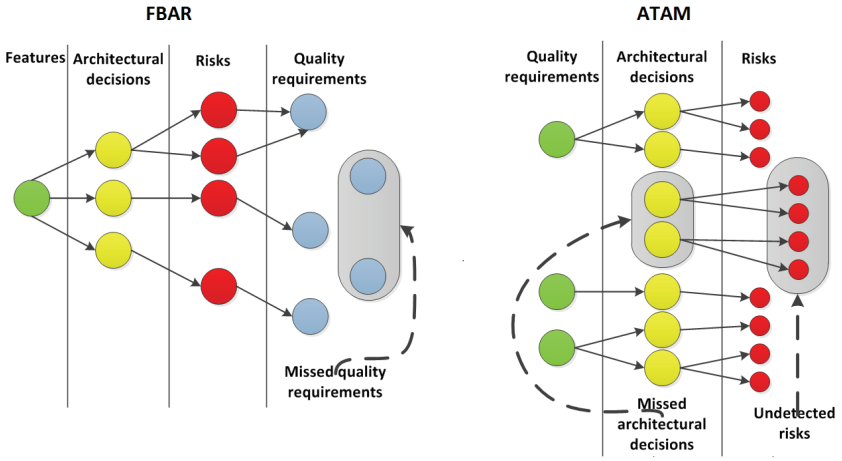


Fig. 5. Comparison: assessment logic of FBAR and ATAM

Traditionally, quality requirements (in ATAM expressed as quality scenarios, in PBAR – in any way, even informally) are the starting point of the review, then architecture decisions (e.g. in case of PBAR – identified patterns) affecting these requirements are retrieved from the software architecture. Next, the risks concerning the quality requirements identified at the beginning of the assessment are identified.

This ensures that all the identified quality requirements are reviewed during the architecture analysis. Hence, the completeness of the analysis depends on the requirements elicitation. Poor requirements analysis can possibly leave important parts of the architecture "untouched" and risks posed by these omitted architecture decisions undetected.

The comprehensive option of FBAR does not guarantee that all the specified quality requirements have been assessed, but it does ensure that all the architectural solutions (decisions) addressing software features have been reviewed. In any case, the method works well without any requirements specification – in such case, risks to certain quality attributes are identified. However, if a requirements specification is available, then the percentage of the specified requirements covered by the assessment can be calculated. The question remains, what can be concluded about the requirements not covered by the analysis? It can indicate: a poor analysis (probably missed architecturally relevant features), requirements that are not architecturally relevant, or quality requirements that have simply not been addressed so far – typical for an incremental assessment.

Tracing the dependencies between features, decisions, risks and quality attributes is straightforward in FBAR. If features and quality requirements were linked appropriately to business goals, it would be also possible to assess how effectively business goals are supported.

The fundamental issue with regard to the completeness of the analysis is whether architecture is exclusively defined by the superposition of the decisions implementing software features, or whether it is more than that. We would argue that the first option holds, where a complete set of features is concerned. 'Complete' means here that all the specific properties of the software, i.e. functional and non-functional, have been defined. Architecture decisions that do not address any feature are redundant.

Pattern-Based Architecture Reviews seems to have become the main reference point for light-weight architecture analysis methods, departing from a scenario-oriented assessment paradigm. Both PBAR and FBAR address similar issues: enable incremental assessments and offer a simplified assessment process, thereby saving on cost and effort typical for traditional, ATAM-based reviews. Neither method assumes any documentation as input to the assessment, while both accept face-to-face communication and offer quite a simple assessment process. PBAR have been intended for an assessment of sufficient parts of software architecture, starting from the "walking skeleton". FBAR scales well from assessing a single architectural decision up to a comprehensive architecture analysis. The relationships between architecture patterns and quality attributes enable the discovery of mismatches between architecture and quality requirements in the case of PBAR. The same approach can be used with FBAR as one of the risk identification techniques supporting architecture assessment. However, we would argue that relying solely on patterns is in many cases insufficient, as many features are, in practice, addressed by a more complex architectural solution encompassing a number of architectural patterns all "working together", or structures that cannot be directly related to patterns. The example of section 4 shows that architectural patterns are just one of a number of techniques that can facilitate the identification of architectural risks.

6. SUMMARY. FURTHER RESEARCH OUTLOOK

Feature-Based Architecture Reviews resolve many problems limiting the application of architecture assessment in industrial practice: the challenge of tracing the dependencies between the quality requirements and software architecture, integration with existing software development processes, the scalability and comprehensibility of the method.

Further developments should include:

- Enhancing the assessment model by enabling feature-to-business-goals assignment (similar to the approach of the service-oriented analysis of SOMA [16]);
- Integrating FBAR with architecture modelling frameworks;
- Enhancing FBAR with sensitivity and trade-off points identification;
- Developing software supporting FBAR assessment.

ACKNOWLEGEMENT

This work was sponsored by the Polish Ministry of Science and Higher Education under grant number 5321/B/T02/2010/39.

REFERENCES

[1] DOBRICA L., NIEMELÄ E., *A Survey on Software Architecture Analysis Methods*, IEEE Transactions on Software Engineering, Vol. 28, Iss. 7, 2002, 638–653.
[2] SUNTAE KIM, DAE-KYOO KIM, LUNJIN LU, SOOYONG PARK, *Quality-driven architecture development using architectural tactics*, Journal of Systems and Software, Volume 82, Issue 8, August 2009, 1211–1231.
[3] KOZIOLEK H., *Sustainability Evaluation of Software Architectures: A Systematic Review*, QoSA-ISARCS '11, ACM SIGSOFT, 2011.
[4] ALI BABAR, M., GORTON, I., *Software Architecture Review: The State of Practice*, IEEE Computer, Vol. 42, Iss. 7, 2009, 26–32.
[5] HARRISON N., AVGERIOU P., *Pattern-Based Architecture Reviews*, IEEE Software, Vol. 28, Iss. 6, 2011, 66–71.
[6] ABRAHAMSSON, P., BABAR, M.A., KRUCHTEN, P., *Agility and Architecture: Can They Coexist?,* IEEE Software, Vol. 27, Iss. 2, 2010, 16–22.
[7] NORD R.L., TOMAYKO J.E., *Software architecture-centric methods and agile development*, IEEE Software, Vol. 23, Iss. 2, 2006, 47–53.
[8] KAZMAN R., KRUCHTEN P., NORD R., TOMAYKO J. E., *Integrating Software-Architecture-Centric Methods into the Rational Unified Process*, SEI, Technical Report CMU/SEI-2004-TR-011, July 2004.
[9] NORD R., TOMAYKO J. E., WOJCIK R., *Integrating Software-Architecture-Centric Methods into Extreme Programming (XP)*, SEI, Technical Note CMU/SEI-2004-TN-036, September 2004.
[10] HOFMEISTER C., KRUCHTEN P., NORD R. L., OBBINK H., RAN A., AMERICA P., *A general model of software architecture design derived from five industrial approaches*, Journal of Systems and Software, Vol. 80, Iss. 1, Jan. 2007, 106–126.
[11] IEEE Standard for Software and System Test Documentation, *IEEE Std 829™-2008(Revision of IEEE Std 829-1998)*, IEEE Computer Society, 18 July 2008 Revision of IEEE Std 829-1998, E-ISBN. 978-0-7381-5746-7, ISBN: 978-0-7381-5747-4
[12] KAZMAN R., BASS L., KLEIN M., LATTANZE T., NORTHROP L., *A Basis for Analyzing Software Architecture Analysis Methods*, Software Quality Journal, Vol. 13, Iss. 4, 329–355.
[13] BASS L., NORD R., WOOD W. G., ZUBROW D., *Risk Themes Discovered Through Architecture Evaluations*, SEI, Technical Report CMU/SEI-2006-TR-012, September 2006.
[14] FABER R., *Architects as Service Providers*, IEEE Software, Vol. 27, Iss. 2, 2010, 33–40.
[15] HARRISON N. B., AVGERIOU P., *Leveraging Architecture Patterns to Satisfy Quality Attributes*, Lecture Notes in Computer Science, Vol. 4758, Software Architecture, 2007, 263–270.
[16] ARSANJANI A., GHOSH S., ALLAM A., ABDOLLAH T., GANAPATHY S., HOLLEY K., *SOMA: A method for developing service-oriented solutions*. IBM Systems Journal, Vol. 47, No. 3, 377–396.
[17] ROBERTSON S., ROBERTSON J., *Mastering the Requirements Process*. Addison-Wesley Professional, 2 edition, Mar. 17, 2006, ISBN 0321419499
[18] LEFFINGWELL D. AND D. WIDRIG, *Managing software requirements: A Use Case Approach*, 2nd ed, Addison-Wesley, Boston, 2003.

natural language processing, text correction,
morphological analysis, link grammar

Dariusz BANASIAK*, Jarosław MIERZWA*, Antoni STERNA*

AUTOMATIC CORRECTION OF ERRORS IN POLISH TEXTS

The paper presents an approach to detection and correction of errors in computerized edition of texts in Polish. Some errors in texts may result from weak language competence but mostly they are caused by erroneous keystrokes. Contemporary text editors are equipped with modules performing detection and correction of errors. However, they allow to eliminate only some types of errors. Correction methods used in editors are mostly dictionary based, word context is usually not considered. In Polish texts there is special class of errors, relatively difficult to correct. These errors result from the presence of diacritical marks in some letters which are typed by combined keystrokes. Inflection related errors may produce word forms present in dictionary but incorrect in given context. To detect and correct such errors word dependencies should be considered. Modified Link Grammar equipped with inflection related linking requirements is proposed. The process of error correction and detection consists of three stages. First, erroneous word is identified and then possible correction candidate words are generated. To limit the number of correction alternatives some methods based on word statistics or technical cause of error may be used. In last stage word dependencies are used to select the word best matched in given context. Proposed method may be used as supplement in existing text editors. It may be also used for preliminary test analysis in automated text processing systems (e.g. information extraction systems).

1. INTRODUCTION

Today almost all texts used in communication are stored and transferred in digital form. First, however, all texts must be created and typed in and this involves human activity which is always susceptible to errors at different levels: lexical, syntactic, semantic and contextual. Errors in texts may be classified according to their origin as typographical errors or errors caused by weak language competence. Typographical

* Institute of Computer Engineering, Control and Robotics, Faculty of Electronics, Wroclaw University of Technology, Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland.

errors present still growing problem because miniature keyboards or keypads, with densely packed keys, are extensively used on mobile devices such as notebooks, intelligent phones or personal digital assistants. Typical keyboard relative errors are accidental key repetitions, substitutions of letters on the keyboard directly adjacent to the intended key and unintentional key insertions.

Two different error detection and correction strategies may be distinguished. In first strategy, correction is performed on-line (during typing), forward information is inaccessible. In second strategy syntactic information may be used, therefore only complete sentences may be analyzed.

In Polish texts diacritical marks are significant cause of errors because some letters (e.g. ą, ę, ć) require key combination. This results in common letter substitution like ą-a, ę-e and so on (usually diacritical mark is omitted, redundant marks are rather infrequent). In some texts, especially in quick communication via internet or mobile phone, diacritical marks may by omitted intentionally. If such texts are introduced into public domain they should be corrected, preferably in an automated way.

There are two main problems in error correction. First, the list of correction candidates may be incomplete if only most probable errors are admitted or too large if all possible errors are considered. Second, reliable word selection may not be possible if word dependencies are not considered. This is apparent in Polish due to rich inflection.

2. ERROR DETECTION

In order to correct errors, first they should be detected. It is relatively simple if the correct word is distorted into the form that cannot be identified by straightforward dictionary look-up. Detection is more difficult if typing error leads to another correct word. Usually misspelled word contains the following errors:

– deletion (character missing),
– insertion (an extra character),
– substitution (wrong character),
– transposition (two adjacent characters interchanged).

In Polish texts letters with diacritical marks are obtained by key combination (Alt + actual key). Consequently there is another class of frequent errors resulting from the following problems with Alt key:

–not pressed (diacritical mark missing),
–pressed unintentionally (redundant diacritical mark),
–pressed too early or too late (diacritical mark at wrong letter),
–pressed too long (diacritical mark duplicated at adjoining letters),
–pressed too short (missing diacritical mark at one of adjoining letters),
–pressed along with wrong key (wrong letter with diacritical mark).

To illustrate the problem of diacritics the result of fast typing test will be quoted here [1]. Eight participants of the test were given the task of typing out the text containing about 180 words, they were encouraged to type as fast as possible. Test results, presented in Table 1, although not statistically significant, clearly show predominance of errors caused by missing diacritical mark. Deletion and insertion are almost equally probable.

Table 1. Results of fast typing test

Error type	Number of occurrences
diacritical mark missing	35
deletion	12
insertion	11
redundant diacritical mark	9
orthographical error	8
substitution	6

After detection of erroneous word the list of correction candidates should be generated. For example, if missing letter is considered, all possible letters should be inserted at all possible positions and generated word should be looked for in dictionary (it is assumed, that dictionary contains all inflectional forms). This action should be repeated for all supposed types of errors such as letter insertion, substitution and so on. Although the process is simple with regard to the principle it may produce excessive number of correction candidates.

In texts from a very restricted domain, vocabulary and word collocations are usually very limited, therefore the list of correction candidate words may be significantly reduced. Many successful researches were done in the field, they are usually based on statistical model of language. For example, the concept of N-grams may be used to predict next word [2].

For general domain texts some statistical measures for words may be used only as a preliminary measure to limit the number of correction alternatives. To make final decision about correction it may be necessary to analyze word dependencies. It is especially important in Polish, due to rich inflection and free word order. It is common case that after distortion the word changes its grammatical form or turns into another correct word. Sometimes the problem may be straightened out only by some analysis of word dependencies.

Correction candidate words may be ordered according to probability of errors resulting from physical aspects of device used in typing (keyboard):

– diacritical mark errors are frequent due to multiple keystroke,

– substitution errors are more probable for adjoining letters (keyboard layout).

Another statistical measures, not keyboard related, may also be taken into account, for instance frequency of appearance of given word in text corpus or even in currently processed text. Last measure may be particularly justified for ”rare words” (the words with low statistics in general corpus which already appeared several times in currently processed text).

3. MODEL OF TEXT ANALYSIS IN POLISH

Automatic analysis of texts in Polish is quite complicated due to inherent properties of language such as complex inflection, flexible word order, discontinuous phrases, lexical ambiguity and so on. Let us illustrate the problem by the following sentence:

Piotr dał Marii ciekawą książkę.
(*Peter gave Mary an interesting book.*) (1)

The following elements may be distinguished in sentence (1): subject (*Piotr*), predicate (*dał*), direct object (*ciekawą książkę*) and indirect object (*Marii*). Functions played in sentence by separated phrases may be determined on the basis of inflectional properties of individual words. Noun in nominative plays the role of subject, personal form of verb acts as predicate, direct object is represented by noun in accusative (linked with adjective in this case) and noun in dative constitutes indirect object.

It should be also noted that some words in sentence have influence on the form of other words (in the phrase *ciekawą książkę* gender, number and case are consistent). This feature of language is known as accommodation and consists in adaptation of one syntactic unit (accommodated unit) to the requirements of another unit (accommodating unit). There are three types of accommodation: morphological, lexical and syntactic [3]. Let us consider the following two sentences:

Piotr przeczytał ciekawą książkę.
(*Peter read an interesting book.*) (2)

Ewa przeczytała długi list.
(*Eve read a long letter.*) (3)

Presented sentences exemplify morphological accommodation. Accommodating unit imposes specific values of inflectional category on accommodated partner. Typical example of morphological accommodation is adaptation of predicate to subject with respect to gender, number and person (*Piotr przeczytał ...* but *Ewa przeczytała ...*). This type of accommodation may be also observed between noun and adjective. Accommodating unit (noun) imposes number, gender and case on accommodated adjective. In sentence (2) *ciekawą* must appear as feminine (in order to match

the noun *książka*) whereas in sentence (3) adjective *długi* is masculine because it should conform to noun *list*.

In proposed approach the phenomenon of accommodation constitutes essential factor allowing detection and subsequent correction of errors in texts. It is assumed that errors leading to the change of class of word (e.g. from verb into noun) or word form (case change for nouns) may be detected through word conformity test within the sentence.

3.1. MORFEUSZ – MORPHOLOGICAL ANALYZER

Morfeusz analyzer performs morphological analysis for Polish. It uses data from Grammatical Dictionary of Polish (SGJP) which contains full grammatical description for about 245 000 Polish words. For input word, given in any inflectional form, this analyzer produces all alternative interpretations containing basic form of word associated with tags describing values of grammatical categories of particular forms. This is illustrated in Table 2 which presents Morfeusz output for sentence (1):

Table 2. Morfeusz output for sentence (1)

Piotr	piotr	subst:sg:nom:m1
dał	dać	praet:sg:m1.m2.m3:perf
Marii	maria	subst:sg:gen.dat.loc:f subst:pl:gen:f
ciekawą	ciekawy	adj:sg:acc.inst:f:pos
książkę	książka	subst:sg:acc:f

The tags shown in third column of Table 2 are positional. First position defines part of speech, the following items represent values of grammatical categories. For instance, in first row tag *subst* denotes noun, and is followed by specific values of number (*sg – singular*), case (*nom – nominative*) and gender (*m1 – masculine personal*). Detailed information on conventions used in tagging can be found in [4].

3.2. MODIFIED LINK GRAMMAR

Link Grammar [5] belongs to the class of dependency grammars. It is based on planarity, phenomenon common in many natural languages. Relations between individual words are represented by arcs with labels describing the type of dependency between words. Each arc may be divided into two half-arcs called connectors. Each connector has its owner (the word from which the connector originates), direction (left or right) and the name called label. The set of connectors, defined for each word (or class of words [6]), characterizes the ability of word (or class of words) to link with

other words in sentence. During analysis of the sentence the connectors originating from two different words may be joined and thus create link if their labels are compatible and directions opposite. In its original form Link Grammar is not very useful in analysis of inflected languages (including Polish).

As was mentioned earlier, the phenomenon of accommodation plays important role in Polish. Therefore, during analysis of sentences it is necessary to consider grammatical categories (number, case, gender, person etc.). To effectively use connectors they should be supplemented with information about inflectional properties of linked words. Table 3 presents additional linking requirements for selected types of connectors.

Table 3. Inflectional requirements for selected types of connectors

Connector label	Linked words	Additional linking requirements
A	adj – noun	$n_1 = n_2$; $c_1 = c_2$; $g_1 = g_2$
S	verb – noun	$n_1 = n_2$; $p_1 = p_2$; $g_1 = g_2$; c_2= NOM
O_G	verb – noun	c_2 = GEN
O_D	verb – noun	c_2 = DAT
O_A	verb – noun	c_2 = ACC
MVp(prep)	verb – prep	
J_A	prep – noun	c_2 = ACC
J_I	prep – noun	c_2 = INST

Type A connector denotes connection between adjective and noun. Linked words must be compatible with respect to number ($n_1 = n_2$), case ($c_1 = c_2$) and gender ($g_1 = g_2$). Connector of type S is used to link personal form of verb (sentence predicate) to noun playing the role of subject. Linked words must be compatible with regard to the number ($n_1 = n_2$), person ($p_1 = p_2$) and gender ($g_1 = g_2$). Additionally, it is required that the case of noun should be nominative (c_2= NOM). Type O connector denotes link between verb and its object (noun). Since different verbs require objects in different cases, subscript at connector label specifies required case of noun (G – genitive, D – dative, A – accusative and so on). Connector of type MVp represents link between verb and prepositional phrase (playing the role of object or adverbial). Optional information in parenthesis defines preposition which should be connected with specific verb. Type J connector characterizes dependency between preposition and noun. Since different prepositions may require different noun cases, label subscript specifies expected case (A – accusative, I – instrumental etc.).

