1st Workshop on Intelligent and Knowledge oriented Technologies

WIKT 2006 Proceedings

Michal Laclavík
Ivana Budinská
Ladislav Hluchý (Eds.)

November 28 - 29, 2006
Bratislava, Slovakia

The workshop was organized by

Institute of Informatics, Slovak Academy of Sciences, Bratislava
Faculty of Informatics and Information Technologies,
Slovak University of Technology in Bratislava
Faculty of Electrical Engineering and Informatics, Technical University of Košice

The workshop was supported by

NAZOU SPVV 1025/2004, VEGA 2/7098/27, RAPORT APVT-51-024604

Program Committee

Ladislav Hluchý	Institute of Informatics, Slovak Academy of Sciences
Pavol Návrat	Institute of Informatics and Software Engineering, FIIT STU
Ján Paralič	Faculty of Electrical Engineering and Informatics, TU
Peter Vojtáš	Institute of Computer Sciences, Faculty of Science, UPJŠ
Mária Bieliková	Institute of Informatics and Software Engineering, FIIT STU
Michal Laclavík	Institute of Informatics, Slovak Academy of Sciences

Organizing Committee

Ladislav Hluchý
Michal Laclavík
Ivana Budinská
Zoltán Balogh
Marián Babík
Oľga Schusterová
Institute of Informatics, Slovak Academy of Sciences
Dúbravská cesta 9, 845 07 Bratislava, Slovakia
E-mail: wikt.ui@sav.sk

Proceeding Editors

Michal Laclavík
Ivana Budinská
Ladislav Hluchý
Institute of Informatics, Slovak Academy of Sciences
Dúbravská cesta 9, 845 07 Bratislava, Slovakia
E-mail: {laclavik.ui, ivana.budinska, hluchy.ui}@savba.sk

ISBN 978-80-969202-5-9

Predhovor

Teší nás, že vám môžeme predstaviť zborník 1. workshopu zameraného na inteligentné a znalostne orientované technológie - WIKT 2006, ktorý sa uskutočnil 28. – 29. novembera 2006 v Bratislave.

Dôležitým faktorom v budovaní znalostnej ekonomiky je schopnosť organizácií zhodnotiť svoj znalostný kapitál. Informačné technológie a znalosti sa stávajú kľúčovým prvkom zvyšovania produktivity. Slovenské firmy a slovenská ekonomika nemôžu obstáť v týchto zmenách bez výskumu a vývoja v tejto oblasti.

Cieľom WIKT bolo a je podporiť posun organizácií k znalostnej ekonomike na základe výskumu a vývoja v oblasti inteligentných a znalostne orientovaných technológií, výmena informácií, diskusia o aktuálnych problémoch, skúsenosti s použitím technológií a softvéru v danej oblasti ako aj jej zviditeľnenie v rámci Slovenska a susedných krajín.

Hlavné témy workshopu boli:

- umelá inteligencia
- modelovanie znalostí, ontológie
- sémantický web
- sémantické spracovanie informačných zdrojov
- spracovanie informačných zdrojov v slovenskom jazyku
- sémanticky a servisne orientované architektúry
- znalostné bázy a organizačné pamäte
- usudzovanie a odvodzovanie

Chceli by sme poďakovať všetkým, ktorí prispeli k úspešnému uskutočneniu workshopu ako aj pri príprave tohto zborníka. Chceme poďakovať programovému a organizačnému výboru a hlavne všetkým autorom, za ich príspevky a prezentácie na workshope. Za hlavný prínos workshopu pokladáme, že sa podarilo naštartovať diskusiu a spoluprácu v tejto oblasti a dúfame, že táto spolupráca prinesie výsledky ktoré budu prezentované a diskutované na ďalších workshopoch WIKT.

Michal Laclavík, Ivana Budinská, Ladislav Hluchý
Marec 2007, Bratislava

Preface

We are pleased to introduce the Proceedings of the 1st Workshop on Intelligent and Knowledge-oriented Technologies - WIKT 2006, held on 28 – 29 November 2006 in Bratislava.

An important aspect in building knowledge economy is the ability of organizations to assess their knowledge capital. Technologies and knowledge are becoming a key factor in productivity growth. Slovak companies and Slovak economy cannot be successful in such global challenge without research and development in this field.

The Workshop on Intelligent and Knowledge-oriented Technologies aims at supporting collaboration and information exchange among researchers and students from Slovakia and the neighboring countries working in this area.

The topics of the workshop include:

- Artificial intelligence
- Knowledgemodeling, ontologies
- Semantic Web
- Semantic processing of information resources
- Processing of information resources in Slovak language
- Semantic and service-oriented architectures
- Knowledge bases and organizational memories
- Reasoning and inference.

Many people have assisted in the success of this workshop. We would like to thank all the members of the Programme and Organizing Committees and to the workshop secretariate for their work and assistance for the workshop. We would also like to express our gratitude to all authors for contributing their research papers as well as for their participation in the workshop that made our cooperation more fruitful and successful.

Michal Laclavík, Ivana Budinská, Ladislav Hluchý
March 2007, Bratislava, Slovakia

Table of contents

Invited section

Slovak morphology analyzer based on Levenshtein edit operations

Radovan Garabík

Ľudovít Štúr Institute of Linguistics
Slovak Academy of Sciences
Bratislava, Slovakia
korpus@juls.savba.sk
http://korpus.juls.savba.sk/

Abstract. Levenshtein edit operation is a basic string operation – insertion, deletion or substitution of a character in a string. Sequence of edit operations can be used to transform basic word form (lemma) into an inflected form, and the same sequence can can be used to transform lemmata belonging to the same inflectional paradigm. Presented system contains inflection paradigms of over 56000 lemmata from Short Dictionary of Slovak Language and from the most frequent word forms in the Slovak National Corpus, together with detailed grammar information about each generated word form.

1 Levenshtein distance and some definitions

Levenshtein distance[1] is a metric defined on the space of strings as a minimum number of Levenshtein edit operations needed to transform one string into the other, where by a Levenshtein edit operation we understand insertion, deletion or a substitution of a character.

A Levenshtein edit operation e can be formally described as $e = (o, s, d)$ – a triple of operation type o, position in the source string s and position in the destination string d, where operation type o is one of *replace*, *insert* or *delete*. For *replace* or *insert*, the replacement/new character is taken from the destination string.

Sequence of edit operations $q = (e_1, e_2, e_3, ...)$, together with the destination string D, when applied to a string $S \in \mathbb{S}$ defines a mapping function $f : \mathbb{S} \mapsto \mathbb{S}$, where $\mathbb{S}$ is a set of all strings.

To each word form $w \in \mathbb{W}$, where $\mathbb{W}$ is a set of all the words we can assign a set of *grammar categories* $G_w = \{g_1, g_2, g_3, ...\}$ represented by short mnemotechnical strings (called *morphological tags*).

Now for each tagged word form together with its morphological tag $(w_i, g_i) \in \mathbb{W} \times \mathbb{G}$ there exists a mapping function f_i consisting of Levenshtein edit operations such that $f_i(l) = w_i$, where l is a deliberately chosen word, called *lemma* and considered to be a basic word form for a given lexeme.

2 Technical implementation

Our system is really just a morphology generator – for each lemma known to it, it is able to generate all the forms, together with their respective tags. By putting all the forms and tags with information about lemma into a database[2], the system is able to work as a morphology analyzer – we just look up the analysed form in the database and find out corresponding morphological tag and lemma.

The system consists of two logically disjunct parts. One part is responsible for creating tables of paradigm templates and lists of mapping of all the lemmata into appropriate paradigm templates. This contains also helper programs used by linguists to create, evaluate and modify these tables and lists.

The second part is meant for end users queries and is nothing more than a simple wrapper around the database query library, to facilitate the lookup, with some simple logic implemented to account for creating superlatives out of comparatives (by adding the prefix *naj-*) and for creating verb negation (by adding the prefix *ne-*, with the exception of the verbs *ísť* and *byť*),

The software is published under GNU General Public License[3] version 2 and can be obtained from the Slovak National Corpus WWW page[1].

3 General principles

The system responsible for creating, testing and editing paradigms is written completely in the Python programming language[4], paradigm editing and testing is done using a simple CLI interface.

All the texts, input and output in our system is unconditionally in UTF-8 encoding[5], and all the internal logic of the system uses Python unicode strings. Word forms in the tables are kept in UTF-8 encoding.

Since it is a suffix morphology we are interested in, we need to count the position for Levenshtein edit operations from the end of the words, so that words of different lengths but sharing the same suffix inflections can be declined by the same paradigm template, in order not to inflate unnecessarily the number of paradigm templates. This is easily realised by reversing the input strings before applying the edit operations, and by reversing the output obtained as the result – all done transparently to the users.

To keep the number of paradigm templates down we let our system work in NFD Unicode normalization[6] internally, normalizing user input into NFD before processing, and normalizing the output to NFC. This takes into account changes in orthographic palatalization of the last consonant *d, t, n* or *l*, represented only by adding or removing the final háček (combining diacritical character in the NFD normalization), but in the more usual NFC normalization it would have to be represented by changing the last character, and therefore requiring separate paradigms for each consonant.

[1] `http://korpus.juls.savba.sk`

4 API

System contains three constant database tables:

- `form2lemma.cdb` – table containing word forms as keys and corresponding lemmata as values
- `form2taglemma.cdb` – table containing word forms as keys and morphological tags and lemmata as values
- `lemma2tagforms.cdb` – table containing lemmata as keys and morphological tags and word forms as values

The constant database tables can be accessed directly, from any programming language supporting tdb database, or converted into a convenient form. However, the databases do not contain neither verb negation, comparatives nor superlatives. Preferred higher level API written in the Python programming language is contained in the module `mlv_skling`. The module has one public class, `Morphology`, that takes during instantiation as an argument path to the directory containing all the necessary morphology tables. Methods available are `form2taglemma`, `lemma2tagforms`, `get_stems` and `get_stem`. `form2taglemma` and `lemma2tagforms` analyze given word form or lemma and return tuple of morphological tag and lemma, or morphological tag and form. `get_stems` applies simple stemming algorithm to the word and return a list of all possible stems. `get_stem` returns just the first stem found.

5 Stemming algorithm

Stemming is the process of finding the stem (base or root form) of inflected words, regardless of the existence of the stem alone – it is sufficient if the word forms of the same lexeme map to the same stem, in order to facilitate full text indexing and search. Usually, for full text query purposes, we are not interested in the grammar analysis, and we want to maximize recall at the expense of precision. Our stemming algorithm uses lemma as the basic form, stripping vowels following the rightmost consonant in the lemma, and in case of verbs, stripping the infinitive suffix *-t'* and then stripping the vowels. This collapses many near homonyms to the same form, directly usable as the word stem.

6 Language coverage

At the time of writing, the database contains all the words from the 3^{rd} edition of the Short dictionary of Slovak language, with several thousand additional most frequent words present in the Slovak national corpus, adding up to 56269 different lemmata (54315 unique lemmata, accounting for homonymy). These lemmata are inflected by 1365 different paradigms, giving 601 253 unique word forms and 1 616 379 different pairs of morphological tags and word forms.

An average tokenized fiction text contains 19 % of punctuation and other nonword elements. On average, the analyzer covers 91 % of the remaining tokens,

where 45 % of tokens are unambiguously assigned their morphology categories and lemmata, 61% of tokens have unambiguously assigned lemmata, but not morphological tags.

Stemming is markedly better, only 6.6 % of tokens cannot be stemmed unambiguously (and this is mostly due to rather frequent ambiguous words *je*, lemma *byt'* or *jest'* and *si*, lemma *byt'* or *si*) – in an information retrieval system, these words would be probably included in the list of stopwords, further improving unambiguity of the stemming.

7 Conclusion and future work

At the time of writing, vocabulary of the presented system is still being improved. The next task will be to add most frequent acronyms, abbreviations and proper names (toponyms and anthroponyms). Numerals will be taken care with the help of additional module, exploiting their regular formation. To improve the percentage of unmarked words in the analyzed texts, a "guesser" module will be implemented, trying to find out the nearest appropriate morphologic tag for words not found in the dictionary, based on suffix similarity with existing words.

The analyzer is able to obtain lemmata and grammar categories for a broad range of most frequent Slovak words, including punctuation and digits, and is successfully used in the Slovak National Corpus database.