Due to free word order, characteristic of Polish, linked words may appear in sentence in different order (direction of connector is not significant). However, in some cases the word order is essential (e.g. in prepositional phrases noun is always preceded by preposition). Therefore, three types of connectors are introduced: right-handed (e.g. J_A+ attached to preposition), left-handed (e.g. J_A – attached to noun) and undirected (e.g. A which may be attached to noun as well as to adjective).

According to original Link Grammar in definitions of linking requirements both connector labels and logical operators may be used. The following logical formula (fragment of linking requirements for the verb *czytać*):

$$\text{S and } (O_A \text{ or } MVp(„o”)) \tag{4}$$

means that the verb *czytać* may be connected to noun in nominative (connector S) and noun in accusative (connector O_A) or prepositional phrase containing specific preposition „o” (connector MVp(„o”)).

It is assumed that linking requirements may be defined both for specific words and word classes.

4. APPLICATION OF MODEL TO TEXT CORRECTION

In the process of error detection and subsequent correction two kinds of information are essential. The first is information on inflectional properties of words (obtained from Morfeusz analyzer) and the second are linking requirements (defined as appropriate combination of connectors). Let us examine the following example sentence:

Mój dobry kolega ma pięknego psa.
(*My good friend has beautiful dog.*) (5)

The result of morphological analysis of sentence (5), obtained from Morfeusz analyzer, is presented in Table 4.

Table 4. Morfeusz output for sentence (5)

Mój	mój	adj:sg:nom:m1.m2.m3:pos adj:sg:acc:m3:pos
dobry	dobry	adj:sg:nom:m1.m2.m3:pos adj:sg:acc:m3:pos
kolega	kolega	subst:sg:nom:m1
ma	mieć mój	fin:sg:ter:imperf adj:sg:nom:f:pos
pięknego	piękny	adj:sg:gen:m1.m2.m3.n1.n2:pos adj:sg:acc:m1.m2:pos
psa	pies	subst:sg:gen.acc:m2

Table 5 shows linking requirements for all words contained in sentence (5). As was pointed before, description of acceptable connections may be defined at the level of specific words in basic form (such case is represented by verb in this example) or word classes (applied to nouns and adjectives in this sentence).

Table 5. Linking requirements for words in sentence (5)

Mój	mój	A
dobry	dobry	A
kolega	kolega	{@A} and (S or O or J)
ma	mieć mój	verb: S and O_A adj: A
pięknego	piękny	A
psa	pies	{@A} and (S or O or J)

Some symbols used in notation for nouns (*kolega* and *pies*) need an explanation. Curly brackets { } show that given link is optional (nouns may be linked to adjectives but this is not obligatory). Symbol @ denotes so called multilink (the connector may be connected to more than one connector of the same type).

Analysis of the sentence correctness consists in verification of linking requirements for all words in the sentence. The check order corresponds to words order in the sentence. Let us consider the word *mój*. It has only one linking requirement denoted by label A (there are some simplifications in this example; certainly, adjectives may be also linked to other words, for example adverbs). All remaining words in this sentence have connectors with label A (in theory they might be connected to word *mój*). However, definition of A type connector implies connection between adjective and noun. Since word *mój* is an adjective, the other word must be noun (*kolega* or *psa* are possible candidates). Additionally, adjective and noun must agree with regard to number, case and gender (see Table 3). The word nearest to word *mój* will be checked first. Table 6 presents comparison of inflectional properties for words *mój* and *kolega*.

Table 6. Comparison of inflectional properties for words *mój* and *kolega*

mój	mój	adj:**sg**:**nom**:**m1**.m2.m3:pos adj:sg:acc:m3:pos
kolega	kolega	subst:**sg**:**nom**:**m1**

For both words required inflectional categories are compatible (number: singular, case: nominative, gender: masculine personal). Comparison of words *dobry* and *kolega* is carried out in similar way.

In the next step linking requirements of word *kolega* are checked. Optional connection with connector A was already considered. In order to fulfill remaining requirements, it is necessary to verify if the word *kolega* may be linked to other words with connector of the type S, O or J. There is only one possible connection, via connector S, to verb *ma*. Comparison of inflectional properties for words *kolega* and *ma* is shown in Table 7.

Table 7. Comparison of inflectional properties for words *kolega* and *ma*

kolega	kolega	subst:**sg**:**nom**:m1
ma	mieć	fin:**sg**:**ter**:imperf
	mój	adj:sg:nom:f:pos

From Table 7 it follows that the verb is consistent with noun with respect to number (singular), person (third, all nouns connect to verb in third person). The case of noun (nominative) is also correct. It does not follow from Table 7 that gender is matched. It should be observed, however, that the form of verb in third person, in present tense is the same, regardless of gender.

After similar analysis for remaining words it may be concluded that there is a link of type O_A between words *ma* and *psa* and link of type A joins words *pięknego* and *psa*. The final result of analysis for sentence (5) is presented in Figure 1.

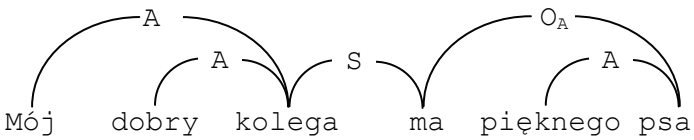


Fig. 1. The result of analysis for sentence (5)

Let us assume that as a result of error (missing Alt key) sentence (2) was entered in the following form:

Moj *dobry kolega ma pięknego psa.* (6)

The sequence of characters *moj* does not represent any correct word. On the basis of described earlier causes of errors (deletion, substitution etc.) it is possible to generate replacement candidates for *moj*. We obtain the following words: *mój*, *moje*, *moja*, *maj*, *moc*, *myj* (presumably the list is not complete). To determine correct word it is necessary to verify if the word matches remaining words in the sentence. In this way the following words may be excluded: *moje* (wrong number), *moja* (wrong gender), *myj* (two verbs in sentence, linking requirements cannot be fulfilled), *maj* and *moc* (there would be two subjects in the sentence). All linking requirements can be fulfilled only for word *mój*.

Obviously, sometimes the process of correction may be more complex, the case of more than one error may be an example. Many combinations of correction candidates may result in several potentially correct solutions. The following criteria may be used in order to arrive at final conclusion:

– statistical frequency of errors of given type (see Table 1),
– frequency of words in text corpora,

– probability of specific structure of sentence (determined by analysis of available texts corpora),
– analysis of word surroundings (N-grams, Markov models etc.),
– analysis of semantic dependencies (it may be difficult, considering state of the art in Natural Language Processing).

In doubtful cases final decision should be made by human supervisor.

5. CONCLUSION

Error detection and correction is complex task, especially for Polish texts due to diacritical marks, free word order and complex inflection. Contemporary text editors offer fairly effective, dictionary based, error detection but decision on correction is usually left to the user. Presented approach is based on the assumption that effective, unsupervised error correction cannot ignore dependencies between words. Proposed model of analysis is based on Link Grammar but linking requirements were considerably extended. They include dependencies between values of grammatical categories in order to cope with inflection related phenomena.

REFERENCES

[1] WABIK A., *Detekcja błędów fleksyjnych w komputerowych edytorach tekstu*, M.Sc. thesis, Wrocław University of Technology, Wrocław, 2010, 5–11.
[2] MYKOWIECKA A., MARCINIAK M., *Domain-driven automatic spelling correction for mammography reports*, Proceedings of IIPWM'06 Conference, Ustroń, Springer-Verlag, 2006, 521–530.
[3] SALONI Z., ŚWIDZIŃSKI M., *Składnia współczesnego języka polskiego*, Warszawa, Wydawnictwo Naukowe PWN, 1998, 108–123.
[4] WOLIŃSKI M., *System znaczników morfosyntaktycznych w korpusie IPI PAN*, Polonica XXII-XXII, Kraków, Wydawnictwo LEXIS, 2003, 39–54.
[5] SLEATOR D.D.K., TEMPERLEY D., *Parsing English with a link grammar*, Technical Report CMU-CS-91-196, School of Computer Science, Carnegie Mellon University, Pittsburgh, 1991, 7–26.
[6] MIERZWA J., *Model formalny komputerowej dekompozycji zdań języka naturalnego na grupy słów do celów wnioskowania przez komputer*, Ph.D. thesis, Wrocław University of Technology, Wrocław, 2001, 29–40.

Decisional DNA, Set of Experience Knowledge Structure,
Smart TV, Digital TV, XML

Haoxi ZHANG*, Cesar SANIN**, Edward SZCZERBICKI***

APPLYING FUZZY LOGIC TO DECISIONAL DNA DIGITAL TV

In this paper, we propose the idea of applying fuzzy logic methods to the Decisional DNA Digital TV. The integration of the Decisional DNA DTV and fuzzy logic provides the Digital TV viewer with better user experience. Decisional DNA is a domain-independent, flexible, and standard experiential knowledge representation structure that allows its domains to acquire, reuse, evolve, and share knowledge in an easy and standard way. The Decisional DNA DTV enables TV players to learn the viewer's watching habit discovered through past viewing experience and reuse such experience in suggestion of channels. The presented conceptual approach demonstrates how the Decisional DNA-based systems can be integrated with fuzzy logic technique, and how it captures and deals with the TV viewer's watching experience in a fuzzy logic way.

1. INTRODUCTION

Decisional DNA Digital TV (DDNA DTV) is an experiential knowledge-based system that captures the viewer's TV watching habit and reuses such knowledge on suggestion of channels to the viewer [6]. This paper discusses the continuation of the development of the DDNA DTV introduced in previous research published in [6], [7]. This research extension explores the possibilities and the benefits of applying fuzzy logic technique to the DDNA DTV.

* The University of Newcastle, University Drive, Callaghan, 2308, NSW, Australia. Haoxi.Zhang@uon.edu.au
** The University of Newcastle, University Drive, Callaghan, 2308, NSW, Australia. Cesar.Sanin@Newcastle.edu.au
*** Gdansk University of Technology, Gdansk, Poland.Edward.Szczerbicki@zie.pg.gda.pl

1.1. FUZZY LOGIC

The concept of fuzzy logic began with the 1965 proposal of fuzzy sets by Zadeh [10]. Rather than exact and fixed values in traditional logic theory,where binary sets have only two values (i.e. true and false), fuzzy logic may have many values that ranges in degree between 0 and 1.Therefore, fuzzy logic is most suitable for modeling the uncertainty in human reasoning, *e.g. warm, small*.

There are three fundamental concepts in fuzzy logic theory, namely, fuzzy sets, linguistic variables, and possibility distributions [9]. Let U be a collection of values between 0 and 1, which could be discrete or continuous. *U* is called the universe of discourse, or simply the universe, and *u* represents the generic element of *U*.

Fuzzy Set: a fuzzy set *F* in a universe *U* is determined by its membership function (MF) μ_F where $\mu_F(x) \in [0, 1]$.

$$F = \{(u, \mu_F(u)) \mid u \in U\} \tag{1}$$

A fuzzy set can also be represented by

$$F = \begin{cases} \sum_{i=0}^{n} \frac{\mu_F(u_i)}{u_i}, & \text{If } U \text{ is discrete} \\ \int_U \frac{\mu_{F(u)}}{u}, & \text{If } U \text{ is continuous} \end{cases} \tag{2}$$

Linguistic Variable: a linguistic variable is a variable whose value can be defined quantitatively using an MF and qualitatively using a linguistic term [9]. For example, the speed of a car is a linguistic variable, whose value can be slow, moderate, and fast.

Possibility Distribution: a possibility distribution is the elastic constraint on the possible values of the variable imposed when a linguistic variable is assigned toa fuzzy set.

Fuzzy logic is an effective tool for modeling the uncertainty in human reasoning. In fuzzy logic, the knowledge of human beings is codified by means of linguistic IF-THEN rules which build up the fuzzy inference systems (FISs). FISs can be used in many areas such as in data analysis, control and signal processing.

1.2. DECISIONAL DNA AND SOEKS

The Decisional DNA is a knowledge repository that organizes and manages formal decision events stored in the Set of Experience Knowledge Structure (SOEKS) [1]. The SOEKS has been developed to acquire and store formal decision events in an explicit way [2]. It is a model based upon available and existing knowledge, which must adapt to the decision event it is built from (i.e. it is a dynamic structure that depends on the information provided by a formal decision event) [1]; besides, it can be represented in XML or OWL as an ontology in order to make it transportable and shareable [3], [4].

SOEKS is composed of variables, functions, constraints and rules associated in a DNA shape permitting the integration of the Decisional DNA of an organization [1]. Variables normally implicate representing knowledge using an attribute-value language (i.e. by a vector of variables and values) [2], and they are the center root of the structure and the starting point for the SOEKS. Functions represent relationships between a set of input variables and a dependent variable; moreover, functions can be applied for reasoning optimal states. Constraints are another way of associations among the variables. They are restrictions of the feasible solutions, limitations of possibilities in a decision event, and factors that restrict the performance of a system. Finally, rules are relationships between a consequence and a condition linked by the statements IF-THEN-ELSE. They are conditional relationships that control the universe of variables [1].

Additionally, SOEKS is designed similarly to DNA at some important features. First, the combination of the four components of the SOE gives uniqueness, just as the combination of four nucleotides of DNA does. Secondly, the elements of SOEKS are connected with each other in order to imitate a gene, and each SOE can be classified, and acts like a gene in DNA [1]. As the gene produces phenotypes, the SOE brings values of decisions according to the combined elements. Then a decisional chromosome storing decisional "strategies" for a category is formed by a group of SOE of the same category. Finally, a diverse group of SOE chromosomes comprise what is called the Decisional DNA [1].

In short, SOEKS and Decisional DNA provide an ideal approach which can not only be very easily applied to various embedded systems (domain-independent), but also enable standard knowledge communication and sharing among these embedded systems [6].

2. DECISIONAL DNA DIGITAL TV

Digital television (DTV) is the television broadcasting system that uses the digital signals to transmit program contents [13]. In order to capture, reuse, and share viewers' TV watching experiences, we applied the knowledge representation approach – Decisional DNA, to our research, and combined it with digital TV, called the Decisional DNA Digital TV.

In order to make the DDNA DTV more compatible with different TV players, we modified the architecture based on our previous work [6]; and the newly designed DDNA DTV is more like an open API (Application Programming Interface) [12] to whom interested in adding new features into their TV products and making their products be capable of capturing users' watching habits and reusing such knowledge in improving the user experience of their products.

The DDNA DTV consists of the Decisional DNA Repository, the Decisional DNA Repository Manager, the Convertor, the Prognoser, the Client, and the Server (see Fig. 1).

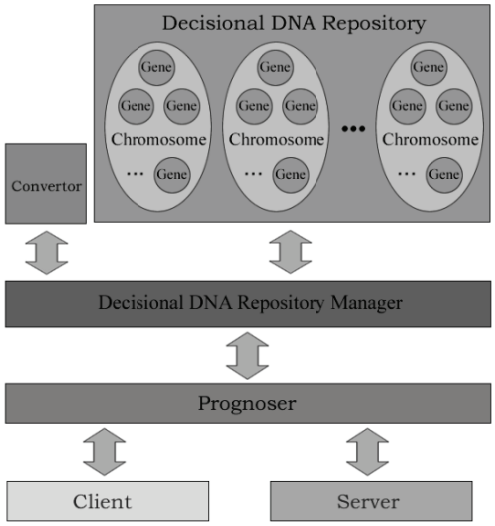


Fig. 1. System architecture for Decisional DNA DTV

• *Decisional DNA Repository*: The Decisional DNA Repository is the place where the viewer's TV watching experiences are stored and managed. It uses the Decisional DNA concept introduced in Section 1.2 of this paper to organize such experiential knowledge. It is composed of the Genes and the Chromosomes:

a) Genes: The Gene carries a single TV watching experience that represented by a set of XML tags described in [3], and stored as an XML file.

b) Chromosomes: The Chromosome stores a group of Decisional DNA Genes under the same category. It is used to capture the decisional "strategies" for a category.

• *Decisional DNA Repository Manager*: The Decisional DNA Repository Manager is the interface of the Decisional DNA Repository. It answers operation commands sent by the Prognoser and manages the XML files.

• *Convertor*: The Convertor translates the XML-represented Decisional DNA experience stored in Genes into SOEKS for reusing.

• *Prognoser*: The Prognoser is in charge of sorting, analyzing, organizing, creating and retrieving experience. It sorts data received from the Client, and then, it analyzes and organizes the data according to the system configuration. Finally, it interacts with the Decisional DNA Repository Manager to store or to reuse experience according to different tasks.

• *Client*: The Client is designed as the interface of the DDNA DTV approach in order to receive data from its domain. In particular, through the Client the digital TV player transfers data to DDNA DTV, and then, the Client will pass those data to the Prognoser for further processing.

• *Server*: The Server is developed as another interface of the DDNA DTV approach in order to send data to its domain. By importing the Server, the DTV player can query the recommended TV channels for the viewer, and get reminders for the next suggested TV program.

3. APPLYING FUZZY LOGIC TO DECISIONAL DNA DTV

In the latest research of the Decisional DNA DTV [7], we combined the DDNA DTV with the TVHGuide [11], and implemented this combination on a Toshiba Shriving tablet. It enables the viewer to watch live TV on the Android tablets,as well as captures and reuses the viewer's TV watching experience. However, we want to improve the Decisional DNA DTV in many ways. In this section, we introduce how we add the functionalities of the Viewer Group Classification and the Viewer Group Sharing to the Decisional DNA DTV.

The Viewer Group Classification enables the Decisional DNA DTV to estimate the viewer's profile according to the viewer's watching habit. For example, if a viewer who watches TV usually on weekdays between 9:00 a.m. and 4:00 p.m., we can estimate him/her as a househusband/housewife, because it looks like he/she doesn't need to work. And the Viewer Group Sharing allows viewers to share their TV watching experiences among groups and get TV program suggestions from their group mates. For instance, if viewer A is classified as a housewife, viewer A will contribute her TV watching experience to the Housewife group, also, viewer A will get TV program suggestions from Housewife group. Furthermore, the group sharing can help the viewer quickly adapt to a new TV network when he switches his TV network or move to other country; and give the possibility of playing more interested advertisements to the viewer. In order to give the capabilities of the Viewer Group Classification and the View Group Sharing to the Decisional DNA DTV, we introduce fuzzy inference technique to our research.

The fuzzy inference system (FIS) is a systematic framework that uses fuzzy rules to map between inputs and outputs, in which the data flow usually goes through three procedures, namely fuzzification, fuzzy inference, and defuzzification (see Fig. 2) [9].

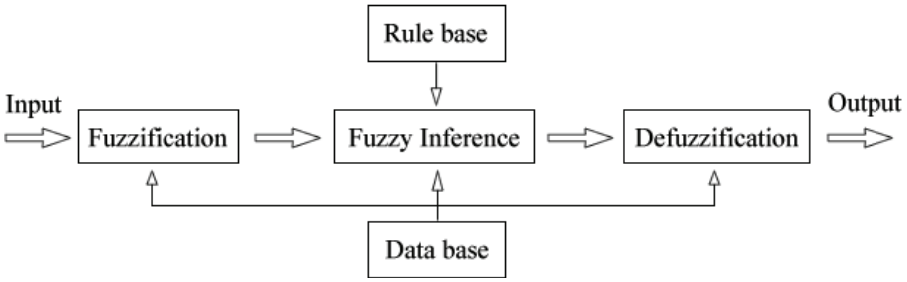


Fig. 2. The architecture of a fuzzy inference system

First, the input data is converted into suitable linguistic values or fuzzy setsin fuzzification. Then, the fuzzy inference process performs simulation of human reasoning based on its rule base and data base. The rule base defines the mapping relationship between inputs and an output using linguistic terms,also, it describes reasoning policy by a set of linguistic IF-THEN logic. While in the data base, necessary definitions of linguistic control rules and fuzzy data manipulation are addressed [9], [10]. Finally, the defuzzification produces a crisp output for further use.