References

1. Левенштейн, В. И.: Двоичные коды с исправлением выпадений, вставок и замещений символов, Докл. АН СССР, 163, 4, (1965) 845–848.
2. http://cr.yp.to/cdb.html
3. Free Software Foundation, Inc. (1989, 1991)
4. http://www.python.org/
5. The Unicode Consortium. The Unicode Standard, Version 4.0 Boston, MA, Addison-Wesley Developers Press, ISBN 0-321-18578-1 (2003)
6. The Unicode Consortium. Unicode Technical Report #15: Unicode Normalization Forms. http://www.unicode.org/unicode/reports/tr15/

Speech is more than only its linguistic content

Rusko Milan

Institute of Informatics of the Slovak Academy of Sciences, Bratislava
milan.rusko@savba.sk

Abstract. The paper focuses on the extra-linguistic content of speech and discusses possible ways to handle the information carried by it. Expressive speech synthesis and automatic emotion recognition are examples of future application areas of the studied phenomena. The paper tends to explain what does „expressive speech research“ mean, and what phenomena does it study. A short survey is given on some of the research which has been done in this field, partly led by an attempt to develop a comprehensive expressive speech synthesis. Acoustical characteristics that are believed to be in close correlation with some of the factors of expressiveness are pointed out and some methods of their measurement and visualization are mentioned.

Introduction

The users of the speech synthesizers often complain that the synthetic speech suffers from being too monotonous and unnatural. It does not have appropriate dynamics, phrasing, style, and interpretation and appropriate variations in dynamics and tempo. It does not tell listeners anything about the speaker's thoughts, feelings, and/or personal experience.
This should be changed in expressive speech synthesis.

Expressive speech

“Expressive speech” designates the whole vocal display of a speaker, that contains not only that part of information that can be encoded in general written text message, but contains also various information about the speaker himself – his age, cultural background, education, sex, attempt, relation to the listener, as well as his individuality etc. The expression “individuality” is used here to denote personality, mood (attitude) and emotions of a speaker.

Personality

Personality is considered to be a set of constant features of an individual.

Personality and temperament

In his study on Hans Eysenck's theory of personality C. G. Boeree states, that temperament is that aspect of personality that is genetically based, inborn. The ancient Greeks came up with two dimensions of temperament, leading to four types of temperament.
The **sanguine** type is cheerful and optimistic, healthfully looking, pleasant to be with, comfortable with his/her work.
The **choleric** type is characterized by a quick, hot temper, often an aggressive nature. Physical features of the choleric person include a yellowish complexion and tense muscles.
The **phlegmatic** type is characterized by slowness, laziness, and dullness.
The people with **melancholy** temperament tend to be sad, even depressed, and take a pessimistic view of world.[1]

Generalized model of personality

In a generalized model can the personality p have n dimensions, and so it can be represented by the following vector [2]:

$$p^T = [\alpha_1 ... \alpha_n], \forall i \in [1, n] : \alpha_i \in [0,1] \quad (1)$$

The OCEAN model - The Big Five model of personality

In the last couple of decades, an increasing number of theorists and researchers have come to the conclusion that five dimensions are enough to express the personality. The Big Five model also known as OCEAN model takes into account the following five dimensions of personality: Openness, Consciousness, Extraversion, Agreeableness, and Neuroticism. [3, 4]

Emotion

Mood and Emotion

As mentioned by Ksirsagar&Magnenat-Thalmann a mood can be defined as a rather static state of being, that is less static than personality and less fluent than emotions. Mood can be defined as one-dimensional (e.g. good or bad mood) or perhaps multi-dimensional (feeling in love, being paranoid etc.) [5]

Generalized model of emotion and mood

According to [2] an emotional state has a similar structure as personality, but it changes over time. The emotional state et is defined as an m-dimensional vector, where all m emotion intensities are represented by a value in the interval [0, 1]. A value of 0 corresponds to an absence of the emotion; a value of 1 corresponds to a maximum intensity of the emotion. This vector is given as follows:

$$e_t^T = [\beta_1 ... \beta_m], \forall i \in [1,m] : \beta_i \in [0,1] \quad \text{if } t>0$$
and
$$e_t^T = 0 \quad \text{if } t=0 \qquad (2)$$

Emotions are features dynamically changing in time. The actual emotional state is dependent on the preliminary evolvement of emotins. Therefore the scientists tend to model the emotins respecting their previous trends (history).
Therefore an emotional state history ωt is defined, that contains all emotional states until et, thus:

$$\omega_t = \langle e_0, e_1, ..., e_t \rangle \qquad (3)$$

Egges continues with defining the individual It as a triple (p, m_t, e_t), where m_t represents the mood of the individual at a time t.
Mood dimension is defined as a value in the interval [-1,1]. Supposing there are k mood dimensions, the mood can be described as follows:

$$m_t^T = [\gamma_1 ... \gamma_K], \forall i \in [-1,1] \quad \text{if } t>0$$
and
$$m_t^T = 0 \quad \text{if } t=0 \qquad (4)$$

Moreover the mood and emotional values, as they are changing in time, have to be updated regularly. [2]

Basic emotions

There are many theories of emotions and many different classifications exist:

Table 1. This table, taken from [6] gives a short overview of basic emotion sets used by different authors

Author	Basic Emotions
Arnold	Anger, aversion, courage, dejection, desire, despair, fear, hate, hope, love, sadness
Ekman, Friesen, and Ellsworth	Anger, disgust, fear, joy, sadness, surprise
Frijda	Desire, happiness, interest, surprise, wonder, sorrow
Gray	Rage and terror, anxiety, joy
Izard	Anger, contempt, disgust, distress, fear, guilt, interest, joy, shame, surprise
James	Fear, grief, love, rage
McDougall	Anger, disgust, elation, fear, subjection, tender-emotion, wonder
Mowrer	Pain, pleasure
Oatley and Johnson-Laird	Anger, disgust, anxiety, happiness, sadness
Panksepp	Expectancy, fear, rage, panic
Plutchik	Acceptance, anger, anticipation, disgust, joy, fear, sadness, surprise
Tomkins	Anger, interest, contempt, disgust, distress, fear, joy, shame, surprise
Watson	Fear, love, rage
Weiner & Graham	Happiness, sadness

Placement on emotion dimensions

Semantic differential scales are often used for measuring emotion dimensions. We present here a set of dimensions as proposed by Mehrabian & Russell (1974, Appendix B, p. 216) [7]. It is evident that the authors have included moods and personality dimensions in this system too:

- **Pleasure:** Happy - Unhappy, Pleased - Annoyed, Satisfied - Unsatisfied, Contented -Melancholic, Hopeful - Despairing, Relaxed - Bored
- **Arousal:** Stimulated - Relaxed, Excited - Calm, Frenzied - Sluggish, Jittery - Dull, Wide-awake - Sleepy, Aroused - Unaroused
- **Dominance:** Controlling - Controlled, Influential - Influenced, In control - Cared-for, Important - Awed, Dominant - Submissive, Autonomous - Guided

Acoustic correlates of emotions

The problem of speech parameter identification responsible for the expression of personality, moods and emotions is very difficult mainly due to the fact that the acoustic means used to encode the expressive information in the speech are shared by all the components of expressivity. Moreover the code (expressive speech) is not decoded (understood) by all the speakers in the same way. The understanding is very subjective. This is evident in the literature, as results taken from numerous studies rarely agree with each other. Nevertheless, a general set of the speech parameters responsible for the expression of emotion can be constructed. There are three main categories of speech correlates of emotion:

- Pitch contour
- Timing
- Voice quality

It is believed that value combinations of these speech parameters are used to express vocal emotion.[8]

Pitch contour

Pitch contour is a representation of the intonation of an utterance, which describes the nature of accents and the overall pitch range of the utterance. Pitch is expressed as fundamental frequency (F0). One of the most frequently used methods for F0 measurement is the method using autocorrelation function of the LP residual.
Pitch parameters include average pitch, pitch range, contour slope, and final lowering.

Table 2. A summary of human vocal emotion effects of four of the so-called basic emotions: anger, happiness, sadness and fear. The parameter descriptions are relative to neutral speech.

	Anger	**Happiness**	**Sadness**	**Fear**
Speech rate	Faster	Slightly faster	Slightly slower	Much faster
Pitch average	Very much higher	Much higher	Slightly lower	Very much higher
Pitch range	Much wider	Much wider	Slightly narrower	Much wider
Intensity	Higher	Higher	Lower	Higher
Pitch changes	Abrupt, down-ward, directed contours	Smooth, upward inflections	Downward inflections	Down-ward terminal inflections
Voice quality	Breathy, chesty tone	Breathy, blaring	Resonant	Irregular voicing
Articu-lation	Clipped	Slightly slurred	Slurred	Precise

Models of intonation can be divided into two main categories – phonetic and phonological models. The phonetic model (Fujisaki model, Tilt model and many others) models the intonation curve. The phonological model (e.g. ToBI) is used to model the speaker's concept of distribution of accents in the intonation phrase.

Timing

Timing describes the speed that an utterance is spoken, as well as rhythm and the duration of emphasized syllables. The results of measurement of syllable and phoneme lengths are often given in a form of z-scores i.e. the instantaneous value is normalized be the mean value of the same elements in the whole database (Z-score is equal to a value of X minus the mean of X, divided by the standard deviation). Timing parameters include speech rate, hesitation pauses, and exaggeration.

Voice quality

Voice quality denotes the overall ‘character’ of the voice, which includes effects such as whispering, hoarseness, breathiness, and intensity. The voice quality is influenced mainly by the function of glottis and the function of the vocal tract. A detailed classification scheme of voice qualities was published by Laver [9].

Conclusion

This work tries to explain the term expressive speech. It discusses the generalized model of personality, mood and emotions. It gives a short overview of different classifications of these three components of individuality. Last but not least it shows some methods of acoustical analysis used in an effort to find acoustical correlates of personality, mood and emotions in speech and lists some promising results in expressive speech synthesis.

Acknowledgement

This work was funded by the Ministry of Education of the Slovak Republic, grant 2003 SP 20 028 01 03 and by VEGA, grant No. 2/2087/22.

References

1. Boeree G., Hans Eysenck and other personality theorists http://www.ship.edu/~cgboeree/eysenck.html
2. Egges, A., Kshirsagar, S., Magnenat-Thalmann, N.: A Model for Personality and Emotion Simulation, www.miralab.unige.ch/papers/162.pdf
3. Digman, J. M., Personality structure: Emergence of the five factor model, Annual Revue of Psychology, 41, 1990, pp. 417-440.
4. McRae, R.R.; John, O.P.; 1992 “An introduction to the five-factor model and its applications”, Journal of Personality 60, pp.175-215.
5. Kshirsagar, S., Magnenat-Thalmann, N.: A multilayer personality model. In. Proceedings of 2nd International Symposium on Smart Graphics, ACM Press, 2002, pp. 107-115
6. Ortony, A., Turner, T. J. What's basic about basic emotions? Psychological Review, 97, (1990), pp. 315-331.
7. Mehrabian, A., Russell, J.A. (1974). An approach to environmental psychology. Cambridge, M.I.T. Press.
8. Schröder M., Speech and Emotion Research, An Overview of Research Frameworks and a Dimensional Approach to Emotional Speech Synthesis.

Agents with incomplete and evolving knowledge

Ján Šefránek

Institute of Informatics, Faculty of Mathematics and Physics, Comenius University,
Mlynska dolina, 842 15 Bratislava
sefranek@fmph.uniba.sk

Abstract. Knowledge representation and reasoning research. Reasoning with incomplete and evolving knowledge. Non-monotony of reasoning, updates. A programming language schema for development of intelligent agents (a proposal by Peter Novák). The main ideas: encapsulated BDI modules, accessible only via queries/updates, interaction rules (an answer to a query activates an update), capabilities module. The role of updates. Logic programming languages and representation of beliefs, desires, intentions, capabilities. (Multidimensional) dynamic logic programming, EVOLP, logic-based action languages. Logic program updates.

Semantic and service oriented architectures

HYDRA Project - Use of Semantic Technologies for Networked Embedded System Middleware

Tomáš Sabol, Peter Kostelník, and Martin Sarnovský

Technical University of Košice,
Letná 9, 04001 Košice, Slovakia
{Tomas.Sabol,Peter.Kostelnik,Martin.Sarnovsky}@tuke.sk

Abstract. The paper describes the IST-2005-034891 project HYDRA funded within the IST, 6^{th} Framework Programme of the EU. HYDRA aims to develop middleware based on service-oriented architecture, deployable on both new and existing networks of distributed wireless and wired devices. The embedded service-oriented architecture will provide interoperable access to data, information and knowledge across heterogeneous platforms. The vision of the project, overall design objectives and scientific objectives are outlined.

Key words: semantic technology, service-oriented architecture, ontology modelling, networked embedded systems, ambient intelligence, middleware

1 Introduction

The IST-2005-034891 Project HYDRA (in full "Networked Embedded System Middleware for Heterogeneous Physical Devices in a Distributed Architecture") is an Integrated Project funded by the EC within Information Society Technologies (IST) Programme within FP6. Project started on July 1^{st}, 2006 and its expected duration is four years. The project consortium consists of 13 partners from UK, Sweden, Denmark, Germany, Spain, Italy, and Slovakia (9 companies, 3 universities and one research institute), the Project Coordinator is C International Ltd. from UK, and one of the project partners is also the Technical University of Košice. The project was submitted and approved within the 5^{th} IST Call under the Strategic Objective 2.5.3 Embedded Systems. Total estimated effort behind this project is 1395 person-months.

The HYDRA project is addressing the problem, which is frequently faced by producers of devices and components - the need for (which is actually becoming a trend) networking the products available on the market in order to provide higher value-added solutions for their customers. This requirement is implied by citizen centred demands requiring intelligent solutions, where the complexity is hidden behind user-friendly interfaces to promote inclusion.

The vision of the HYDRA project is rather ambitious: *To create the most widely deployed middleware for intelligent networked embedded systems that will allow producers to develop cost-effective and innovative embedded applications for new and already existing devices.*

To put it in practical terms: In the ambient world of the near future, interconnected intelligent devices will surround us, at home, work, or while travelling. These devices and their local networks will also be connected to the outside world through broadband and/or wireless networks. Numerous services to support us in our personal life will be provided through these ambient devices and over the connection to the outside world. To adapt to our personal lifestyle, and to offer the right service at the right time in the right place, such services will rely on the use of private data - which means putting emphasis also on security and privacy. It is expected that the HYDRA will contribute to this scenario.

2 Challenges addressed and Project objectives

In comparison with the state-of-the-art on the technology market the project is facing several challenges:

1. The first challenge is to allow for the seamless access to the features of many devices, regardless of its manufacturer, technology, interfaces, location, communication mechanism, etc. and to create seamless, intelligent and secure interoperability between such devices.
2. Second challenge is related to fast changing environments of mobile users - ambient services and applications should thus adapt to changing local and global sets of accessible sensors and actuators, and must put together partial states of internal and location-determined information. When an end-user moves around interacting with any device in either private or public space, it is the right information that must follow their migration from different locations in changing surroundings.
3. Third challenge is to develop a framework for secure, trustworthy communication among networked embedded systems and supporting self-adaptive interplay of different components, not only sensors but also controlling components and actuators.

Overall project objectives can be summarised in the following points:

1. Development of a middleware based on a Service-oriented Architecture, to which the underlying communication layer is transparent, and consists of:
   (a) Support for distributed as well as centralised ambient intelligent architectures;
   (b) Support for reflective (i.e. self-) properties of components of the middleware;
   (c) Support for security and trust enabling components
2. Design of a generic semantic model-based architecture supporting model-driven development of applications.

3. Development of a toolkit for developers to develop applications on the middleware.
4. Design of a business modelling framework for analysing the business sustainability of the developed applications.

From scientific point of view the project will carry out foundational and component research as well as application and system integration within the following research areas:

- Embedded and mobile service-oriented architectures for ubiquitous networked devices;
- Semantic Model-Driven Architecture for Ambient Intelligence implementation;
- Ontology-based knowledge modelling;
- Hybrid architectures for Grid enabled networked embedded systems;
- Wireless devices and networks with self-* properties (self-configuring, self-healing, etc.);
- Ambient intelligence autonomic computing;
- Distributed security and privacy.

The implemented HYDRA middleware and toolkit will be validated in real end-user scenarios in three user domains: a) Facility management (intelligent homes), b) Healthcare, c) Agriculture (to be specified).

3 Conclusions

The paper gives an overview, in terms of challenges addressed, project objectives and technologies used, of the R&D EU project HYDRA. The project is aimed at development of middleware for networked embedded systems with the use of advanced technologies, enabling intelligent properties of the whole system. Technical University of Košice, as one of the project partners, will be within the project responsible for: ontology modelling, ontology evolution, annotation of dynamic events, use of semantic technologies for security and privacy, knowledge discovery, classification and inference etc.; For more information on HYDRA look at http://www.c-lab.de/en/research-projects/hydra/index.html.

Universal semantic knowledge middleware

Marek Paralič, Jozef Wagner

Centre for Information Technologies
Technical University of Košice, Boženy Němcovej 3, 042 00 Košice
{Marek.Paralic, Jozef Wagner}@tuke.sk

Abstract. In this paper we briefly describe a middleware that offers high level abstraction for work with the knowledge artefacts. Consumers for the middleware services are KP-Lab[1] tools, which are designed within the Knowledge Practices Laboratory project. KP-Lab system is a modular, flexible and extensible system consisting of a cluster of inter-operable applications working in a distributed environment and loosely coupled via services of the middleware.

Introduction

The work shortly described in this paper is related to the Knowledge Practices Laboratory (KP-Lab, FP6-2004-27490) project focused on developing a learning system aimed at facilitating innovative practices of sharing, creating and working with knowledge in education and workplaces. The focus of the KP-Lab system is the process of creating, modifying and utilizing of knowledge artefacts (KAs). These artefacts are employed to alter, extend or preserve group knowing, sense-making or decision-making [1]. Sharing artefacts can be viewed as a collective group problem-solving activity for the purpose of aiding, enhancing, or improving individual and group cognition [2]. Artefacts are created, used or evolved within a cognitive space and a socio-cultural environment and they are employed by the learners to configure and facilitate group decision-making, thinking and communication [3].

KP-Lab is a modular, flexible and extensible system consisting of a cluster of inter-operable applications. The user environment is a virtual shared space and a set of tools that enable collaborative knowledge practices around shared knowledge artefacts [4].

[1] KP-Lab is an integrated EU funded IST project Nr. 27490 running within FP6 since February 2006 (www.kp-lab.org)

KP-Lab System Architecture

If we simplify the KP-Lab Architecture, we obtain three main parts, which interact and fulfill the main goal of working with KAs – KP-Lab tools, knowledge repositories and content repositories (see Figure 1).

Tools in KP-Lab are grouped in the Shared space that offers to the end users the functionality of working with KAs – e.g. Knowledge artefact tool for creating and browsing KAs, Knowledge Processes Tool for creating and managing of knowledge creation processes or Knowledge Annotation tool.