The core of fuzzification is building the proper membership function [9]. In our case, we introduce some TV audience related studies [5] to the Viewer Group Classification part. Fig. 3 shows a very initial test of this concept.

Fig. 3. Screenshot of initial fuzzy inference test

The main idea of the Viewer Group Classification is that using statistic data and study to estimate viewer profile according to the viewer's watching habit, for example, if we know: when, what, and how long per week a viewer usually watches TV, we can measure him through the FIS, and finally create a profile for him. The fuzzy logic tool, jFuzzyLogic [8], is used in our work. It allows us to use fuzzy logic very easily in Java based projects.

4. CONCLUSIONS AND FUTURE WORK

In this paper, we introduce the conceptual ofapplying fuzzy logic tothe Decisional DNA DTV. Also, the initial tests we did on a DELL laptop with jFuzzyLogic are presented. As the result shows, fuzzy logic technique can be integrated with the Decisional DNA DTV, and bring functionalities of fuzzy inference to its domain.

To continue with this idea, further research and refinement are required, some of them are:

– Build proper membership function.
– Further development of the fuzzy rule base and data base.
– Refinement and further development of algorithm using in the Prognoser.

REFERENCES

[1] SANIN C., MANCILLA-AMAYA L., SZCZERBICKI E., CAYFORD-HOWELL P., *Application of a Multi-domain Knowledge Structure: The Decisional DNA*, Intel. Sys. For Know. Management, SCI 252, 65–86, 2009.

[2] SANIN C., SZCZERBICKI E., *Experience-based Knowledge Representatio SOEKS*, Cybernetics and Systems, Vol. 40, No. 2, 99–122, 2009.

[3] SANIN C., SZCZERBICKI E., *Extending Set of Experience Knowledge Structure into a Transportable Language Extensible Markup Language*, International Journal of Cybernetics and Systems, Vol. 37, No. 2-3, 97–117, 2006.

[4] SANIN C., SZCZERBICKI E., *An OWL ontology of Set of Experience Knowledge Structure*, Journal of Universal Computer Science, Vol. 13, No. 2, 209–223, 2007.

[5] COMSTOCK G., CHAFFEE S., KATZMAN N., McCOMBS M., ROBERTS D., *Television and Human Behavior*, Columbia University Press, 1978.

[6] ZHANG H., SANIN C., SZCZERBICKI E., *Making Digital TV Smarter: Capturing and Reusing Experience in Digital TV*, Cybernetics and Systems: An International Journal, Taylor & Francis Group, LLC. Vol. 43, Issue 2,.127–135, 2012.

[7] ZHANG H., SANIN C., SZCZERBICKI E., *The Development of Decisional DNA DIGITAL TV*, 16th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems, San Sebastian, September 2012 (in press).

[8] jFuzzyLogic, http://jfuzzylogic.sourceforge.net/html/index.html

[9] DU K.-L., SWAMY M.N.S., *Neural Networks in a Softcomputing Framework*, Springer, London 2006.

[10] ZADEH L., *Fuzzy sets*, Information and Control, Vol. 8, Issue 3, 338–353, 1965.
[11] TVHGuide, About, http://john-tornblom.github.com/TVHGuide/index.html
[12] Wikipedia, Application Programming Interface, http://en.wikipedia.org/wiki/Application_programming_interface
[13] WU Y., HIRAKAWA S., REIMERS U., WHITAKER J., *Overview of digital television development worldwide*, Proc. IEEE, Vol. 94, No. 1, 8–21, 2006.

digital video, multimedia retrieval, content-based video indexing,
Automatic Video Indexer AVI, video indexing strategies

Kazimierz CHOROŚ *

NEW CONTENT-BASED INDEXING ALGORITHMS IN AUTOMATIC VIDEO INDEXER AVI

The Automatic Video Indexer AVI is a research project investigating tools and techniques of automatic video indexing for retrieval systems. The main goal of the project AVI is to develop efficient algorithms of content-based video retrieval. Several strategies have been proposed, implemented and tested, and they are still being intensively developed. The most simple techniques are based on the comparison of video frames histograms. The most advanced approaches use different algorithms of content analysis based on image recognition and artificial intelligence. New investigations on content-based video indexing performed, being curried out, or envisaged in the Automatic Video Indexer will be presented and some new results will be also discussed.

1. INTRODUCTION

It becomes trivial to state that multimedia became very popular not only in local multimedia systems but also in the Web. Multimedia retrieval systems demand specific technologies, methods, algorithms, techniques, frameworks, etc. for efficient access to desired multimedia data. In visual retrieval systems allowing the storage and easy and efficient access to desired videos appropriate content-based indexing and retrieval methods of these video data should be applied. Manual indexing is unfeasible for very large video collections. On the other hand, automatic indexing methods are not satisfactory and, furthermore, the content is very subjective to be completely characterized. The content of videos covers much more than traditional text. It is related to many technical and formal parameters of videos such as movie category, duration, length, compression method, resolution, color depth, language, subtitles, production or post-

* Wrocław University of Technology, Institute of Informatics, Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland; e-mail: kazimierz.choros@pwr.wroc.pl

ing date, etc. But we are much more interested in content aspects of videos, such as main objects, humans (main actors as well as extras), animals, second plan, background, domain, context, etc. and then all these interesting elements in special actions, situations, environments, surroundings, or in special sequences of events.

Therefore, we are looking for effective tools to identify the special video segments in television broadcast like shots or scenes with a specific content, for example news on weather, sports, science, finances, technology, world travel, national economy, interviews, music video clips, or entertainment news. Sometimes we would like also to detect and to remove advertising spots during the broadcasting.

In the case of sports videos the content-based analyses should lead to the detection of player, playing fields, main events, special highlights, replays, and special occurrences for a given sport discipline such as goal, penalty, free or corner kicks, fouls, jumps, race finishes, tennis serve, runs, ski slalom running, downhill skiing, plunge, boxing hook, javelin throw, and many, many others.

The main purpose of sport video processing is to categorize the sport shots and scenes for example in TV sports news. The automatic categorization of sport events videos and then the detection of main sports highlights are fundamental processes in automatic indexing for content-based retrieval. The retrieval of news presenting the best or actual games, tournaments, matches, contests, races, cups, championships, etc. or special player behaviors, actions, wins or losses in a desirable sports disciplines will become more effective.

Methods of an automatic semantic categorization of sport video shots mainly of shots from TV sports news are of increasing importance because of a very high popularity of sport games in TV broadcasts, of a huge amount of broadcast sport videos generated every day, and of the large share of sport video materials in multimedia databases. Then, a great commercial appeal is also observed for sport video automatic indexing and retrieval systems. There is no one method that is best suited for every category of movie or for every sport disciplines. Therefore, many different strategies have been proposed and they are still developing. Because of a great variety of sports videos many tests, many experiments, and many comparative investigations should be carried out.

The chapter is organized as follows. At the beginning some main related works in the area of automatic indexing of sport videos and of video scene categorization will be discussed. Next the Automatic Video Indexer AVI project will be presented. Then strategies in sport categorization will be shortly remind: colour histogram comparison, text detection, sport object, face, player and audience emotions detection and analysis. Finally, the new approaches in the categorization of sport video shots of TV sports news will be discussed. The final conclusions and the future research work areas are discussed in the last part of the chapter.

2. RELATED WORKS

Many investigations have been carried out in the area of automatic recognition of video content and of visual information indexing and retrieval [1, 12, 13, 23]. The main goal of all these investigations is to develop procedures for efficient retrieval of videos stored in more and more huge multimedia databases in multimedia archives in local systems and in the Web. The automatic video indexing also includes such processes as automatic detection or generation of highlights, video summarization, and video categorization.

A unified framework for semantic shot classification in sports videos has been proposed in [22]. The framework has been tested over three videos types of very popular sports disciplines: tennis, basketball, and soccer. Tennis is one of the sport disciplines very frequently used on content-based indexing experiments. The goals of the proposed approaches are an automatic detection of highlights in tennis games [14], action recognition [29], player detection and tracking [16], detection of faces in tennis video scenes of TV sports news [4], and event detection in tennis videos based on trajectory analysis [3].

Automatic annotation of soccer videos is also a very common approach. This approach has resulted in detecting principal highlights including goals [18] and replays [27], and recognizing identity of players based on face detection, and on the analysis of contextual information such as jersey's numbers and superimposed text captions. Some tests have also been performed in the AVI Indexer leading to the detection of soccer shots in TV sports news [7, 8].

New methods have been proposed and experiments have been carried out not only with tennis, soccer, or basketball videos but also with for example baseball videos [21], with badminton [28], as well as with other sports.

3. AUTOMATIC VIDEO INDEXER

The Automatic Video Indexer AVI [7, 8] is a research project investigating tools and techniques of automatic video content-based indexing for retrieval systems. The standard process of automatic content-based analysis and video indexing is composed of several stages. The first step of video indexing is a temporal segmentation leading to the segmentation of a movie into small parts called video shots. Next, shots are grouped to make scenes, and then content of shots and scenes is analyzed [5–9]. The analyses of scenes permit to classify TV sports news and to detect important events and people, and then to extract interesting highlights, which facilitate browsing and retrieval of sports video. For several years the AVI Indexer enables us to carry out a rich variety of experiments on content-based indexing.

4. STRATEGIES FOR CONTENT-BASED VIDEO INDEXING

Because the content is very subjective and not easy to recognize different strategies are used for content analyses of digital videos. They may be based on the traditional comparison of still frames with image patterns and on the detection of different specific elements of digital videos. In the case of TV sports news such elements are: lines in playing fields, player faces, sport equipments, etc. [10, 11]. Let us remind these basic approaches in content analyses.

4.1. COMPARISON OF IMAGE PATTERNS

Color histogram provides a useful clue for measuring the similarity between images [2]. Therefore, histogram matching is a commonly-adopted technique not only in the applications of pattern recognition. Such an approach has also been applied in the Automatic Video Indexer for the detection of shots as well as for the content analysis of scenes. A similarity measure is used to compare given patterns to the video frames. The procedure is time-consuming because of a great number of frames in every video clip. Therefore, it has been proposed to transfer each video frame to a color string using straightforward rules. Then it has been shown that by transferring the video frames and image patterns comparison to strings comparison the computational complexity is significantly decreased.

4.2. LINE DETECTION IN THE PLAYING FIELDS

The second strategy leads to the detection of playing fields by detecting boundary lines, the penalty area, goal line – the end line between the goal posts in soccer, back boundary lines in tennis or basketball, etc. The identification of the pixels that belong to court lines is possible because court lines are usually white or significantly distinguishable from the background. The objective is also to discard the audience area possibly present on the sides and/or on the top of the frame. The pixels near the borders of the frame are analyzed to look for those pixels whose hue value is not belonging to the court color nor to the court line color (white) [29]. Of course in ice hockey, basketball, volleyball, handball, or in many other mainly hall sports or tennis, box, etc. lines are not white but always significantly stand out from the field.

Many sports have well-defined line structures on the playfield. Nevertheless, not all lines of a playing filed must be detected to recognize a genre of a playing field. A minimum, sufficient set of detected lines for every playing field can be defined for the categorization of sport shots. The results of tests performed in the AVI Indexer have shown the usefulness of this strategy of these reduced requirements.

4.3. DETECTION OF SUPERIMPOSED TEXT

Text is frequently present in a video. Text is usually superimposed on the images, or included as closed captions. It Is in the form of title or names of movie stars and of other artists, of the director and of the producers, screenwriters, stage designers, etc., Text is omnipresent because in any movie we can observe different words on different objects, products, cars, buildings, publicity billboards, etc. Very nice examples of text superimposed in videos, mainly television broadcast are presented in [26]. Extraction of text information involves detection, localization, tracking, extraction, enhancement, and recognition of the text from a given video frame [17].

Text is also frequently present in TV sports news in the form of player names, team names, league tables, numeric results, time, etc. These textual elements are usually very characteristic for a specific sport discipline, so, they can serve as an important indicator in content-based indexing process.

4.4. DETECTION OF PLAYER FACES

The most important issue in automatic face recognition process is face detection. Main aspect is the distinction and the rejection of objects resembling faces but which are not. It happens that raised up hand is taken as a face because of factors considered during face detection such as shape and color. Then, when a single face is located, we should extract the specific points such as eyes, eyebrow line, corners of the mouth, the whole mouth, nose, and other related to the chosen method of identification. These points are used to determine the values of such face parameters as: symmetry, distance between the eyes, the distance between the line of eyes, and lips [20].

Face detection is a crucial technology for applications such as face recognition, automatic lip-reading, and facial expression recognition and can be also crucial for content-based video indexing. The module for automatic face detection in digital videos is developing in the Automatic Video Indexer [4].

4.5. DETECTION OF SPORT OBJECTS

In many sport disciplines different objects are used such as ball, disc, cricket bat, javelin, tennis racket, hockey stick, net, soccer post, springboard, diving board, and many others. Players are using different sport equipments, protective equipments, wear, footwear, etc. The recognition of these objects in videos can help to identify the content and the sport discipline in a given video scene. The Haar cascades are the most popular technique for object recognition including face recognition. A rectangular Haar-like feature is defined as the difference of the sum of pixels of areas inside the rectangle, which can be at any position and scale within the original image.

4.6. DETECTION OF PLAYER AND AUDIENCE EMOTION

Emotion analysis is relatively a novel viewpoint which tries to recognize user reactions such as "exciting", as well as "happy" and "sad" emotion while observing a sports video broadcast. The segments with different kinds of emotions can be further used for highlight summarization and event detection to comply with user preference. Several promising machine learning algorithms for emotion detection have been tested which include techniques such as Bayesian networks, decision trees, and others [24].

Humans interact with each other mainly through speech, but also through body gestures. Humans display emotions through facial expressions. Emotions can be classified into six categories, such as anger, disgust, fear, happiness, sadness and surprise. In the case of sports videos two emotions are generally observed: happiness and sadness emotions reflecting the sport results: wins or losses.

The strategy consists in creating an authentic facial expression database based on spontaneous emotions and then in comparing these patterns with video frames. The reactions of players and fans are willingly presented during the broadcasting because these video scenes are very attractive for the TV audience.

5. NEW APPROACHES

Several news strategies are envisaged to be developed and tested in the AVI Indexer. These new approaches are mainly based on the conclusions of previous research [10] that the most promising strategy seems to be that one based on the structure of video shows and on the repetitive patterns of scenes due to the fact that TV editing studios make TV shows in very standard regular ways.

5.1. EXTRACTION OF CAMERA MOTION PARAMETERS

I has been observed [25] that in most of sport videos, camera motions are closely related to the actions taken in the sports, which are mostly based on a certain rule depending on types of sports. In consequence parameters of camera motions contain very significant information for categorization of sports video. Camera motion parameters can be extracted directly from the analyzed video by analyzing motion information. The camera motions do not depend on the dominant color. They are similar for the tennis games on both green grass court like the Wimbledon courts as well as on red clay court like the Roland Garros courts in France (French Open) or on the hard courts (decoturf courts in New York – US Open or plexicushion courts in Melbourne in Australia – Australian Open) which are of any color. Camera motions do not depend on whether conditions or holding time (day-time games as well as night-time games).

Such an approach has been already tested in [25]. Several sport disciplines such as baseball, football, soccer, sumo, and tennis have been statistically characterized and these statistical characteristics mainly depend on the types of sports.

5.2. DETECTION OF OBJECTS OF INTEREST

The detection of objects of interest has been widely used in many recent works in video analysis, especially in video similarity and video retrieval [19]. In the case of the categorization of sports videos the detection of objects of interest can be seen as an extension or generalization of such already defined strategies like face detection and sport object detection.

5.3. AUDIO-VISUAL INDEXING

Audio information is an important data for automatic categorization of sports videos [15]. Most of the common video genres have very specific audio characteristics, e.g. specific music, fan spot music, fan chants, in news there are a lot of monologues and dialogues, natural sounds, the specific audience noise and vocal reactions, etc.

The integration of an audio analysis with a temporal video segmentation seems to be very effective for a sports video categorization and for a sport highlights detection.

5.4. CONTENT RECOGNITION BASED ON A SCENE STRUCTURE

For structure analysis, the domain specific features used in existing systems make it difficult to extend an approach from one kind of video to another. However, comparing to other kinds of video documents, sports videos have definite structures, with so-called repetitive patterns. A sports game usually occurs in one specific playfield and is often recorded by a number of cameras with fixed positions. Generally, a dominant camera, placed along one of the long sideline of playfield, is used to follow the action in game and provide a global view to audiences (especially in "field" sports videos such as soccer, basketball, volleyball, football, and so on). Many sport videos such as for example archery, diving, soccer, or tennis have easily detected repetitive structure patterns.

6. NEW EXTENDED SCHEME OF THE AUTOMATIC VIDEO INDEXER

The development of the AVI Indexer takes place mainly in the area of scene analyses and video structure detections.

The new extended scheme of the Automatic Video Indexer AVI is presented in Figure 1.

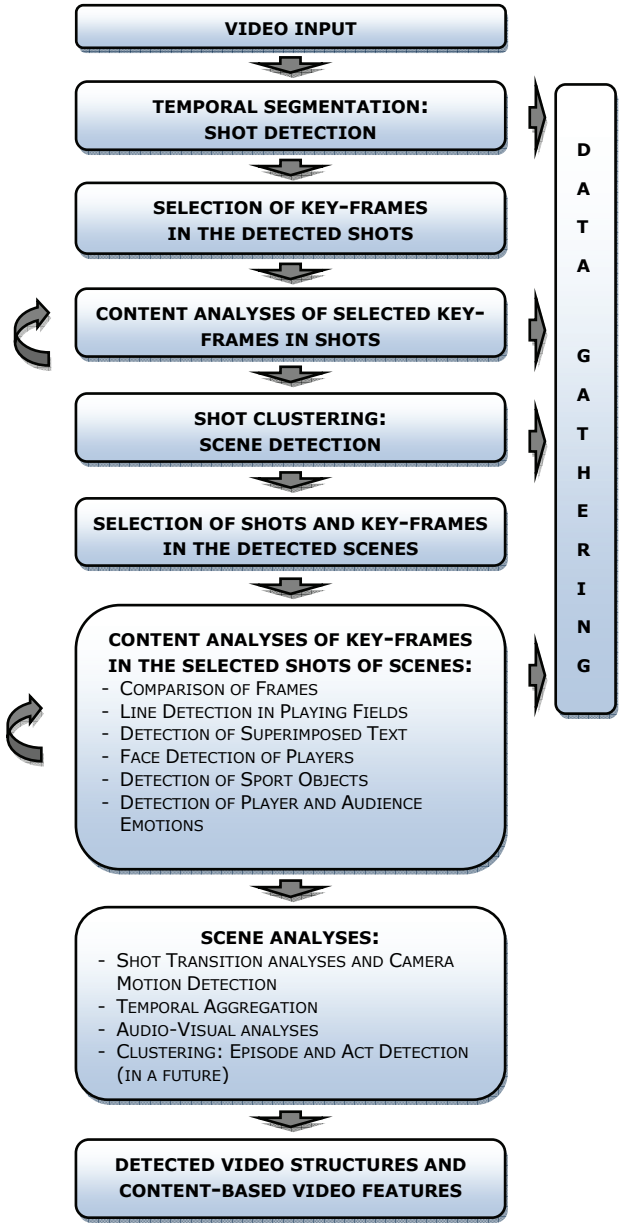


Fig. 1. Extended scheme of the Automatic Video Indexer AVI (extended version of previous schemes [8, 11])

7. FINAL REMARKS AND FURTHER STUDIES

Content-based video indexing and retrieval are widely studied in the literature and intensively investigated theoretically as well as experimentally. The content is very subjective to be easily and completely characterized. Many indexing frameworks have been already proposed. Several strategies for content-based video indexing have been presented which are implemented or being implemented in the Automatic Video Indexer. The Automatic Video Indexer is a research project investigating tools and techniques of automatic video indexing for retrieval systems. New approaches defined in this chapter should enlarge the spectrum of algorithms applied for sports video indexing and categorization.