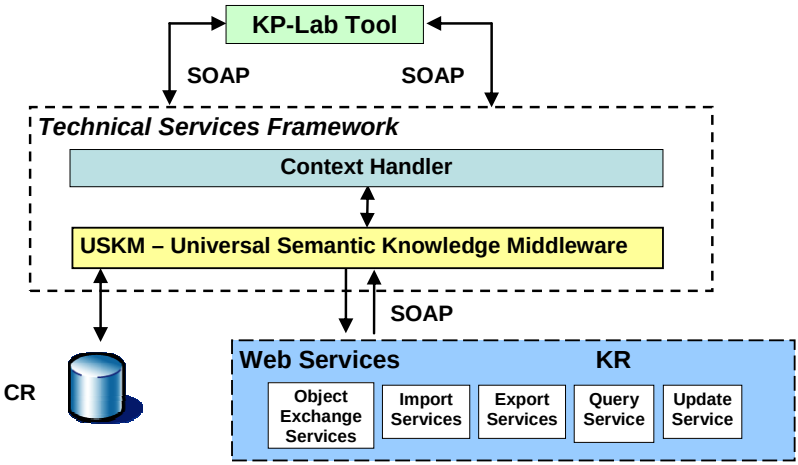


Fig. 1. KP-Lab System Architecture

The objective of the knowledge repositories is to provide generic management services for capturing and archiving; discovering and accessing; combining, modifying and tracking [5] of formal forms of explicit knowledge (i.e. declarative, procedural and causal). The descriptions of knowledge artefacts as well as their involved conceptualizations is represented and handled in knowledge repository as RDF/S schemas and resource descriptions. The knowledge repository in KP-Lab will be built upon the FORTH-ICS RDFSuite open source platform [6] and will provide scalable persistence services for large volumes of knowledge artefacts' descriptions and ontologies.

KR relies on ontologies in order to support interactions among learners and knowledge artefacts involved in learning processes. Ontology-based interactions include but are not limited to the ones that deal with organizing or annotating shared artefacts created by an individual or a group as well as sorting, classifying and retrieving background knowledge artefacts relevant to the problem at hand. Several ontologies are used to capture declarative (about), procedural (how), causal (why) or temporal (when) knowledge regarding a problem addressed by a group. Moreover, ontologies themselves might be possibly formed collaboratively based on the

individual conceptualizations, if a consensus on the concepts and relations that are relevant to the task at hand can be reached.

Digital representation of KA consists of a set of metadata (stored in knowledge repository) and from optional content, which is stored in internal or external content repository. Content repositories (CR) cover the wide range of available persistent storages from simple file systems up to enterprise content management systems (e.g. The Alfresco Network[2]) and learning management systems (e.g. Moodle[3]).

Universal Semantic Knowledge Middleware

To bridge the gap between the KAs aware tools and combination of knowledge and content repositories we proposed a Universal Semantic Knowledge Middleware (USKM), that offers a high level, distributed service based on knowledge artifacts. USKM offers KAs distribution transparency by involving a naming service for knowledge artifacts, gateways to content repositories services, gateways to knowledge repositories services and the context handler services. Basic architecture of the USKM in the context of the KP-Lab Technical Services Framework is depicted in Figure 2.

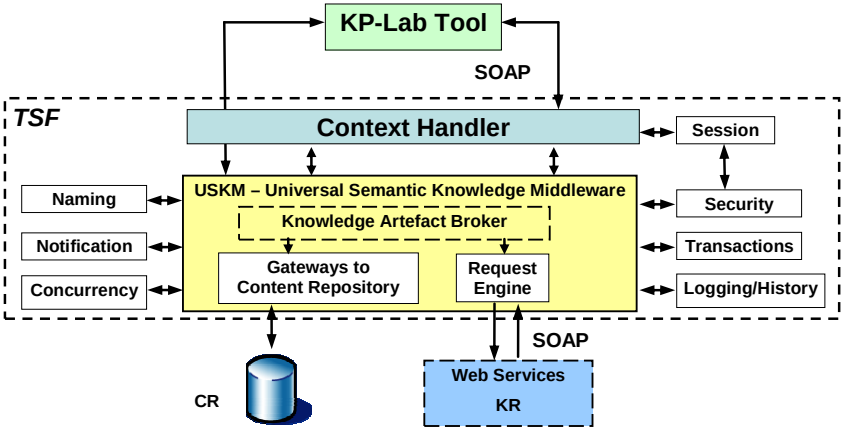


Fig. 2. Universal Semantic Knowledge Middleware architecture

The basic services of USKM include life cycle service, query service, naming service and security service. The life cycle service provides means to create, modify and destroy KAs. The query service provides the means to find a specific KA or a collection of KAs based on defined criteria as a result of browsing the KAs structure. The naming service by which KAs can be given human readable name maps them to the KA's unique identifier. The security service provides facilities for

[2] http://www.alfresco.com/
[3] http://moodle.org/

authentication, authorization, auditing, secure communication, nonrepudiation and administration.To further services of USKM belong specific notification service in case of modification of the content part of the KA in external content repository, transaction service that allows a user to define a series of KA operations in a single transaction and concurrency control service that offer advanced locking mechanisms by which users can access shared KAs.

Acknowledgments

The work presented in this paper was supported within the KP-Lab project by the European Commission DG INFSO under the IST program, contract No. 27490 and by the Slovak Grant Agency of Ministry of Education and Academy of Science of the Slovak Republic within the project No. 1/3135/06 “Methods and tools for design of the integrated distributed applications based on ambients – higher-level agents”.

The KP-Lab Integrated Project is sponsored under the 6th EU Framework Programme for Research and Development. The authors are solely responsible for the content of this article. It does not represent the opinion of the KP-Lab consortium or the European Community, and the European Community is not responsible for any use that might be made of data appearing therein.