The most promising strategy seems to be the strategy based on the recognition of a video structure and the detection of repetitive patterns of scenes typical for a given TV editing studios or for a given TV show. The next step of our experimental investigations will be the verification of hybrid strategies. Individual algorithms are quite efficient, but the simultaneous application of several, different methods may result in synergistic effect and may lead to the implementation of practically useful method.

REFERENCES

[1] BALLAN L., BERTINI M., Del BIMBO A., SEIDENARI L., SERRA G., *Event detection and recognition for semantic annotation of video*. Multimedia Tools and Applications, 2011, Vol. 51, 279–302.

[2] CHA S., *Taxonomy of nominal type histogram distance measures*. In: Proceedings of the American Conference on Applied Mathematics, 2008, 325–330.

[3] CHI-KAO C., MIN-YUAN F., CHUNG-MING K., NAI-CHUNG Y., *Event detection for broadcast tennis videos based on trajectory analysis*. Proc. 2nd International Conference on Communications and Networks (CECNet), 2012, 1800–1803.

[4] CHOROŚ K. and FIJAŁKOWSKI D., *Detection of faces in tennis video scenes of TV sports news*. In: Information Systems Architecture and Technology: System Analysis Approach to the Design, Control and Decision Support, 2010, Wrocław, Oficyna Wydawnicza Politechniki Wrocławskiej, 127–137.

[5] CHOROŚ K., *Digital video segmentation techniques for indexing and retrieval on the Web.* In: Advanced Problems of Internet Technologies. Academy of Business, 2008, 7–21.

[6] CHOROŚ K., *Video shot selection and content-based scene detection for automatic classification of TV sports news.* In: Internet – Technical Development and Applications, AISC, Vol. 64/2009. Heidelberg, Publisher Springer, 2009, 73–80.

[7] CHOROŚ K., *Video structure analysis and content-based indexing in the Automatic Video Indexer AVI.* In: Nguyen N.-T. et al. (Eds.), Advances in Multimedia and Network Information System Technologies, AISC, Vol. 80/2010. Heidelberg, Publisher Springer, 2010, 79–90.

[8] CHOROŚ K., PAWLACZYK P., *Content-based scene detection and analysis method for automatic classification of TV sports news.* Rough Sets and Current Trends in Computing, Lecture Notes in Artificial Intelligence LNAI 6086, 2010, 120–129.

[9] CHOROŚ K., *Reduction of faulty detected shot cuts and cross dissolve effects in video segmentation process of different categories of digital videos.* Transactions on Computational Collective Intelligence V, Lecture Notes in Computer Science LNCS 6910, 2011, 124–139.
[10] CHOROŚ K., *Strategies for content-based digital video indexing and retrieval*. In: Information Systems Architecture and Technology: New Developments in Web-Age Information Systems, 2010, Wrocław, Oficyna Wydawnicza Politechniki Wrocławskiej, 211–222.
[11] CHOROŚ K., *Video structure analysis for content-based indexing and categorisation of TV sports news*. Int. Journal of Intelligent Information and Database Systems, Vol. 6 (in press) (2012).
[12] GEETHA P., NARAYANAN V., *A survey of content-based video retrieval*. Journal of Computer Science, 2008, Vol. 4, No. 6, 474–486.
[13] HU W., XIE N., LI L. ZENG X., MAYBANK S., *A survey on visual content-based video indexing and retrieval*. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, 2011, Vol. 41 , No. 6, 797–819.
[14] HUANG Y., CHOIU C., SANDNES F.E., *An intelligent strategy for the automatic detection of highlights in tennis video recordings.* Expert Systems with Applications, 2009, Vol. 36, No. 6, 9907–9918.
[15] IONESCU B., SEYERLEHNER K., RASCHE C., VERTAN C., LAMBERT P., *Content-based video description for automatic video genre categorization*. In: K. Schoeffmann et al. (Eds.): MMM 2012, Lecture Notes in Computer Science LNCS 7131, 51–62.
[16] JIANG Y., LAI K., HSIEH C., LAI M., *Player detection and tracking in broadcast tennis video*. In: Wada T. et al. (Eds.): PSIVT, Lecture Notes in Computer Science LNCS 5414, 2009, 759–770.
[17] JUNG K. , KIM K.I., JAIN K.A., *Text information extraction in images and video: a survey*. Pattern Recognition, 2004, Vol. 37, No. 5, 977–997.
[18] KANG Y.-L., LIM J.-H., KANKANHALLI M.S., XU C., TIAN Q., *Goal detection in soccer video using audio/visual*. Proceedings of the ICIP, 2004, 1629–1632.
[19] KOWDLE A., CHANG K.-W., CHEN T., *Video categorization using object of interest detection.* Proc. of the 17th International Conference on Image Processing (ICIP'2010), 2010, 4569–4572.
[20] LI S.Z, JAIN A.K., *Handbook of face recognition*. New York, Springer, 2005.
[21] LIEN C.-C., CHIANG C.-L., LEE C.-H., *Scene-based event detection for baseball videos.* Journal of Visual Communication and Image Representation, 2007, 1–14.
[22] LING-YU D., MIN X., QI T., CHANG-SHENG X., JIN J.S., *A unified framework for semantic shot classification in sports video*. IEEE Transactions on Multimedia, 2005, Vol. 7, No. 6, 1066–1083.
[23] MONEY A.G., AGIUS H., *Video summarisation: a conceptual framework and survey of the state of the art*. Journal of Visual Communication and Image Representation, 2008, 121–143.
[24] SUN Y., SEBE N., LEW M., GEVERS T., *Authentic emotion detection in real-time video.* In: N. Sebe et al. (Eds.), HCI/ECCV, LNCS 3058, 2004, 94–104.
[25] TAKAGI S., HATTORI S., YOKOYAMA K., KODATE A., TOMINAGA H., *Sports video categorizing method using camera motion parameters*, International Conference on Multimedia and Expo, July 2003, Vol. II, 461–464.
[26] WOLF C., JOLION J-M., *Détection et extraction de texte de la vidéo.* RFV (EAD 3038), Lyon, INSA de Lyon, 2001 (http://liris.cnrs.fr/m2disco/coresa/coresa-2001/coresa2001/articles/39.pdf).
[27] YANG Y., LIN S., ZHANG Y., TANG S., *A statistical framework for replay detection in soccer video.* IEEE International Symposium on Circuits and Systems ISCAS'2008, 2008, 3538–3541.
[28] YANG, Y., LIN, S., ZHANG, Y., TANG, S.: *Statistical Framework for Shot Segmentation and Classification in Sports Video*. Lecture Notes in Computer Science LNCS 4844, 2007, 106–115.
[29] ZHU G., XU C., HUANG Q., GAO W., *Action recognition in broadcast tennis video*. Proc. 18th International Conference on Pattern Recognition (ICPR), 2006, Vol. 1, 251–254.

parallel programming, CUDA environment,
photometric data reduction

Jan KWIATKOWSKI*, Rafał PAWŁASZEK*

ASTRONOMICAL PHOTOMETRIC DATA REDUCTION USING GPGPU

Astronomical photometry is one of the sciences, that benefit from the recent technological development in order to augment the quality and the quantity of the processed data. The planned projects, such as the European SOLARIS and the American LSST promises to generate the amount of data that will be a challenge for modern astronomical data reduction methods. It creates the need to search for new methods of data reduction. In the chapter a method that uses GPGPU for data reduction is investigated. The graphics processor that in its beginning aimed at fast screen image computation and presentation naturally adopt SIMD model of processing. This model fits very well in the reduction process of the contemporary photometric data received with the use of CCD cameras, that are in the two-dimensional form. The chapter presents the library for the photometric data reduction that uses flat field reduction, dark and bias current reduction with the use of CUDA environment, which enables to pass the computation onto graphics processors.

1. INTRODUCTION

Contemporary science has been developing in a rapid pace. One of the foundations of this growth is the computer development that took place by the end of the last century and progresses in an insatiable manner. A wide range of sciences such as quantum physics, nanotechnology, biomedical sciences, astronomy and many more profit from the emerged technical facilities. Astrophysical photometry is one of the sciences that has been experiencing this increased growth.

Astrophysical photometry is the science of measuring the brightness of stars. Photometric surveys give the data for various subfields, radial velocities' measurements, timing, astrometry, and many more [1], [8]. Contemporary projects, such as

* Institute of Informatics, Wrocław University of Technology, Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland.

Polish SOLARIS and American Large Synoptic Survey Telescope promises to provide a constant stream of high-resolution astronomical photometric data. The former, focused on the search of extraterrestrial planets is based on a set of telescopes that are able to observe the sky constantly, with the observations with time ranging from seconds to minutes. This will give a huge amount of data daily to be reduced, processed and analysed scientifically. The latter is to be started in 2014, and prides itself to be „the widest, fastest, deepest eye of the new digital age". Each observational night shall give terabytes of data, and the data is stated to be given to public. As it is a satellite telescope it will be able to also perform constant observations. The amount of data produced in each of them poses a great challenge for scholars, but also promises new insights and breakthroughs. What is then important is the automation of the reduction process.

Recent activities in the industry of chip development clearly show that the future HPC systems will be hybrid in nature [3]. The data-driven processing make them inclined towards the Single-Instruction-Multiple-Data approach in the Flynn's taxonomy. In the last years it has been realized that this approach is natively supported in graphical processing units (GPUs) and they have left central processing units (CPUs) far behind in the performance of computation, which is because the increasingly large portion of time is spent in CPUs on data movement rather than arithmetic. It is a reason that in the present time a wealth of scientific and commercial problems harness the computational power of GPUs and heterogeneous systems in general to solve the problems that once reached time limits due to the speed of processing on CPUs.

The chapter presents an approach of astrophysical photometric data preparation including bias-, dark-, and flat-field reduction with the usage of GPGPU component.

The structure of the chapter is as follows. The second section outlines the data contamination sources in the CCD photometry that must be reduced from the actual sky image for it to be ready for the scientific analysis. Third section presents the developed reduction library with the division onto the CUDA data-intensive computation and the interface provided for the usage. The natural hierarchy of data and processes in the field of astrophysical data reduction is preserved in the names and functionality of the library making it easy to understand and use by the astronomers. Section four presents the preliminary results of the usage of the library in comparison with the sequential approach to applied data reduction techniques. As the reduction of the observational signature, i.e. flat-field, dark and bias current is done by means of average functions, for an actual image to process multiple flat-, dark-, and bias-frames' exposures are taken and then their execution times and speedups are presented. The amount of frames is set to 5, 10, and 15 and the CCD chip resolution ranges from 1 to 100 megapixels. Finally section 5 concludes the work and presents the future plans.

2. IMAGE PROCESSING AND ANALYSIS IN CCD PHOTOMETRY

Nowadays, in astronomical photometry charge-coupled devices (CCD) are widely used. CCD plates are grids of photo-sensitive sensors that record the amount of photons that reach them [5], [7].

The images' preparation phase is an important element in astronomical photometry. Ideally, an image recorded using a CCD camera should give accurate information about the light flux distribution over a portion of the sky. However, this is not generally the case. Instrument imperfections and the discrete nature of light itself concur to introduce errors in the measured data. The image must undergo a reduction and analysis process to weed out all possible artefacts such as thermal changes, non-uniformity in the sensitivity of the CCD area and any other cosmetic faults, called in general the instrument signature, that may lead to the incorrect interpretation of the data gathered. Another equally important goal is to preserve information about the noise sources, so that users of the reduced data can evaluate the random errors of the data.

After the exposure each pixel of the image contains the value describing the amount of gathered energy. Let us denote it by *f*. *f* is a function of the position on the CCD chip *x* and *y,* respectively. It also is the function of time *t*, as the longer is the observation the more photons reach the surface of the CCD, giving eventually for each pixel on a CCD chip of the size *N***M*.

Telescopes usually do not illuminate the CCD plate homogeneously. Many physical characteristics along the way of the light, such as dust, eventually lead to lower data acquisition on the detector areas.

Also the sensitivity on each pixel does not necessarily has to be equal on the chip. Thus, the function *f* should be divided into

$$f(x,y,t) = t * S(x,y) * I(x,y) \tag{1}$$

where $S(x, y)$ defines the sensitivity function of the system and $I(x, y)$ the actual intensity of the brightness. The linear dependence on time *t* comes due the characteristics of the CCD chips. Normally, before or after the measurement of the real astronomical object to address this malicious feature there is an exposure of a uniform extended source of light, which states that $I(x, y) = \text{const} = L$ giving

$$f_F(x,y,t) = t * S(x,y) * L \tag{2}$$

It means that for a given exposure time *t* the function $f(x, y, t)$ varies over the frame of pixels only if the sensitivity $S(x, y)$ of the system that is not constant. Thus the equation defines a response of the system for a homogeneous source of light, and is called a *flat field*.

The CCD chip itself during the observation has a non-zero (in astronomy temperature is expressed in terms of Kelvin's) temperature. This temperature measures the kinetic energy of the particles that comprise the CCD chip. It means that there can occur a situation in which an electron can end up on a pixel even without the actual hit by a photon. This process on CCD chips is called a *dark current* and has to be taken into account during each measurement.

Fortunately, even though it varies for every pixel the dark current is very stable for the chip as a whole. The dark current exposure (or dark exposure for short) is done by taking the measurement without the actual exposure of the CCD chip to the source of light. This means that $I(x, y) = 0$. Dark current gives an additional factor

$$f_D(x,y,t) = t * D(x,y) \tag{3}$$

where $D(x, y)$ denotes the *unitary* dark current. Clearly, the function D is also a function of temperature, that is $D = D(x, y, T)$. Hence, it must be taken into account also during the data processing and reduction.

There is another effect that should be considered. The device itself is an electronic system and from that even the readout produces an error. This means that even in a zero-time exposure ($t = 0$), there will be captured an electronic and systematic error connected with the CCD camera thought of as an electronic system, giving:

$$f_B(x,y,t=0) = f_B(x,y) = B(x,y) \tag{4}$$

3. MOST IMPORTANT IMPLEMENTATION DETAILS

The proposed library is divided onto two parts (fig. 1). The first part is the C++ interface that is assumed to be used by programmers. The aim of this construct for the library is to be easily understood and adopted by not only the professional developers, but also by the amateurs and astronomers who do not have very profound knowledge of the C++ language nor CUDA programming models. Obviously, there is a minimal level of C++ that the users must possess.

The second part of the library is the CUDA bindings that the exposed interface shall consume. The C++ interface is exposed to the user hiding the CUDA implementation [2], [4], [6].

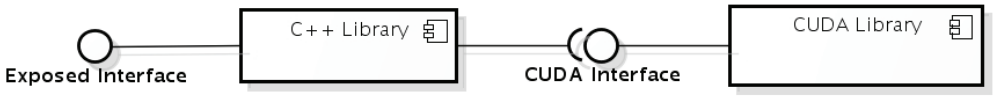


Fig. 1. The internal structure of proposed library

The interface exposed as the C++ library consists of `Frames`, that are the containers for the observational frames, there are four basic frame types:

- image frame – this is a result of an observation of the sky,
- bias frame – this is a result of an exposure with time set to zero to identify the natural noise in the work of the system,
- dark frame – this is a result of an exposure with the camera closed to determine the thermal noise of the CCD chip,
- flat frame – this is a result of an exposure of a homogeneous source of light to identify the discrepancies in the sensitivity of the CCD camera along the surface of the CCD chip.

In astronomical photometry there is also a notion of master-frames that are combined frames of the same kind. For example, a set of dark frames is combined to form one master-dark frame. The same idea stands behind master-bias and master-flat frames, although there is a different procedure in each. Therefore on the top of above types there are defined the new one:

- master bias frame – an extension to a bias frame that is defined as preprocessed set of bias frames to reduce the noise factor,
- master dark frame – an extension to a dark frame that is defined as preprocessed set of dark frames to reduce the noise factor,
- master flat frame – an extension to a flat frame that is defined as preprocessed set of flat frames to reduce the noise factor.

The frames serve as carriers of basic information from the cameras. Defined set of frames has to be extended with the reduction mechanisms. Therefore there is also `Reducer`, a class that provides the computation capabilities. This is the actual worker class over a set of frames. The main functions are `Combine` functions that take a set of bias-, dark- or flat frames and result with master-bias, master-dark, and master-flat frames respectively. These functions form the preprocessed set of auxiliary frames to reduce the actual `Image` frame by means of the function `Reduce`.

The path of the reduction process is shown in the figure 2. Having series of bias-, dark-, and flat-frames the algorithm requires the combination of a set of each kind to produce a master-bias, master-dark, and master-flat frames, respectively. For each type of auxiliary frames there is prepared a combination to reduce the noise part. In the next step bias current has to be reduced from dark- and flat-frames, whereas preprocessed dark current from the resulting flat frame. In the end of processing the auxiliary frames flat frame is normalized, as only the variation of the sensitivity is to be reduced. Then, the auxiliary frames are used to reduce the observational signature from themselves and finally, to prepare the image frame. Each of the function within the `Reducer` class is needed to fully implement the image reduction as prescribed by the equations in section 2.

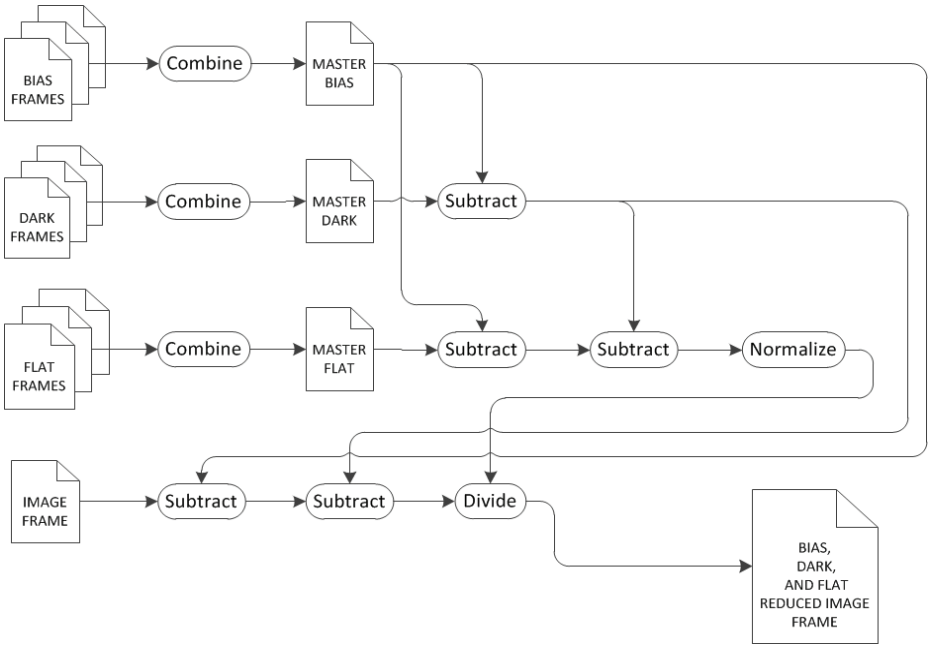


Fig. 2. The way of parallelization the data reduction process

Figure 3 and 4 present the image before and after reduction process, respectively. The image of the sky is taken with all intrinsic effects, including bias- dark-, and flat-field. Here the flat-field signature, that stands for the damping the sensitivity in the middle of the image, is emphasized most.