References

1. Stahl, G.: Building collaborative knowing: Elements of a social theory of learning. In What We Know about CSCL in Higher Education, Kluwer, Amsterdam, NL, J.-W. Strijbos, P. Kirschner, & R. Martens (Eds.), 2004
2. Hutchins, E.: Cognition in the Wild. Cambridge MIT Press, 1995, pp.381
3. Spillers F., Loewus-Deitch D.: Temporal attributes of shared artefacts in collaborative task environments. In Proceeding of HCI2003 Workshop on the Temporal Aspects of Tasks. Bath, United Kingdom, 2003
4. Babič, F. Paralič, J. Smatana, P. Smrž, P.: Knowledge-practices laboratory. Informacni a komunikacni technologie ve vzdelavani 2006, Roznov pod Radhostem, p.157-161, ISBN 80-7368-199-4
5. Knowledge Management Case Study: Knowledge Management at Ernst&Young, 1997, Available at http://www.bus.utexas.edu/kman/e_y.htm
6. The ICS-FORTH RDFSuite: High-level Scalable Tools for the Semantic Web, http://139.91.183.30:9090/RDF/
7. G. Karvounarakis, A. Magkanaraki, S. Alexaki, V. Christophides, D. Plexousakis, M. Scholl, K. Tolle: Querying the Semantic Web with RQL. Computer Networks and ISDN Systems Journal, Vol. 42(5), August 2003, Elsevier Science, pp. 617-640

Projekt Access-eGov a elektronizácia verejnej správy

Marek Skokan, Peter Džupka

Technická univerzita v Košiciach, Letná 9,
04200 Košice, Slovensko
Marek.Skokan@tuke.sk, Peter.Dupka@tuke.sk

Abstrakt.. Cieľom tohto článku je prezentovať víziu projektu Access-eGov vo vzťahu k e-Governmentu. Projekt využíva sémantické technológie preto je vykreslený ich význam. Je načrtnutá problematika e-Governmentu a miesto projektu v tejto oblasti spolu s pilotnými aplikáciami, ktoré majú význam projektu v e-Governmente potvrdiť. Je taktiež stručne popísaná vízia služieb Access-eGov systému.

Kľúčové slová: e-Government, elektronizácia verejnej správy, sémantické spracovanie informačných zdrojov

1 Úvod

Medzinárodný projekt Access-eGov (Access to e-Government Services Employing Semantic Technologies - prístup k službám elektronickej verejnej správy pomocou využitia sémantických technológií) je financovaný Európskou komisiou v rámci programu IST - Technológie pre Informačnú spoločnosť (Information Society Technologies). Konzorcium projektu, je zložené z jedenástich partnerov z piatich krajín (Slovensko, Poľsko, Nemecko, Grécko a Egypt). Koordinátorom projektu je Technická univerzita v Košiciach, pričom na projektových aktivitách sa podieľajú najmä ekonomická a elektrotechnická fakulta. Projekt začal v januári 2006 a jednotlivé aktivity projektu sú plánované tri roky.

Cieľom projektu je vytvoriť platformu podporujúcu budovanie elektronických služieb verejnej správy, teda platformu, ktorá bude motivačným činiteľom pre transformáciu klasických služieb na on-line služby. Súčasná situácia v oblasti „elektronizácie“ tradičných služieb poskytovaných verejných sektorom je v podmienkach Slovenska zložitá. Práve preto prichádza projekt Acces-eGov s inovatívnym riešením, ktoré je založené nie len na podpore transformácie tradičných služieb na elektronické, ale zároveň na riešeniach podporujúcich kombináciu tradičných a elektornických služieb. Tento nový inovatívny prístup by mal prispieť k rýchlejšej a ľahšej elektronizácii verejnej správy.

Elektornizácia verejnej správy sa všeobecne považuje za jednu z možností podpory ekonomického rozvoja. Pojem ekonomický rozvoj zahŕňa v sebe nie len hospodárske, ale aj sociálne, politické a iné aspekty spoločnosti. Cieľom každej spoločnosti je zabezpečiť pre svojich obyvateľov rovnomerný a dlhodobo udržateľný ekonomický rozvoj. Metódy pre napĺňanie tohto cieľa sú rôzne. Jednou z metód je aj transformácia

tradičných služieb na služby elektronické ktorá je bežne označovaná ako e-Governemnt.

2 e-Government

V súčasnosti sa používa viacero definícii e-Governmentu. Preto existujú aj rozdiely v rozsahu e-Governmentu, ktorý tiež závisí na štruktúre riadenia štátu, na alokácií finančných prostriedkov, dostupnosti technológií, na skúseností a centrálnej vízii e-Governmentu. Posledne menované je prvým krokom v procese budovania eGovernmentu. Je potrebné si uvedomiť, že slovo e-government sa ešte donedávna prakticky nenachádzalo v slovníkoch, zatiaľ čo dnes je v odborných kruhoch „populárne“ a používa sa na označenie mnohých skutočností spojených s budovaním informačnej spoločnosti. Z toho pramení aj množstvo výkladov tohto slova. Uveďme aspoň niektoré z uznávaných výkladov a priblížme si súvisiace kľúčové pojmy.
Európska komisia (EK) definuje e-Government ako: „*Zavádzanie informačno-komunikačných technológií (IKT) do verejnej správy spoločne s organizačnými zmenami, novými postupmi a zručnosťami v snahe zvýšenia efektívnosti pri poskytovaní služieb, zvýšenia transparentnosti a posilnenia verejnej politiky*.“
Podľa United Nations (www.unpan.org) je e-Governemnt definovaný ako využitie internetu a webu pre dodávanie vládnych informácií a administratívnych služieb občanom.
Podľa World Bank (www.worldbank.org) sa e-Government vzťahuje k využitiu takých informačných technológií vo vládnych organizáciách, ktoré umožňujú zmeniť vzťah s občanmi a podnikateľmi. Tieto technológie môžu poskytovať napr. zlepšenie interakcie medzi administratívou a občanmi, zlepšiť prístupnosť administratívnych služieb pre koncových užívateľov atď. Výsledný prínos môže byť v menšej korupcii, zvýšenej úrovne transparentnosti, zlepšenej „pohodlnosti“, raste výnosov, a/alebo redukcii nákladov.
Pri úvahách o zefektívnení vzájomných vzťahov v tejto oblasti je možné naraziť na skratky G2C, G2B, G2G a niekedy aj G2E označujúce v tomto poradí vzťahy„vláda ku občanom – government to citizens“„vláda k podnikateľom – government to business enterprises“, „vláda k vláde – government to government“ „vláda k zamestnancom – government to employees“.