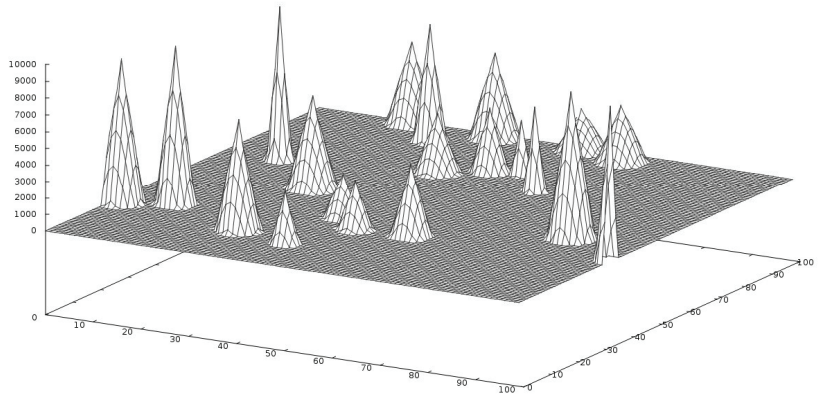


Fig. 3. The image of the sky without reduction

The image of the sky presented in figure 4 is reduced for the observational 'contamination'. The deformation of the stellar flux due to lower sensitivity of the CCD in the middle of the image has vanished after the reduction.

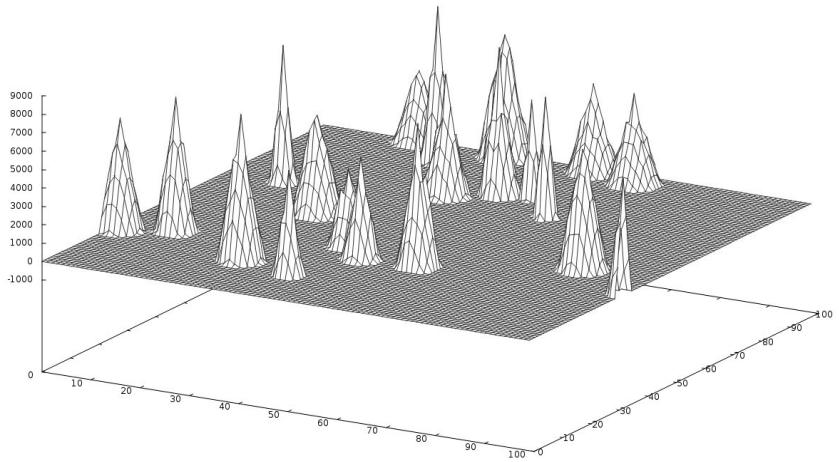


Fig. 4. The preprocessed image of the sky

4. RESULTS OF PERFORMED EXPERIMENTS

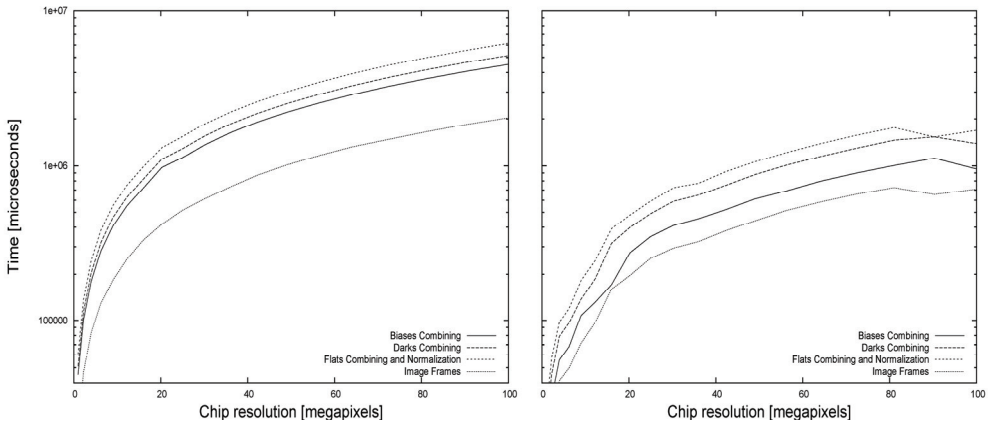
Typically, the CCD cameras that are used in the photometry range from 2–10 MPix, mostly with the side-size as the power-of-two. The survey was done in the range of 1–100 MPix. As the CCDs mainly return the array of photon counts and that is why the model frames were based on the integer type as well.

The experiments have been conducted for the 5, 10, and 15 additional frames (biases, darks and flats) to check the behavior of proposed parallel algorithm under heavy computations. The change of execution time for sequential and parallel algorithms as a function of the image size for 5 additional frames is shown in the figure 5. The overall execution time does not exceed 2 sec. even for the size of 100 MPix that is an extreme situation.

The next set of experiments have been conducted for the frames count 10 for bias-, dark-, and flat-frames. The obtained results are similar to observed before. Figure 6 shows the execution time of these experiments (Fig 6b) in comparison to similar experiments performed in the sequential manner (Fig 6a). The parallel implementation stays below 3 sec. even for the 100 MPix frames.

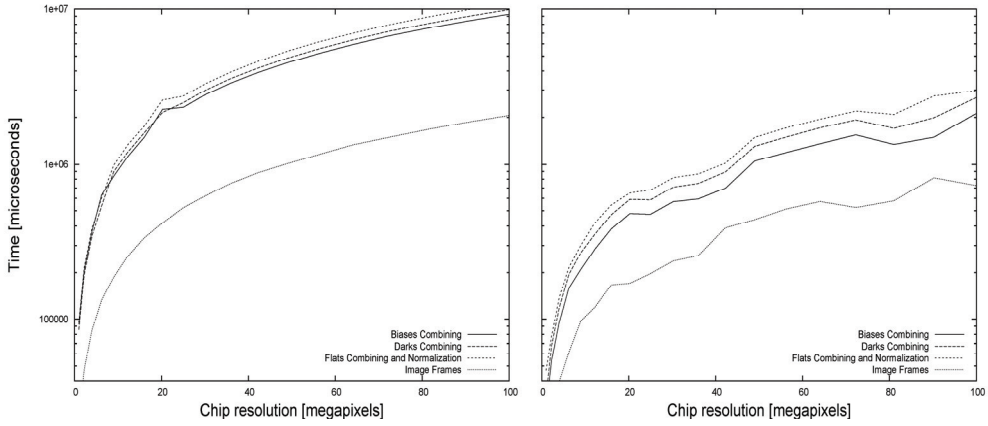
Finally, the experiments have been performed the frames count 15 for the bias-, dark-, and flat-frames. The results of experiments are shown in figure 7. The behavior of algorithm is similar to the two previous series of experiments, as shown in the figure 7a and 7b.

In all experiments the speedup received using GPGPU is clearly seen. The speedup comparison is shown in the figure 8. The heavier in data is the simulation the better performance can be observed with the usage of GPGPU. It can be observed when each frame size grows as well as when the data abundance increases.



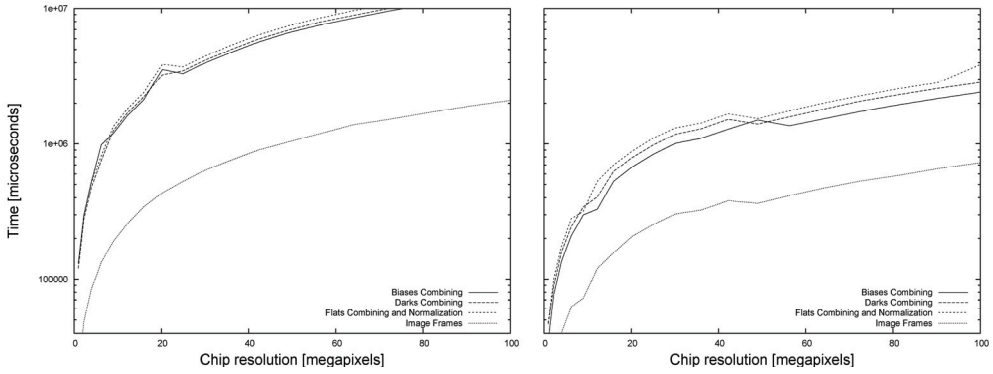
(a) sequential algorithm (b) parallel algorithm

Fig. 5. The execution time for the 5 auxiliary bias-, dark-, and flat-frames.



(a) sequential algorithm (b) parallel algorithm

Fig. 6. The execution time for the 10 auxiliary bias-, dark-, and flat-frames



(a) sequential algorithm (b) parallel algorithm

Fig. 7. The execution time for the 15 auxiliary bias-, dark-, and flat-frames

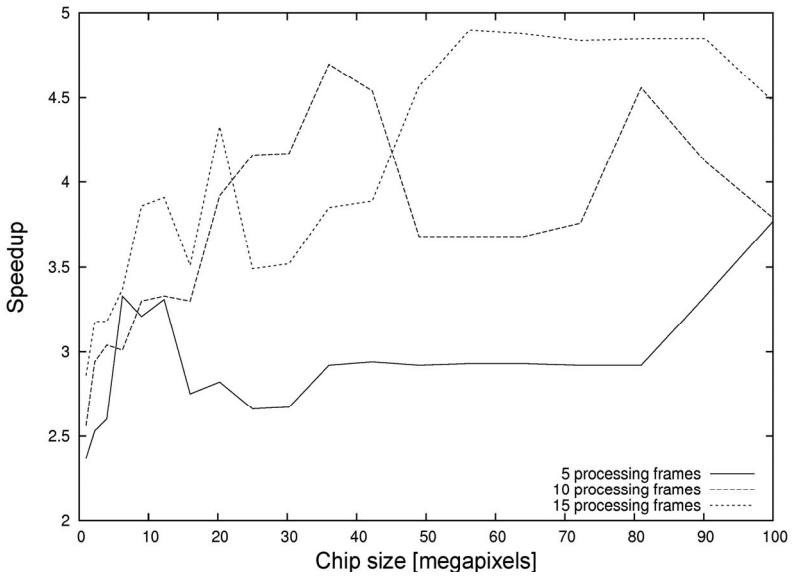


Fig. 8. The speedup as a function of chip size for the series of experiments for 5-, 10-, and 15- auxiliary frames

The main advantage in the use of graphical processing units comes when the data moving takes minor time comparing to data processing. For that each processing method, i.e. combining, subtraction, normalization and reduction is implemented as a whole in the GPU regime. Eventually, for 5 auxiliary frames the total speedup, which is the time to reduce one frame, remains around the value 3, and grows for 10 frames to 3,5 reaching for the 15 frames the value above 4,5. It is important to notice that even for small chip sizes, the library provides the speedup no less than 2, which

makes it also a reasonable choice even for a regular camera sizes when the data acquired each observational night is abundant, that is for the image integration times ranging from seconds to minutes.

5. CONCLUSION

The data gathered from the observations, to be scientifically valuable must undergo a reduction and analysis process. For that the technological advances must be followed by the software applications, aiding astronomers in the reduction process and providing the basis for analysis.

The implementation of the astronomical photometric data reduction on graphical processing units presented in this chapter provides a fast and easy way of reducing observational images. To fully cover reduction process of the gathered data, other observational signatures must be taken into account, e.g. cosmic rays and bad pixels reduction, that is why this work sets a starting point for high-technology data-intensive astrophysical observations' reduction.

The project is in current study, the first results are very promising and shows that using graphical processing units can be answer on challenges that of data reduction. The future works will be concentrate on improving the parallelization process of data reduction.

ACKNOWLEDGEMENTS

The research presented in this paper has been partially supported by the European Union within the European Regional Development Fund program no. POIG.01.03.01-00-008/08.

REFERENCES

[1] BUDDING E., DEMIRCAN O., *Introduction to Astronomical Photometry*, Cambridge University Press, 2007.
[2] FARBER R., *CUDA Application Design and Development*, Elsevier Inc., 2011.
[3] GASTER B.R., HOWES L., *Heterogeneous Computing with OpenCL*, Elsevier Inc., 2012.
[4] HWU W.W., *GPU Computing Gems. Emerald Edition*, Elsevier Inc., 2011.
[5] KUBIAK M., *Gwiazdy i materia międzygwiazdowa*, Wydawnictwo naukowe PWN, 1994.
[6] KIRK D.B., HWU W.W., *Programming Massively Parallel Processors: A Hands-on Approach*, Elsevier Inc., 2010.
[7] KLOTZ. A., MARTINEZ P., *A Practical Guide to CCD Astronomy*, Cambridge University Press, 2000.
[8] MILEONE E.F., STERKEN E., *Astrohomical Photometry: Past, Present, and Future*, Springer Science+Business Media, 2011.

neural networks, CUDA,
parallelization

Dariusz KONIECZNY, Karol RADZISZEWSKI*

EFFICIENCY OF PARALLELIZATION OF NEURAL NETWORK ALGORITHM ON GRAPHIC CARDS

In this paper we are testing the efficiency of parallelization with use of graphic cards. There are many applications where such systems occurs in common, so we choose the domain of artificial neural networks. Actually sold graphic cards gives us strong potential in speeding up calculations and card vendors provide us with even more, giving access to software and documentation, like in CUDA (Compute Unified Device Architecture). But instead of using prepared libraries for algebra, in this work we use the run-time layer of CUDA technology, which gives us more flexibility and almost full control over the hardware. Also, we will show more technical details of implemented algorithms and methods than in other papers regarding this topic. Because of differences in architectures of systems running sequential and parallel versions of applications there was necessity to redefine the original definition of efficiency to compare the heterogeneous systems. We tested our solutions on selected graphics cards with CUDA capability. Input data for neural network which served as benchmark data were global features extracted from histopathological HER-2 images.

1. INTRODUCTION

Neural networks are commonly used in many applications [1], including classification and clustering tasks. Using them gives to the user many upsides, from recognition speed, through often simple implementation, ending at their generalization skills. However, there are also downsides, like the learning process, which lasts long and mostly must be repeated, when learning parameters are about to change. The time

* Institute of Informatics, Technical University of Wrocław, Wybrzeże Wyspiańskiego 27, 50-370 Wrocław.

needed for learning causes simplification of researches and underdeveloped results. That is why shortening of this procedure is so important, and that will be objective of this work.

Neural networks are good material for parallelization process, because their learning algorithm consists of simple operations repeated many times. In those types of tasks good results can be achieved with use of graphic cards, which can do many simple operations at the same time. Presented solutions are not problem-specific, but can be used in many applications which make use of implemented neural networks. In this paper, the example problem is classification and clustering of histopathological images, however quality of results will not be measured – this is not a part of the work, because attention is focused on time-related characteristics.

2. RELATED WORK

Parallelization of neural networks is a vast topic in actual science and its cause is lack of computational power of modern processors. Exponential growth in quantity of processed data and desire to achieve better results gave a perfect opportunity to expansion of parallel and distributed systems. Contemporary works don't even bother creating sequential versions of algorithms, due to their lack of profitability.

Ease of implementation and use of neural networks resulted in many works describing this topic. Parallelized neural networks are used for resolving many practical problems, like face recognition [2], automatic speech recognition [3] and pattern recognition on various types of images, including medical [4][5]. Besides of many types of applications, there are many types of neural network to be parallelized; from simple ones (self-organizing maps), through those of moderate complication level (multilayer networks with various methods of supervised learning), ending at most complicated models, which precisely describe processes occurring in nature (neocognitron, spiking neurons).

Most of researchers depends on provided libraries with implemented linear algebra functions (examples are BLAS and CUBLAS) or prepared extensions for well-known mathematical frameworks (like Matlab and GPUMat or Jacket). Only few works describe the process of parallelization with use of extensions provided for programming languages and results achieved this way. This work describes the results of parallelization process with use of CUDA extension for C language.

3. GPU IMPLEMENTATION OF NEURAL NETWORKS

3.1. GRAPHIC CARDS

Technology used in this work was provided by NVIDIA Corporation, and it is CUDA – Compute Unified Device Architecture, which consists of hardware and software layers.

Hardware is graphic card with CUDA Compute Capability. Its GPU contains big amount of small and simple cores, grouped in multiprocessors. User does not choose specific processor to do the job, but only their quantity and the rest is on built-in governor, which assigns multiprocessors to desired tasks. Cards with better (higher) Compute Capability can use more modern solutions provided by software. There are also various types of memory which user can utilize, for example global or shared, each one of them has different characteristics and should be used in different situations.

Software gives to the user freedom of choice if it comes to programming method. There are three layers, which differs in difficulty and control over the hardware. Easiest way is to use the ready algebra of Fast Fourier Transform libraries which have similar interface to the ones known from contemporary used sequential libraries (CUBLAS, CUFFT). Next is the run-time layer, which utilizes C language with extensions made by NVIDIA and gives to the user more control over hardware, but its counterpart is needed knowledge about hardware, which have to be more precise. Last layer is the driver layer, which gives full potential in GPU programming, but requires from user a deep knowledge about graphic card and more time. In this work the second, run-time layer was chosen to create parallel implementations of two neural networks. Sequential versions were implemented in pure C.

3.2. SELECTED NEURAL NETWORKS

Two different neural networks were implemented in this work. First one is self-organizing map – SOM, and the second one is multi-layer perceptron – MLP. Those networks were chosen because of well-known mathematical basis and many available works regarding their construction and learning algorithms. Main differences between them are their structure and learning method, unsupervised in case of SOM and supervised in MLP.

3.3. PARALLELIZATION PROCESS

Parallelization of algorithms has its own methodologies and one of them, used in this work, was APOD. This name is abbreviation of the words: Analyze, Parallelize, Optimize and Deploy whose mean particular steps in software production cycle and are further described in [6].

The learning algorithm for SOM network consists of steps repeated for every pattern in every epoch. Those steps was divided to three sub-operations and each of them was parallelized as a separate function (kernel in CUDA terminology). Those sub-operations are described below:

a) First step is to compute the distance between weight vector of neuron and learning pattern. Threads was grouped in two-dimensional structure corresponding to map of neurons, so each thread computes distance of one own neuron. Learning patterns are loaded to shared memory, what increase the speed of operations with their use. The access to global memory must be coalesced, so theoretically each threadcomputespart of distance for many neurons, not only his own.
b) Second step is to choose the best matching neuron (one with smallest distance). This is typical reduction and in contrast to technologies like MPI (Message Passing Interface), CUDA cards don't have built-in mechanism for this types of tasks. Instead, divide and conquer method was used, described further in [7].
c) Last step is modification of neurons weights with method Winner Takes Most. Best neuron from previous point should have higher degree of changes than neuronsdistant from him. It is solved by using the diminishing neighborhood function. Analogically to point a), each thread in two-dimensional array has corresponding neuron and computes its change. Critical in this step is to minimalize quantity of accesses to global memory to one read and one write, there is no sense in use of shared memory because each weight is used only once.

The quantity of threads in each case is equal to the quantity of neurons in map.

Second algorithm, back propagated learning of MLP network also consists of operations repeated for every pattern and they were divided into four parts. But, additionally, distinction of hidden and output layer operations was provided. It is because of few assumptions: in most tasks, one hidden layer is sufficient; usually there is only a few neurons in output layer (N_{Out}); and hidden layer usually has more neurons (N_{Hid}) than network has inputs (N_{Inp}). Remembering those assumptions, following functions were created:

a) Forward propagation of hidden layer, where one thread has one hidden layer neuron assigned. Structure of thread grid is one-dimensional, because there is no sense in using more dimensions like in SOM network. Very important is in-

dexing of tables which contains weights, reads must be coalesced to get the highest efficiency. Quantity of threads is set to N_{Hid}.

b) Forward propagation of output layer, divided into two sub-operations: multiplication of previous layer output and corresponding weight, where each thread is assigned to one weight of output layer neurons (quantity of threads equals N_{Hid}*N_{Out}); and second stage, where reduction operation is needed to compute outputs of neurons (same amount of threads).
c) Computation of output layer error, where thread is assigned to each output layer neuron and computes his error. There are N_{Out} threads.
d) Backward propagation of error to hidden layer, allocation of threads to neurons is the same as in point a).
e) Change of weights of neurons in hidden layer, assignation again is similar to point a)
f) Weights change in output layer where allocation is like in first phase of forward propagation of output layer (b)), where each thread has assigned one weight from output layer neurons (N_{Out} threads).

MLP network turned out to be more difficult to parallelize, due to its more complicated structure and learning algorithm.

4. EXPERIMENTS

4.1. INPUT DATA

Benchmark data used in NN learning process were extracted from histopathological images of breast cancer. Images were annotated with one of four classes (0, 1, 2 or 3), which shows patients level of HER2 protein overexpression. Sample images are shown on Fig. 1.

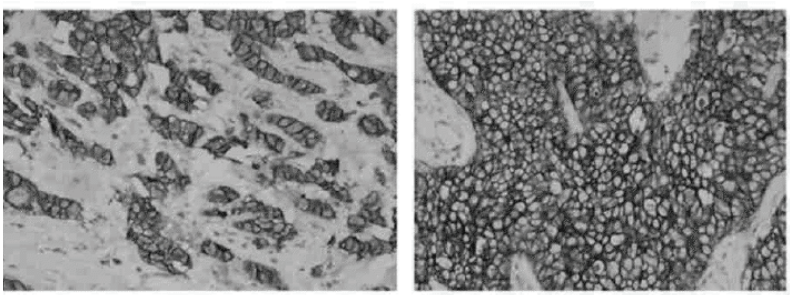


Fig. 1. Sample HER2 images

There was 361 colour images in dataset and from each image feature vector was extracted, composed of following sub-vectors:

a) histogram analysis – 4 features
b) co-occurrence matrix – 440 features
c) run-length matrix features – 20 features

This type of feature vector was proposed by Kostopoulos Spiros in his PhD Thesis [4] and gives 464 floating-point attributes for each image. Features were normalized before the learning process.

4.2. HARDWARE

Experiments were done on following hardware configurations:

a) sequential version – C2D – Core 2 Duo T9400 2,53GHz, one core used, 4GB RAM
b) parallel version 1 – Quadro – NVIDIA Quadro FX3700M 1GB
c) parallel version 2 – Tesla – NVIDIA Tesla T10 4GB

4.3. RESULTS

Studies investigated only time-related characteristics. Effectiveness of classification and quality of clustering were not analyzed (except for basic tests assuring that both sequential and parallel versions work properly), because it was not part of this work.

For each network in each version time of execution was measured, including the time of copying data to and from graphic card in case of parallel versions. Also, for parallel versions, time of those copying processes was measured, to check their percentage occupation in overall execution time.

Execution consist of one hundred learning epochs and was done for few sizes of problems. In case of SOM network, size of problem is one of map dimensions. Both dimensions in this two-dimensional map are the same, so, for example, when size of problem equals 32, map consist of 1024 neurons. Regarding MLP network, size of problem is the quantity on neurons in hidden layer, because quantities of neurons in input and output layers are predefined by the design of input dataset.

Times measured for SOM network are contained in Tab. 1. More legible than chart of measured times is chart of observable speed-up, shown in Fig. 2. Observable speed-up S_O is calculated as follows:

$$S_O = \frac{T_S}{T_P} \tag{1}$$

where

T_S is time of sequential version execution

T_P is time of parallel version execution

Table 1. Execution times in seconds, SOM network

	32	48	64	96	128	160	192
C2D	55.85	129.46	239.18	570.13	1008.99	1577.54	2302.48
Quadro	19.24	22.63	36.62	82.11	153.41	241.11	330.56
Tesla	26.15	28.55	34.19	66.20	112.50	167.46	248.38

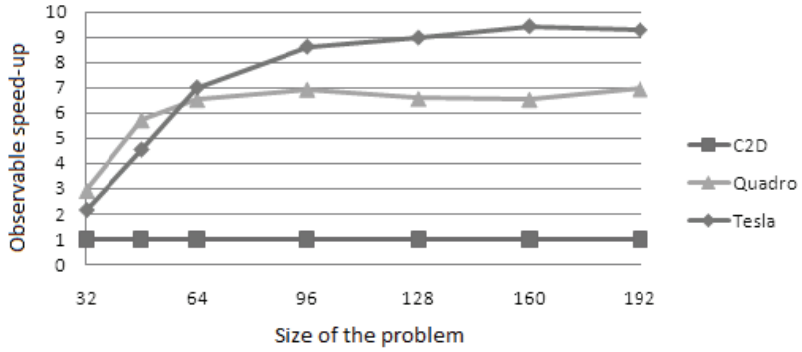


Fig. 2. Observable speed-up, SOM network

The part of execution time occupied by copy operations turned out insignificant, as shown in Fig. 3, but one thing must be remembered. Time of copying is not the only overhead provided by use of CUDA technology, there are also additional operations, like switching the execution flow from CPU to GPU and back. However, copy operations are the most significant part of mentioned overhead and the other part is too hard to measure.

Next step was to measure times of execution for MLP network, which are contained in Tab. 2. Chart of observable speedup is shown in Fig. 4.

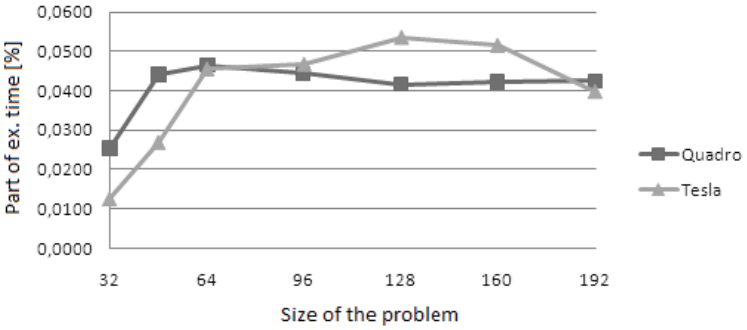


Fig. 3. Time used by copy operations, SOM network

Table 2. Execution times in seconds, MLP network

	32	64	256	1024	4096	8192	16384
C2D	2.16	4.06	16.12	64.67	274.60	558.14	1117.39
Quadro	49.50	50.06	69.02	116.55	119.17	122.18	242.40
Tesla	56.99	57.46	74.12	123.18	123.69	124.50	236.58

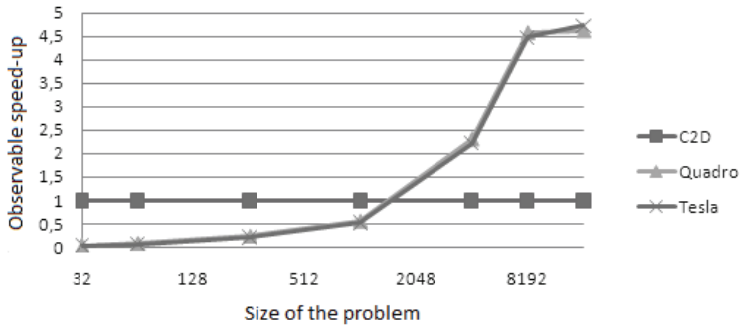


Fig. 4. Observable speed-up, MLP network

Difference in speed-up between both graphic cards is not visible for those sizes of problem, so additional measures was made. Time of sequential execution was predicted based on those measured before. Times of parallel version execution were measured up to size of the problem equal to 131072. For this size of problem the speedup of Tesla card was almost **9-times**, when Quadro card did not increase its speed-up and it stayed at the same level (5-times). Copy operations time in case of MLP network turned out to be insignificant, as it can be seen in Fig. 5. Investigation included sizes of problem up to 131072.

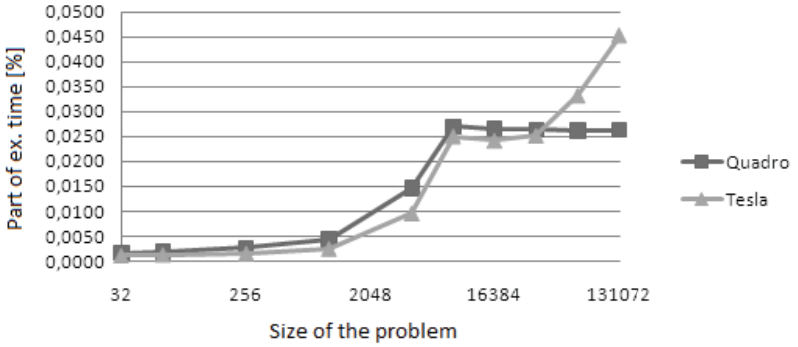


Fig. 5. Time used by copy operations, MLP network

5. CONCLUSIONS

Investigations in this work concerned the time-related characteristics of two parallelized neural network learning algorithms. Experiments were made with use of two graphic card created by NVIDIA, one from professional Quadrosegment and other from segment of computational processors – Tesla. Both of implementations achieved significant speed-up on both cards, 7 and 10 times in case of SOM network and 5 and 9 in case of more complicated MLP network.

It is unprofitable to achieve similar results in sequential computations and it can be done with actual and widely available equipment, even without full studies on CUDA or other general purpose GPU technology. More knowledge in this topic would generate even better results, because of many unused technical possibilities.

REFERENCES

[1] MARKOWSKA-KACZMAR U., KWAŚNICKA H. (red.), *Sieci neuronowe w zastosowaniach*, Oficyna Wydawnicza PWr. Wrocław 2005.

[2] POLI G., SAITO J.H., MARI J.F., ZORZAN M.R., *Processing neocognitron of face recognition on high performance environment based on GPU with CUDA architecture*. In: Computer Architecture and High Performance Computing, 2008. SBAC-PAD ’08. 20th International Symposium on, 81–88, 2008.

[3] ARRIOLA Y., CARRASCO R.A., *Parallel algorithms for automatic speech recognition*. In: Techniques for Speech Processing, IEE Colloquium on, 7/1–7/6, 1990.

[4] SPIROS K.. *Development of supervised and unsupervised pixel-based classification methods for medical image segmentation*. PhD thesis, University of Patras, 2009.
[5] XIE E., McGINNITY M., WU Q-X, CAI J., CAI R. *GPU implementation of spiking neural networks for color image segmentation*. In Image and Signal Processing (CISP), 2011, 4th International Congress on, Vol. 3, 1246 –1250, 2011.
[6] NVIDIA Corporation. *CUDA C Best Practices Guide*, 2012.
[7] HARRIS M.. *Optimizing Parallel Reduction in CUDA*. NVIDIA Corporation.

multiprocessing computer system,
programs scheduling, resources allocation

Zbigniew BUCHALSKI*

PROGRAMS SCHEDULING IN MULTIPROCESSING COMPUTER SYSTEM WITH POSITION DEPENDENT PROCESSING TIMES

The paper presents results of research on the problem of time-optimal programs scheduling and primary memory pages allocation in multiprocessing computer system. We consider an multiprocessing computer system consisting of m parallel processors, common primary memory and external memory. The primary memory contains N pages of identical capacity. This system can execute n independent programs. Because our problem belongs to the class of *NP*-complete problems we propose an heuristic algorithm to minimize schedule length criterion, which employs some problem properties. Some results of executed computational experiments for basis of this heuristic solution procedure are presented.

1. INTRODUCTION

In last years the time-optimal problems of tasks scheduling and resources allocation are intensive developing [4, 6, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19]. Scheduling problems can be understood very broadly as the problem of the allocation of resources over time to perform a set of tasks. By resources we understand arbitrary means tasks compete for. They can be of a very different nature, e.g. energy, tools, money, manpower. Tasks can have a variety of interpretation starting from machining parts in manufacturing systems up to processing information in computer systems. The further development of the research has been connected with applications, among other things in multiprocessing computer systems [1, 2, 3, 5, 7, 20].

* Institute of Computer Engineering, Control and Robotics, Wrocław University of Technology, Wybrzeże Wyspiańskiego 27, 50-370 Wrocław; e-mail: zbigniew.buchalski@pwr.wroc.pl

In multiprocessing computer systems is usually used common primary memory with limited capacity and external memory. The external memory has significantly longer access time and this is why minimization of the number of demands to the external memory during programs processing is necessary.

In this paper the problem optimization of programs scheduling and optimal allocation of primary memory pages to the processors are considered. These programs scheduling and primary memory pages allocation problems are very complicated problems and belongs to the class of *NP*-complete problems. Therefore in this paper we propose an heuristic algorithm for solving of a optimization problem. In the second section formulation of optimization problem is presented. In the third section an heuristic algorithm is given and in the fourth section several experimental results on the base this heuristic algorithm are presented. Last section contains final remarks.

2. DESCRIPTION OF THE PROBLEM

We consider an multiprocessing computer system (as shown in Fig.1) containing m processors, common primary memory and external memory. This system can execute n independent programs.

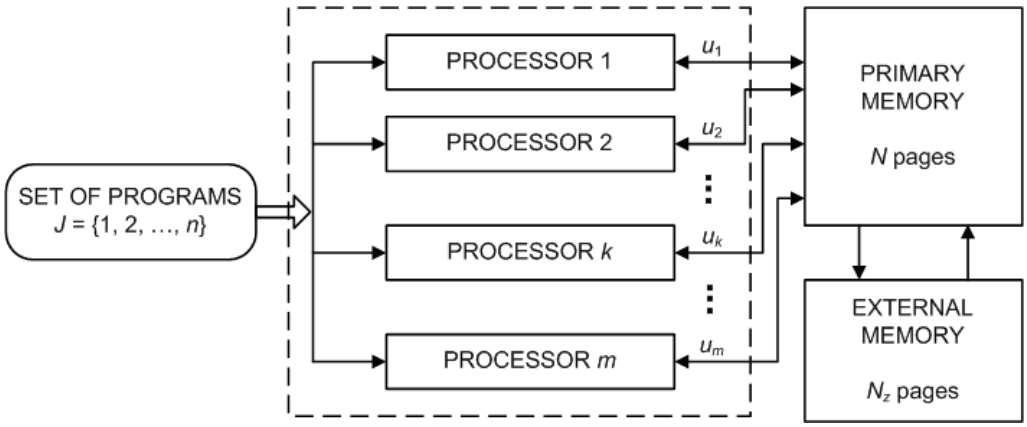


Fig. 1. Multiprocessing computer system

This system can execute n independent programs. We assume about this system, that it is paged virtual memory system and that:

- the primary memory contains N pages of identical capacity,
- each the processor has access to every one of N primary memory pages and may execute every one of n programs,

- the external memory contains N_z pages (the external memory pages capacity is equal to the primary memory pages capacity), $N_z > N$,
- during execution of all n programs, the number of u_k primary memory pages is allocated to the k-th processor; $\sum_{k=1}^{m} u_k \le N$. Each processor may use only allocated to him the primary memory pages.

Let $J = \{1,2,...,n\}$ be the set of programs, $U = \{1,2,...,N\}$ – set of primary memory pages, P denotes the set of processors $P = \{1,2,...,m\}$. Processing time of i-th program on k-th processor is given by following function:

$$T_i(u_k, k) = a_{ik} + \frac{b_{ik}}{u_k}, \quad u_k \in U, \quad 1 \le k \le m, \quad i \in J, \tag{1}$$

where $a_{ik} > 0$, $b_{ik} > 0$ – parameters characterized i-th program and k-th processor.

This programs scheduling and primary memory pages allocation problem in multi processing computer system can be formulated as follows: find scheduling of n independent programs on the m processors running parallel and partitioning of N primary memory pages among m processors, that schedule length criterion is minimized.

Let $J_1, J_2, ..., J_k, ..., J_m$ be defined as subsets of programs, which are processing on the processors 1, 2,..., k,..., m. The problem is to find such subsets $J_1, J_2, ..., J_k, ..., J_m$ and such pages numbers $u_1, u_2, \ldots, u_k, \ldots, u_m$, which minimize the T_{opt} of all set J:

$$T_{opt} = \min_{\substack{J_1, J_2, ..., J_m \\ u_1, u_2, ..., u_m}} \max_{1 \le k \le m} \left\{ \sum_{i \in J_k} T_i(u_k, k) \right\} \tag{2}$$

under the following assumptions:

$$(i) \quad J_s \cap J_t = \varnothing, \quad s, t = 1,2,..., m, \quad s \ne t, \quad \bigcup_{k=1}^{m} J_k = J,$$

$$(ii) \quad \sum_{k=1}^{m} u_k \le N, \quad u_k \in U, \quad k = 1,2,..., m,$$

$$(iii) \quad u_1, u_2, ..., u_m - \text{positive integer.}$$

The assumption (iii) is causing, that the stated problem is very complicated therefore to simplify the solution our problem we assume in the sequel that primary memory pages

are continuous. The numbers of pages obtained by this approach are rounded to the integer numbers (look **Step 12** in the heuristic algorithm) and finally our problem can formulated as following minimizing problem:

$$T_{opt} = \min_{\substack{J_1,J_2,\ldots,J_m \\ u_1,u_2,\ldots,u_m}} \max_{1\le k\le m} \left\{ \sum_{i\in J_k} \widetilde{T}_i(u_k,k) \right\} \tag{3}$$

under the following assumptions:

$$(i) \quad J_s \cap J_t = \varnothing, \quad s,t = 1,2,\ldots,m, \quad s \neq t, \quad \bigcup_{k=1}^{m} J_k = J,$$

$$(ii) \quad \sum_{k=1}^{m} u_k \le N, \quad u_k \ge 0, \quad k = 1,2,\ldots,m,$$

where $\widetilde{T}_i : [0,N] \times \{1,2,\ldots,m\} \to R^+$ is the extension of function $T_i : \{1,2,\ldots,N\} \times \times \{1,2,\ldots,m\} \to R^+$ and formulated by function:

$$\widetilde{T}_i(u_k,k) = a_{ik} + \frac{b_{ik}}{u_k}, \quad u_k \in [0,N], \quad 1 \le k \le m, \quad i \in J. \tag{4}$$

Taking into account properties of the function $\widetilde{T}_i(u_k,k)$, it is easy to show the truth of the following theorem:

Theorem 1.
If the sets $u_k^*, J_k^*, k = 1,2,\ldots,m$ are a solutions of minimizing problem (3), then:

$$(i) \quad \sum_{k=1}^{m} u_k^* = N; \quad u_k^* > 0, \quad k : J_k^* \neq \varnothing, \quad k = 1,2,\ldots,m;$$

$$u_k^* = 0, \quad k : J_k^* = \varnothing, \quad k = 1,2,\ldots,m;$$

$$(ii) \quad \sum_{i\in J_k^*} \widetilde{T}_i(u_k^*,k) = \text{const}, \quad k : J_k^* \neq \varnothing, \quad k = 1,2,\ldots,m.$$

We define function $F(J_1 J_2,\ldots,J_m)$, which value is solution following system of equations:

$$\begin{cases} \sum_{i\in J_k} a_{ik} + \dfrac{\sum_{i\in J_k} b_{ik}}{u_k} = F\left(J_1, J_2, \ldots, J_m\right), \quad k : J_k \neq \varnothing, \quad k = 1,2,\ldots,m \\ \sum_{k:J_k\neq\varnothing} u_k = N; \quad u_k > 0 \qquad\qquad k : J_k \neq \varnothing, \quad k = 1,2,\ldots,m \end{cases} \tag{5}$$

On the basis of **Theorem 1** and (5), problem (3) will be following:

$$T_{opt} = \min_{J_1, J_2, \ldots, J_m} F(J_1, J_2, \ldots, J_m) \tag{6}$$

under the assumptions:

$$(i) \quad J_s \cap J_t = \varnothing; \quad s,t = 1,2,\ldots,m, \quad s \neq t$$

$$(ii) \quad \bigcup_{k=1}^{m} J_k = J; \quad k = 1,2,\ldots,m,$$

If $J_1^*, J_2^*, \ldots, J_m^*$ are solutions of problem (6), it u_k^*, J_k^* $k = 1, 2, \ldots, m$ are solutions of problems (3), where:

$$u_k^* = \begin{cases} \dfrac{\sum_{i\in J_k^*} b_{ik}}{F(J_1^*, J_2^*, \ldots, J_m^*) - \sum_{i\in J_k^*} a_{ik}}; & k : J_k^* \neq \varnothing, \quad 1 \le k \le m, \\ 0 \qquad\qquad\qquad\qquad ; & k : J_k^* = \varnothing, \quad 1 \le k \le m\,. \end{cases} \tag{7}$$

3. THE HEURISTIC ALGORITHM

We assume that the first processor from the set P has highest speed and the last processor from the set P has least speed. We assume also if be of assistance in pages allocation so-called partition of pages coefficient α; $\alpha > 1$. To the last m processor is allocated u_m pages according to the following formula:

$$u_m = \frac{N}{1 + \sum_{k=1}^{m-1} \left[(m-k)\cdot\alpha\right]} \tag{8}$$

To the remaining processors are allocated pages according to the formula:

$$u_k = (m-k)\cdot\alpha\cdot u_m; \quad k = 1,2,...,m-1. \tag{9}$$

The proposed heuristic algorithm is as follows:

Step 1. For given $u_k = \frac{N}{m}$ and random generate parameters a_{ik}, b_{ik} calculate the processing times of programs $T_i(u_k, k)$ according to the formula (1).

Step 2. Schedule programs from longest till shortest times $T_i(u_k, k)$ and formulate the list L of these programs.

Step 3. Calculate mean processing time T_{mean} every processors according to follows formula:

$$T_{mean} = \frac{\sum_{i=1}^{n} T_i(u_k, k)}{m}; \quad i \in J, \quad k \in P, \quad u_k = \frac{N}{m}.$$

Step 4. Schedule first m longest programs from the list L to the succeeding m processors from first processor to the m-th processor and eliminate these programs from the list L.

Step 5. Schedule in turn shortest programs from the list L to the succeeding processor from first processor to the m-th processor for the moment, when the sum of processing times these programs to keep within the bounds of time T_{mean} and eliminate these programs from the list L. If list L is not empty go to the next step, if is empty go to the **Step 7**.

Step 6. Remainder of programs in the list L schedule to the processors according to the algorithm *LPT* (Longest Processing Time) to moment of finish the list L.

Step 7. Calculate total processing time T_{opt} of all programs for scheduling $J_1, J_2, \ldots, J_m$, which was determined in the **Steps 3÷6** and for given numbers of pages $u_k = \frac{N}{m}$.

Step 8. For given partition of pages coefficient α allot pages u_k, $k \in P$ to succeeding processors as calculated according formula (8) and (9).

Step 9. For programs scheduling which was determined in **Steps 3÷6** and for numbers of pages u_k, $k \in P$ allotted to processors in the **Step 8** calculate total processing time T_{opt} of all programs.

Step 10. Repeat the **Step 8** and **Step 9** for the next seven augmentative succeeding another values of coefficient α.

Step 11. Compare values of total processing times T_{opt} of all programs calculated after all samples with different values of coefficient α (**Steps 8÷10**). Take this coefficient α when total processing time T_{opt} of all programs is shortest.

Step 12. Find the discrete numbers $\hat{u}_k$ of pages, k = 1, 2, ..., m according to follows dependence:

$$\hat{u}_{\beta(k)} = \begin{cases} \lfloor u_{\beta(k)} \rfloor + 1 & ; \quad k = 1,2,...,\Delta \\ \lfloor u_{\beta(k)} \rfloor & ; \quad k = \Delta+1, \Delta+2, ..., m \end{cases}$$

where $\Delta = N - \sum_{j=1}^{m} \lfloor u_j \rfloor$ and β is permutation of elements of set P ={1, 2, ..., m} such, that $u_{\beta(1)} - \lfloor u_{\beta(1)} \rfloor \geq u_{\beta(2)} - \lfloor u_{\beta(2)} \rfloor \geq \ldots \geq u_{\beta(m)} - \lfloor u_{\beta(m)} \rfloor$.

4. COMPUTATIONAL EXPERIMENTS

On the base this heuristic algorithm were obtained results of computational experiments for eight another values of coefficient α = 4, 8, 12, … , 32. For the definite number of programs n = 70, 140, 210, 280, 350, number of processors m = 5, 10, 15, 20, 25, 30 and number of primary memory pages N =10.000 were generated parameters a_{ik}, b_{ik} from the set {0.2, 0.4, ... , 9.8, 10.0}. For each combination of n and m were generated 40 instances. The results of comparative analysis of heuristic algorithm proposed in this paper and the algorithm *LPT* are showed in the Table 1.

Table 1. The results of comparative analysis of heuristic algorithm and algorithm *LPT*

n/m	number of instances, when:			Δ^H	S^H	S^{LPT}
	$T_{opt}^H < T_{opt}^{LPT}$	$T_{opt}^H = T_{opt}^{LPT}$	$T_{opt}^H > T_{opt}^{LPT}$	%	sec	sec
70/5	20	0	20	1,8	2,7	2,0
140/5	21	1	18	2,1	4,9	3,7
210/5	23	1	16	3,5	9,3	7,9
280/5	22	1	17	4,7	13,6	11,2
350/5	24	2	14	5,9	16,1	12,1
70/10	21	0	19	2,3	3,1	2,6
140/10	23	1	16	3,3	5,9	4,7
210/10	23	2	15	3,8	10,8	8,9

280/10	25	2	13	4,4	15,1	13,5
350/10	26	1	13	5,6	18,2	15,4
70/15	19	1	20	2,4	3,6	3,2
140/15	21	0	19	3,7	8,9	7,3
210/15	22	1	17	4,5	12,5	10,4
280/15	23	2	15	4,9	16,7	13,6
350/15	25	2	13	5,8	18,7	15,2
70/20	20	0	20	2,1	4,6	3,8
140/20	22	1	17	3,7	9,4	7,8
210/20	23	1	16	4,8	14,1	10,9
280/20	25	1	14	5,6	18,4	15,7
350/20	28	1	11	6,0	20,2	17,8
70/25	20	1	19	2,5	5,9	4,5
140/25	22	0	18	3,9	8,8	6,9
210/25	24	1	15	4,8	12,8	11,5
280/25	25	0	15	5,5	17,8	15,6
350/25	26	1	13	6,6	20,9	18,2
70/30	21	0	19	2,9	6,8	5,9
140/30	23	1	16	3,9	9,9	8,1
210/30	24	2	14	5,1	14,0	12,5
280/30	27	2	11	6,4	18,6	16,5
350/30	29	2	9	8,1	21,8	20,2

In the Table 1 there are the following designations:

n – number of programs,

m – number of processors,

T_{opt}^{H} – total processing time of all set of programs J for the heuristic algorithm,

T_{opt}^{LPT} – total processing time of all set of programs J for the algorithm LPT,

Δ^{H} – the mean value of the relative improvement T_{opt}^{H} in relation to T_{opt}^{LPT}:

$$\Delta^{H} = \frac{T_{opt}^{LPT} - T_{opt}^{H}}{T_{opt}^{H}} \cdot 100\%,$$

S^{H} – the mean time of the numerical calculation for the heuristic algorithm,

S^{LPT} – the mean time of the numerical calculation for the algorithm LPT.

5. FINAL REMARKS

Computational experiments presented above show, that quality of programs scheduling in parallel multiprocessing computer system based on the proposed in this paper heuristic algorithm increased in compare with simple *LPT* algorithm. The few percentages improvement of time T^H in compare with T^{LPT} can be the reason why heuristic algorithms researches will be successfully taken in the future.

Application of presented in this paper heuristic algorithm is especially good for multiprocessing computer systems with great number of programs because in this case the Δ^H improvement is the highest. Proposed heuristic algorithm can be used not only to programs scheduling in multiprocessing computer systems but also to task scheduling in parallel machines or even to operations scheduling in workplaces equipped with production machines.

REFERENCES

[1] BIANCO L., BŁAŻEWICZ J., DELL'OLMO P., DROZDOWSKI M., *Preemptive scheduling of multiprocessor tasks on the dedicated processors system subject to minimal lateness*. Information Processing Letters, 46, 1993, 109–113.

[2] BIANCO L., BŁAŻEWICZ J., DELL'OLMO P., DROZDOWSKI M., *Linear algorithms for preemtive scheduling of multiprocessor tasks subject to minimal lateness,* Discrete Applied Mathematics, 72, 1997, 25–46.

[3] BŁAŻEWICZ J., DROZDOWSKI M., WERRA D., WĘGLARZ J., *Scheduling independent multiprocessor tasks before deadlines*. Discrete Applied Mathematics 65 (1–3), 1996, 81–96.

[4] BŁAŻEWICZ J., ECKER K., SCHMIDT G., WĘGLARZ J., *Scheduling in Computer and Manufacturing Systems*. Springer-Verlag, Berlin–Heidelberg, 1993.

[5] BŁAŻEWICZ J., LIU Z., *Scheduling multiprocessor tasks with chain constraints*. European Journal of Operational Research, 94, 1996, 231–241.

[6] BOCTOR F., *A new and efficient heuristic for scheduling projects will resources restrictions and multiple execution models*. European Journal of Operational Research, vol. 90, 1996, 349–361.

[7] BRAH S.A., LOO L.L., *Heuristics for scheduling in a flow shop with multiple processors*, European Journal of Operational Research, Vol. 113, No. 1, 1999, 113–122.

[8] BUCHALSKI Z., *Application of heuristic algorithm for the tasks scheduling on parallel machines to minimize the total processing time*. Proceedings of the 15th International Conference on Systems Science , vol. 2, Wrocław, 2004.

[9] BUCHALSKI Z., *Minimising the Total Processing Time for the Tasks Scheduling on the Parallel Machines System*. Proc. of the 12th IEEE International Conference on Methods and Models in Automation and Robotics, Domek S., Kaszyński R. (Eds.), Międzyzdroje, Poland, MMAR 2006, 28–31 August 2006, 1081–1084.

[10] CHENG J., KARUNO Y., KISE H., *A shifting bottleneck approach for a parallel-machine flow-shop scheduling problem*, Journal of the Operational Research Society of Japan, Vol. 44, No. 2, 2001, 140–156.

[11] GUPTA J.N.D., HARIRI A.M.A., POTTS C.N., *Scheduling a two-stage hybrid flow shop with parallel machines at the first stage*, Annals of Operations Research, Vol. 69, No. 0, 1997, 171–191.

[12] JANIAK A., KOVALYOV M., *Single machine scheduling subject to deadlines and resources dependent processing times*. European Journal of Operational Research, , vol. 94, 1996, 284–291.
[13] JÓZEFCZYK J., *Task scheduling in the complex of operation with moving executors*, Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław, 1996 (in Polish).
[14] JÓZEFCZYK J., *Selected Decision Making Problems in Complex Operation Systems*, Monografie Komitetu Automatyki i Robotyki PAN, t. 2, Oficyna Wydawnicza Politechniki Wrocławskiej, Warszawa–Wrocław, 2001 (in Polish).
[15] JÓZEFOWSKA J., MIKA M., RÓŻYCKI R., WALIGÓRA G., WĘGLARZ J., *Discrete-continuous scheduling to minimize maximum lateness*, Proceedings of the Fourth International Symposium on Methods and Models in Automation and Robotics MMAR'97, Międzyzdroje, Poland, 1997, 947–952.
[16] JÓZEFOWSKA J., MIKA M., RÓŻYCKI R., WALIGÓRA G., WĘGLARZ J., *Local search metaheuristics for discrete-continuous scheduling problems*, European Journal of Operational Research, 107, 1998, 354–370.
[17] JÓZEFOWSKA J., WĘGLARZ J., D*iscrete-continous scheduling problems – mean completion time result*, European Journal of Operational Research, vol. 94, No. 2, 1996, 302–310.
[18] JÓZEFOWSKA J., WĘGLARZ J., *On a methodology for discrete-continous scheduling*, European Journal of Operational Research, Vol. 107, No. 2, 1998, 338–353.
[19] NOWICKI E., SMUTNICKI C., *The flow shop with parallel machines*. *A Tabu search approach*. European Journal of Operational Research 106, 1998, 226–253.
[20] WĘGLARZ J., *Multiprocessor scheduling with memory allocation – a deterministic approach.* IEEE Trans. Comput., C-29, 1980, 703–710.

network service distribution,
network service monitoring, quality of services.

Mariusz FRAŚ*

THE ESTIMATION OF REMOTELY MONITORED NETWORK SERVICE EXECUTION PARAMETERS

The chapter presents a mechanism for remote monitoring of network services with use of analysis of service request processing on TCP session level. The presented method permits to estimate values of essential non-functional service parameters such as service execution time on remote server. There are considered synchronous services that are commonly used in SOA-based (Service Oriented Architecture - based) systems. The work also presents results of experiments performed in real environment that show effectiveness of described method.

1. INTRODUCTION

The quality of network services is the key point of interest of service providers and one of the most important areas of research. Intensive development of methods and tools in this area was due to the dissemination of services in the Internet, the development of streaming services (multimedia), and as a result of the increasing use and the growing importance of business solutions built using Web services, including systems based on the paradigm of SOA (Service Oriented Architecture).

Among the quality attributes defined for a SOA system, three of them directly relate to everyday perception of the quality of service for the end user and also apply to other types of Web-based systems [1]. These attributes are: availability, usability, and performance. Commonly used solutions at the service abstraction layer for improving the quality of these attributes are redundancy of services (e.g. CDN solutions) and service requests distribution.

For SOA-type systems solutions concerning the quality of services usually have been developed in the context of Web services, usually proposing useful standards for

* Wrocław University of Technology, Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland.

quality of service mechanisms, such as WS-Policy [2] and WSLA [3]. In work [4] a comprehensive overall infrastructure to guarantee SLA (Service Level Agreement) for services has been proposed, including general scheme of the runtime environment, specifications and procedures for handling requests, and measurement of services. The selection of services in order to ensure their quality is also considered in the context of the composition of complex services from service components (atomic). In work [5] the service selection based on utility function on attributes assigned to services (such as price, availability, reliability and response time) has been proposed. In this work a local algorithm of service selection (i.e. selection of single service) and a global algorithm of building the optimal plan of service execution with use of linear programming has been proposed. In all cases an important component for guaranteeing quality in a SOA systems is the right mechanism to monitor service [4, 6] and knowledge or proper estimation of values of non-functional parameters characterized considered services.

The paradigm of SOA says that given service (more strictly functionally equivalent services) can be achievable from different service providers, and at the same different network locations. Because the effectiveness of service provider's servers as well as its distance to the client (and thus communication cost) can be different, the quality of service delivery (i.e. values of non-functional parameters of the service) can be different too [7]. The problem of service (or server) selection taking into account performance issues concerns many types of information systems and many solutions are based on the use of special component called distributor.

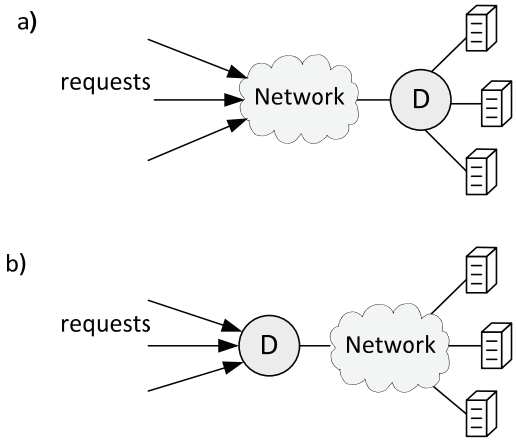


Fig. 1. Request distribution models: a) local, b) global

In general we can distinguish two models of systems which process client requests built with request distribution component (figure 1). Systems with local distributor are characteristic for such systems as local web clusters. The distributor is located at serv-

er's site and closely cooperate with them. The global distribution must take into account possibly different and variable network communication conditions.

The quality of service delivery depends on the state of the processing environment, including the state of the server and result of service execution. Thus it must be accomplished method that permits obtain such essential information. In closely cooperated systems it can be done with direct information exchange. However in loosely cooperated systems such as in SOA-based systems, it may be required to extract some information by monitoring processing and estimating some values of parameters without specific cooperation with other processing components.

For synchronous network services that are commonly found in SOA-based systems an essential non-functional parameter for the assessment of quality of service delivery is the response time. In globally distributed system there is often need to distinguish two parts of this parameter that must be known: request processing time on the server (service execution time) and data transfer time.

2. SERVICE DISTRIBUTION SYSTEM CONCEPT

The concept of effective and quality-aware infrastructure, built in accordance with SOA paradigm, is based on the idea of Virtual Service Delivery System (VSDS) capable to handle client's requests taking into account service instance non-functional parameters. The main component of the system is network service broker (further called Broker), more detailed described in [8]. The main assumptions for Broker operation are:

- the Broker delivers to clients the set of J indivisible basic services (atomic services) as_j, $j \in \langle 1, J \rangle$,
- the Broker knows execution systems es_m, $m \in \langle 1, M \rangle$, where real services (service instances) are available,
- the Broker acts as a service proxy – it hides real service instances, monitors them and distribute client's requests for services to proper instances according to some distribution policy,
- the Broker also supports complex services composed from atomic ones, however this issue is not considered in this work.

The Broker implements the Virtual Service Layer (figure 2). The Virtual Service Layer (VSL) virtualizes real services available on service execution systems (servers) that are hidden from client point of view. Thus, the client deals with virtual service VS_j that is mapped to service instances $is_{j,m}$ on given servers es_m. The client of the system C calls a service visible to it, and the Broker distribute the request to one, chosen service instance.

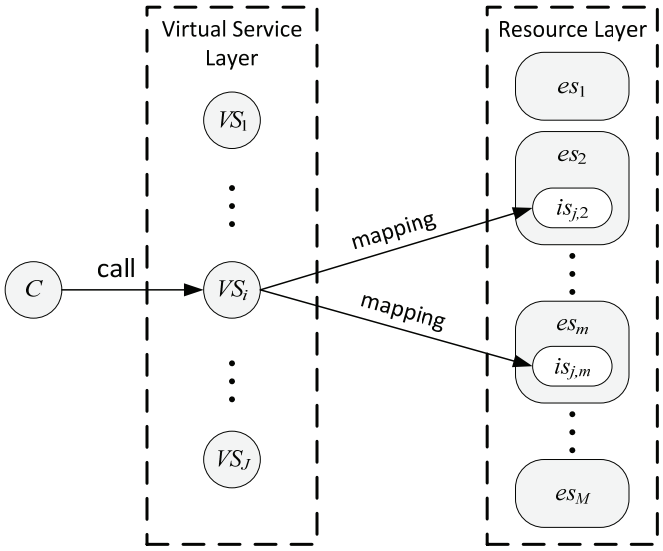


Fig. 2. The layers of Virtual Service Delivery System

Summarizing, the Virtual Service Layer is defined as the four <*ES, CL, AS, IS* >, where:

- $ES = \{es_1,..,es_m,...,es_M\}$ – the set of execution systems es_m, $m \in \langle 1, M \rangle$, M – the number of execution systems,
- $CL = \{cl_1,..,cl_m,...,cl_M\}$ – the set of communication links cl_m to execution systems.
- $AS = \{as_1,..,as_j,...,as_J\}$ – the set of atomic services, where: as_j – j-th atomic service, J – the number of atomic services,
- $IS = \{IS_1,..,IS_j,...,IS_J\}$ – the set of instances of services, where: IS_j – the subset of instances of service as_j , $is_{j,m}$ – m-th instance of j-th service as_j localized in given execution system, M_j – the number of instances of j-th service, $M_j \leq M$.

The instances of given atomic service are functionally the same and differs only in the values of non-functional parameters $\psi(is_{j,m}) = \{\psi^1_{j,m},..,\psi^f_{j,m},...,\psi^F_{j,m}\}$, where $\psi^f_{j,m}$ is f-th non-functional parameter of m-th instance of j-th atomic service.

The quality of delivered services depends on communication link properties and effectiveness of request processing on the server that are expressed by values of service instance parameters. The Broker performs request distribution based on the actual values of these parameters and chosen distribution strategy.

The problem of service request distribution generally can be stated using criterion function Q:

$$is_{j,m^*} \leftarrow \arg\limits_{m} Q(\psi^1_{j,m},...,\psi^f_{j,m},...,\psi^F_{j,m}) \quad (1)$$

It is the task to select such instance is_{j,m^*} to serve request for service as_j that criterion Q is satisfied. The criterion is formulated with use of non-functional parameters of service instance. In the particular case it is the task of finding extreme of the criterion function.

It may be distinguished two kinds of service parameters: static, i.e. constant in long period of time (e.g. service price), and dynamic, i.e. variable in short period of time (e.g. completion time of execution of the service instance). From the client point of view, the most important service parameter is often response time, which is usually very dynamic parameter. In network environment actually it consists of two parts (parameters): data transfer time and execution time on the processing server. In this case the distribution criterion can be formulated as: $is_{j,m^*} \leftarrow \arg\limits_{m} Q(te_{j,m} + tt_{j,m})$, where $te_{j,m}$ is execution time and $tt_{j,m}$ is transfer time. For best effort strategy it will be the task of minimization of the sum of the these times.

In conclusion, the very important function of the Broker is correct evaluation of values of non-functional service instance parameters. This requires suitable monitoring of service execution and proper estimation of values of essential service parameters.

3. SERVICE MONITORING AND ESTIMATION OF VALUES OF PARAMETERS

The architecture of network services broker must take into account requirements for identification of incoming requests, monitoring service execution, estimation of service instance parameters and process of control of service requests distribution according to a given criterion. Evaluation of values of non-functional services parameters is performed with use of three components of the Broker [8, 9]: Service request monitoring module, Network service monitoring module and Estimation module. The Estimation module calculates selected values and statistics characterizing the instances of atomic services. This is performed on the basis of the measured current values of network environment (network links load) and measured values of parameters of service request processing at the Broker. The procedure permits evaluate values of two basic parameters, transfer time and service processing time, without any special cooperation with the server.

Monitoring of the services performed remotely, is based on the analysis of the TCP session. The module measures such time intervals as (figure 3): DNS name resolution time T_{DNS}, the time of establishing session connection (TCP Connect time) T_{TCPC}, the

time to receive the first byte of transferred data T_{FBYTE}, and the total time of the request processing.

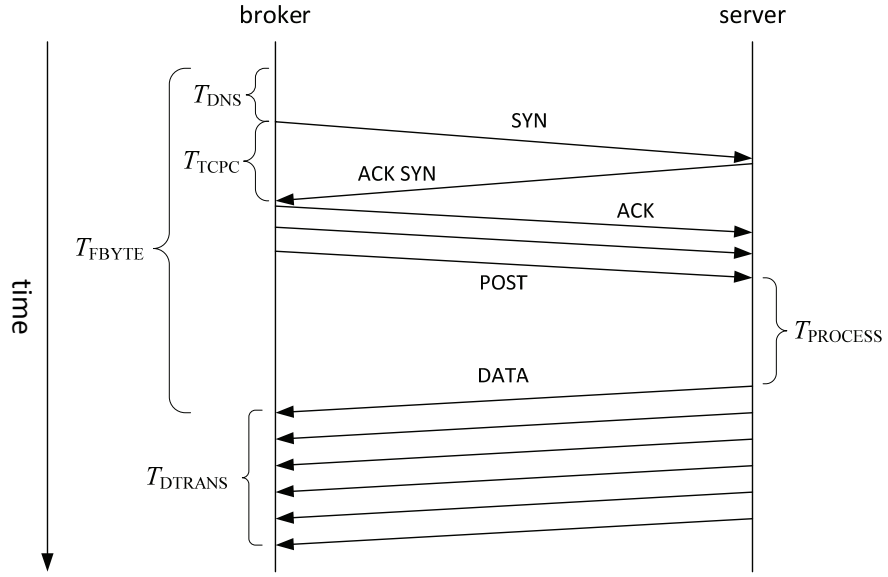


Fig. 3. Monitored intervals of TCP session

The request service time on the execution system is estimated according to the formula (2):

$$T_{\mathrm{PROCESS}} = T_{\mathrm{FBYTE}} - T_{\mathrm{DNS}} - 2 \cdot T_{\mathrm{TCPC}} \tag{2}$$

The formula (2) is can be used when few data is transmitted from the client (the Broker) to the server. It is rather common case. In other case this data transfer time must be also accounted. The data transfer time is the difference of total request processing and the time T_{PROCESS} (i.e. the other components (times) od request processing are included).

The distribution of incoming requests must be performed on the basis of expected values of request processing for all available service instances. These values can be estimated using different forecasting methods utilizing monitored values of previous requests processing. For estimated times $\hat{t}$ (both transfer time and service execution time) the basic methods are:

– moving average of registered times: $\hat{t}^{n}_{j,m} = \frac{1}{L} \cdot \sum_{k=n-1}^{k-L} w_k \cdot t_k$, where: $\hat{t}^{n}_{j,m}$ – forecasted time of *n*-th request served by atomic service instance $is_{j.m}$, L – the

length of the window (number of observed values), w_k – window function, t_k – registered times of previous requests served by instance $is_{j.m}$, n – the index of current request,

- moving median: $\hat{t}_{j,m}^{n} = \text{med}\,(t_{k-1}, t_{k-2}, ..., t_{k-L})$,
- methods based on artificial intelligence approach, e.g. with use of neural networks.

In [9 and 10] the fuzzy-neural controller that models communication link and service instances used for forecasting processing times is presented. Each service instance and each communication link for each service instance, are modeled as two stage fuzzy-neural network with two inputs characterizing the current conditions of modeled entity. An estimated time is an output of the controller. The inputs must be parameters of environment (e.g. communication link) online monitored by the Broker.

4. THE EFFECTIVENESS OF THE ESTIMATION OF THE VALUES OF SERVICE INSTANCE PARAMETERS

For evaluation of the effectiveness of estimation of service request processing times the experiment in real Internet network was performed. The Broker that caries described functionality served a number of clients requesting fixed set of network services. The Broker and clients were located at Wroclaw University of Technology campus. That was established 6 test services running on 6 servers – a total 36 service instances.

Table 1. Services and service instance execution times

Service	Amount of data [kB]	Execution time [s]					
		A	B	C	D	E	F
UA	50	2	2	2	2	2	2
UB	50	4	4	4	3	4	2
UC	100	5	5	4	4	3	3
UD	100	6	4	5	3	4	2
UE	200	6	6	5	5	4	4
UF	200	4	3	3	2	2	1

The service instances differed in basic execution time and amount of data to transfer as shown in table 1. Basic execution time is programmed fixed part of service response time (unpredictable server delays additionally includes).

The servers were located in different sites in the Internet, and had different load thresholds (the number of requests processed in parallel) as shown in table 2.

Table 2. Locations of servers and load threshold

Server	DNS address	IP	Country	Threshold
A	planetlab2.rd.tut.fi	193.166.167.5	Finland	6
B	onelab11.pl.sophia.inria.fr	138.96.116.21	France	6
C	ple1.dmcs.p.lodz.pl	212.51.218.235	Poland	8
D	planetlab1.unineuchatel.ch	192.42.43.22	Switzerland	8
E	planetlab4.cs.st-andrews.ac.uk	138.251.214.78	UK	10
F	planet1.unipr.it	160.78.253.31	Italy	10

The effectiveness of estimation was tested as follows:
- during 3 hours there were generated increasing number of clients (from 0 to 100) requesting all services,
- the requests were distributed according to round-robin algorithm,
- the completion times of service execution were measured on each server,
- the measured values were compared against the estimated ones.

In figure 4 it is presented Mean Absolute Percentage Error calculated for estimation of service execution time T_{PROCESS}, on the basis on service request monitoring at TCP level. The values are given for all 36 service instances presented in such order: instance of service UA on server A, B, C,…, UB on A, B, C,… etc.

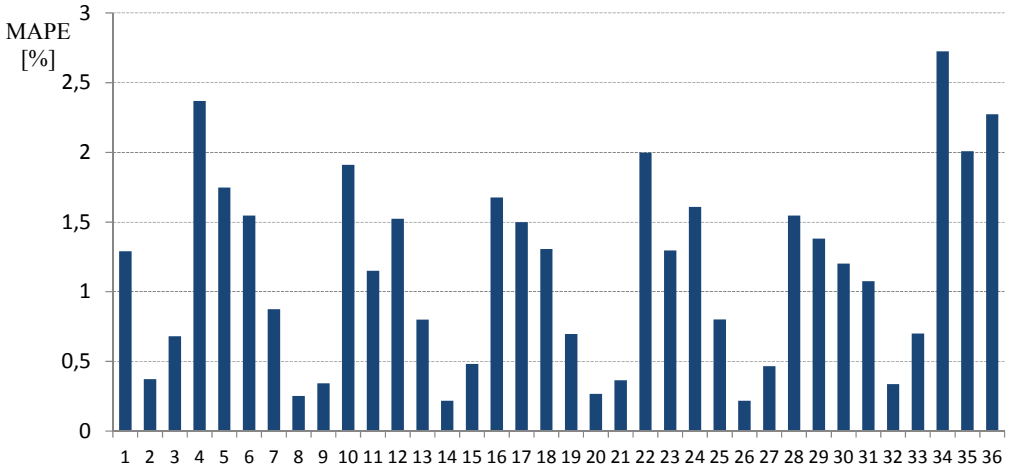


Fig. 4. The MAPE of T_{PROCESS} time estimation for all service instances

For performed experiment the precision of estimation is very high and usually doesn't exceeds 2%. The value of MAPE of all estimation (global metric) is equall 1,13%. It can be noticed that bars are quasi-periodic for every 6 instances what means that the accuracy of estimation correlate with given server.

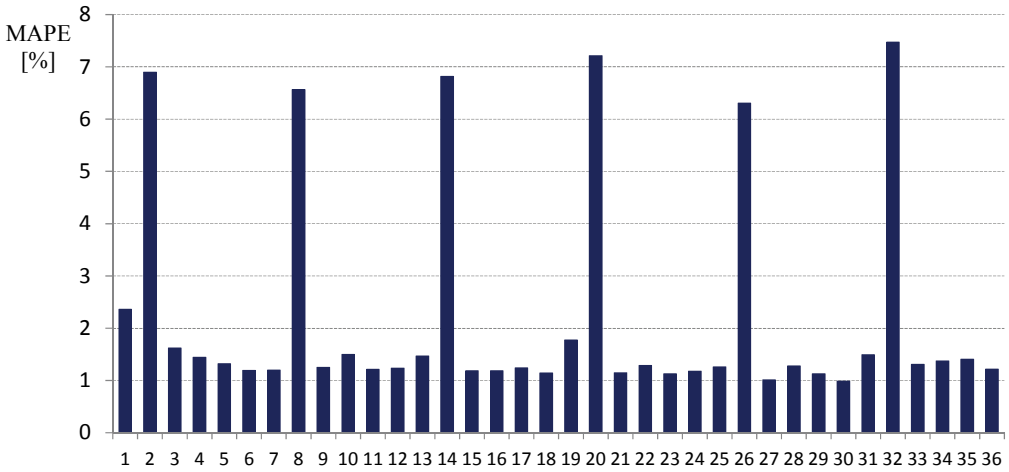


Fig. 5. The MAPE of forecasting $T_{PROCESS}$ with use on fuzzy-neural controller for all service instance

The figure 5 presents the same metric of effectiveness but for forecasting of the time $T_{PROCESS}$ with use of fuzzy-neural controller. It shows two facts. First, the Accuracy of forecasting is clearly worse. The value of global MAPE is equal 2,24%. Socond, it is caused by processing on second server (server B). It was found, that server B was the only which was overloaded. It shows that in heavy conditions of processing the forecasting must be performed carefully.

5. FINAL REMARKS

The performed experiment showed that presented method of estimation of values of network service execution parameters can be successfully used. However, it must be mentioned that all processing components were located fairly close to good backbone – for running services Planet Lab servers were used. But in spite of overload of one server, estimation for it was very precise. It was probably caused by the completion time of execution on this server. Further, the effectiveness of forecasting of values of parameters may significantly decrease when this values are very variable (e.g. when system is overloaded). However, for performed experiment the parameters of fuzzy-neural controller weren't tuned in any way. So the decrease may be lower. In conclusion it must be stated that more extensive measurements will show exact characteristics of presented method.

REFERENCES

[1] 1.O'BRIEN L., MERSON P., BASS L., *Quality Attributes for Service-Oriented Architectures*, Proc. of the Int. Workshop on Systems Development in SOA Environments, IEEE Computer Society, Washington DC, 2007.

[2] BOX D., CURBERA F., HONDO M., KALER C., LANGWORTHY D., NADALIN A., NAGARATNAM N., NOTTINGHAM M., von RIEGEN C., SHEWCHUK J., *Web Services Policy Framework (WS-Policy)*, June 2003; http://public.dhe.ibm.com/software/dw/specs/ws-polfram/ws-policy2003.pdf.

[3] KELLER A., LUDWIG H., *The WSLA Framework: Specifying and Monitoring Service Level Agreements for Web Services*, Journal of Network and Systems Management, Volume 11, Number 1, Springer, New York, 2003, 57-81.

[4] SCHMIETENDORF A., DUMKE R.; REITZ D., *SLA Management – Challenges in the Context of Web-Service-Based Infrastructures*, In Proc. of the IEEE International Conference on Web Services, San Diego, California, 2004

[5] KUNZ M., SCHMIETENDORF A., DUMKE R., WILLE C., *Towards a Service-Oriented Measurement Infrastructure*, In Proc. of the 3rd Software Measurement European Forum (Smef 2006), Rome, Italy, May 2006, 197-207.

[6] ZENG L., BENATALLAH B., NGU A. H. H., DUMAS M., KALAGNANAM J., CHANG H., *QoS-aware middleware for Web services composition*, In IEEE Transactions on Software Engineering, vol. 30, no. 5, 2004, 311-327.

[7] WILLIAMS N., HERMAN R., LOPEZ L. A., EBBERS M., *Implementing CICS Web Services*, IBM Readbook, 2007.

[8] FRAŚ M., *The architecture of complex service requests broker*, Information Systems Architecture and Technology: Networks and Networks' Services, A. Grzech (eds)., Wroclaw University of Technology Publishing House, Wrocław, 2010, 369-379.

[9] FRAŚ M., ZATWARNICKA A, ZATWARNICKI K., *Fuzzy-neural controller in service request distribution broker for SOA-based systems*, Proc. of Int. Conf. Computer Networks 2010, Kwiecień A., Gaj P., Stera P. (eds), Berlin, Heidelberg, Springer, 2010, 121-130.

[10] BORZEMSKI L, ZATWARNICKA A, ZATWARNICKI K., *Global distribution of HTTP requests using the fuzzy-neural decision-making mechanism*, Proc. of 1st Int. Conf. on Comp. Collective Intelligence, In Lecture Notes in AI, Springer, 2009.

BIBLIOTEKA INFORMATYKI SZKÓŁ WYŻSZYCH

Information Systems Architecture and Technology. Web Information Systems: Models, Concepts & Challenges, pod redakcją Leszka BORZEMSKIEGO, Adama GRZECHA, Jerzego ŚWIĄTKA, Zofii WILIMOWSKIEJ, Wrocław 2008

Information Systems Architecture and Technology. Information Systems and Computer Communication Networks, pod redakcją Adama GRZECHA, Leszka BORZEMSKIEGO, Jerzego ŚWIĄTKA, Zofii WILIMOWSKIEJ, Wrocław 2008

Information Systems Architecture and Technology. Models of the Organisations Risk Management, pod redakcją Zofii WILIMOWSKIEJ, Leszka BORZEMSKIEGO, Adama GRZECHA, Jerzego ŚWIĄTKA, Wrocław 2008

Information Systems Architecture and Technology. Designing, Development and Implementation of Information Systems, pod redakcją Leszka BORZEMSKIEGO, Adama GRZECHA, Jerzego ŚWIĄTKA, Zofii WILIMOWSKIEJ, Wrocław 2008

Information Systems Architecture and Technology. Model Based Decisions, pod redakcją Jerzego ŚWIĄTKA, Leszka BORZEMSKIEGO, Adama GRZECHA, Zofii WILIMOWSKIEJ, Wrocław 2008

Information Systems Architecture and Technology. Advances in Web-Age Information Systems, pod redakcją Leszka BORZEMSKIEGO, Adama GRZECHA, Jerzego ŚWIĄTKA, Zofii WILIMOWSKIEJ, Wrocław 2009

Information Systems Architecture and Technology. Service Oriented Distributed Systems: Concepts and Infrastructure, pod redakcją Adama GRZECHA, Leszka BORZEMSKIEGO, Jerzego ŚWIĄTKA, Zofii WILIMOWSKIEJ, Wrocław 2009

Information Systems Architecture and Technology. Systems Analysis in Decision Aided Problems, pod redakcją Jerzego ŚWIĄTKA, Leszka BORZEMSKIEGO, Adama GRZECHA, Zofii WILIMOWSKIEJ, Wrocław 2009

Information Systems Architecture and Technology. IT Technologies in Knowledge Oriented Management Process, pod redakcją Zofii WILIMOWSKIEJ, Leszka BORZEMSKIEGO, Adama GRZECHA, Jerzego ŚWIĄTKA, Wrocław 2009

Information Systems Architecture and Technology. New Developments in Web-Age Information Systems, pod redakcją Leszka BORZEMSKIEGO, Adama GRZECHA, Jerzego ŚWIĄTKA, Zofii WILIMOWSKIEJ, Wrocław 2010

Information Systems Architecture and Technology. Networks and Networks Services', pod redakcją Adama GRZECHA, Leszka BORZEMSKIEGO, Jerzego ŚWIĄTKA, Zofii WILIMOWSKIEJ, Wrocław 2010

Information Systems Architecture and Technology. System Analysis Approach to the Design, Control and Decision Support, pod redakcją Jerzego ŚWIĄTKA, Leszka BORZEMSKIEGO, Adama GRZECHA, Zofii WILIMOWSKIEJ, Wrocław 2010

Information Systems Architecture and Technology. IT TModels in Management Process, pod redakcją Zofii WILIMOWSKIEJ, Leszka BORZEMSKIEGO, Adama GRZECHA, Jerzego ŚWIĄTKA, Wrocław 2010

Information Systems Architecture and Technology. Web Information Systems Engineering, Knowledge Discovery and Hybrid Computing, pod redakcją Leszka BORZEMSKIEGO, Adama GRZECHA, Jerzego ŚWIĄTKA, Zofii WILIMOWSKIEJ, Wrocław 2011

Information Systems Architecture and Technology. Service Oriented Networked Systems, pod redakcją Adama GRZECHA, Leszka BORZEMSKIEGO, Jerzego ŚWIĄTKA, Zofii WILIMOWSKIEJ, Wrocław 2011

Information Systems Architecture and Technology. System Analysis Approach to the Design, Control and Decision Support, pod redakcją Jerzego ŚWIĄTKA, Leszka BORZEMSKIEGO, Adama GRZECHA, Zofii WILIMOWSKIEJ, Wrocław 2011

Information Systems Architecture and Technology. Information as the Intangible Assets and Company Value Source, pod redakcją Zofii WILIMOWSKIEJ, Leszka BORZEMSKIEGO, Adama GRZECHA, Jerzego ŚWIĄTKA, Wrocław 2011

Wydawnictwa Politechniki Wrocławskiej
są do nabycia w księgarni „Tech”
plac Grunwaldzki 13, 50-377 Wrocław
budynek D-1 PWr., tel. 71 320 29 35
Prowadzimy sprzedaż wysyłkową
zamawianie.ksiazek@pwr.wroc.pl

ISBN 978-83-7493-702-3