2.1 Miesto projektu v e-Governmente

Projekt Access-eGov prichádza s víziou riešenia, ktoré by malo zmeniť situáciu v sektore G2C a G2B. Pôjde o technologické riešenie orientované smerom na užívateľa. Užívateľmi budú tak občania ako aj podnikatelia, avšak anotovanie služieb bude úloha pre užívateľov systému zo strany inštitúcii verejnej správy. V rámci uvedených definícii je projekt snahou o zavádzanie IKT do verejnej správy v snahe zvýšenia efektívnosti pri poskytovaní služieb (definícia EK), využíva internet a web pre administratívne služby (definícia United Nations) a využíva také IKT, ktoré umožnia zmeniť vzťah inštitúcií verejnej správy s občanmi a podnikateľmi (definícia World Bank).

Je potrebné zdôrazniť, že v súčasnosti nie je úroveň elektronizácie verejnej správy v takom stave, aby všetky jej služby boli poskytované elektronicky a existuje tiež problém integrácie už existujúcich elektronických služieb. Tento prechodný stav vedie k podmienke, ktorou je neodmysliteľná podpora tradičných služieb verejnej správy na ceste ku elektoronizácii. Avšak, táto podpora bude pravdepodobne stále dôležitá aj v čase keď 100% služieb verejnej správy bude v elektronickej forme. V rámci projektu sa ráta aj s touto podporou, pričom sémantický popis tradičných služieb bude zaregistrovaný v Access-eGov systéme.
Na druhej strane množstvo existujúcich elektronických služieb vytvorených jednotlivými inštitúciami verejnej správy nie je zanedbateľné a ich ďalšie využitie bude potrebné (stav EU). Ak však vezmeme do úvahy ich prepojenosť, situácia sa podobá na izolované ostrovy v mori. Pokračujúc v tejto predstave, snahou projektu Access-eGov je tieto ostrovy spojiť a umožniť ich ďalší „rast“. V terminológií služieb verejnej správy, technologické riešenie má umožniť využitie už existujúcich elektronických služieb a bude zabezpečovať integráciu služieb a ich vzájomnú sémantickú interoperabilitu, bude podporovať tradičné služby, pričom napredovanie smerom k elektronizácii tradičných služieb bude závistieť na inštitúciách verejnej správy – distribúcia kompetencií, decentralizácia.
Proces transformácie klasických služieb na on-line služby sa všeobecne považuje za výzvu dneška v e-Governmente (často aj za samotnú definíciu e-Governmentu). Technologické riešenie projektu má vytvoriť motivujúce prostredie k vytváraniu elektronických služieb verejnej správy tým, že umožní jednoduché zavádzanie nových elektronických služieb pre inštitúcie verejnej správy, ich ľahkú lokalizáciu a bude podporovať spomínanú sémantickú interoperability medzi novými a existujúcimi elektronickými službami. Na tento cieľ budú využité sémantické technológie - sémantická interoperabilita, peer-to-peer (P2P) siet, architektúra orientovaná na (web) služby (SOA) - distribúcia kompetencií. P2P sieť umožní navyše chod celej infraštruktúry aj v prípade, že nejaká jej časť práve nefunguje, keďže rovnaké dáta (napr. sémantický popis služieb) môžu byť uložené redundantne. Prístup na báze P2P siete tiež umožní jednoduché a lacné budovanie infraštruktúry. Stať sa jej súčasťou bude znamenať nainštalovať Access-eGov software na počítač spojený s internetom a zaregistrovať tento počítač ako nový uzol v sieti.

2.2 Pilotné aplikácie

Tri rozličné piloty Access-eGov systému budú implementované a vyhodnotené. Špecifikácia pilotov zo strany užívateľských partnerov bola vykonaná.
Slovenský pilot bude implementovaný Košickým samosprávnym krajom a mestským úradom v Michalovciach. Tento pilot je zameraný na územné plánovanie a stavebné povolenie. Cieľom je urobiť tento komplikovaný proces transparentnejší, efektívny a jednoduchší na pochopenie, teda šetriaci čas (a tým pádom aj peniaze) pre občanov a podnikateľov.
Poľský pilot bude implementovaný v regióne Silesia v kooperácii s Cities on Internet Association (asociácia „mestá na internete“) a City Hall of Gliwice (radnica v Glivciach). Tento pilot je zameraný na proces registrácie firmy.

Nemecký pilot bude rozšírením existujúceho riešenia pre podporu elektronických služieb (volané “Zustaendigkeitsfinder” – “vyhľadávač zdrojov”), v štátnej administratíve Šlezvicka-Holštajnska (Schleswig-Holstein), zapracovaním sémantickej vrstvy (bezpečná sémantická interoperabilita medzi národnými a lokálnymi centrami verejnej správy). Výstupom bude zvýšenie kvality služieb pre občanov a podnikateľov pri vyhľadávaní služieb poskytovaných národnými ako aj lokálnymi centrami verejnej správy.
Navyše Nemecká univerzita v Káhire, vďaka jej sídlu v Egypte, predstavuje výzvu v podobe Egyptského testovacieho prípadu: napríklad užívateľ s egyptským občianstvom hľadajúci elektronické služby verejnej správy alebo takýto užívateľ, ktorý chce získať pracovné povolenie v krajine EU. Toto bude zahrňovať všetky úlohy vnútro-európskeho scenára plus ďalšie výzvy v podobe jazykových a kultúrnych rozdielov.

3 Vízia služieb systému Access-eGov z pohľadu jeho cieľových skupín

Tradične sa interakcia občana a vládnej organizácie nevyhnutná pri riešení príslušnej životnej situácie (alebo pri obchodnom prípade podnikateľa) uskutočňuje fyzicky na náležitom centre verejnej správy týmto centrom poskytovanou službou. S využitím IKT je možné centrá poskytujúce potrebné služby lokalizovať s ohľadom na potreby klienta efektívne. Takéto centrá môžu predstavovať aj stánky, lokalizované bližšie ku klientovi alebo aj osobné počítače na klientovi najbližšom úrade. Dnes sú zvyčajne poskytované len atomické služby (limitovaná integrácia atomických služieb). Užívatelia majú problém tak s lokalizáciou ako aj s kombináciou viacerých služieb pri komplikovanejších situáciách.

3.1 Občania a podnikateľské subjekty

Access-eGov [1] sa zameriava na zvýšenie prístupnosti služieb verejnej správy pre užívateľskú skupinu občanov ako aj pre užívateľskú skupinu podnikateľov podporou interoperability medzi existujúcimi elektronickými ako aj „tradičnými“ administratívnymi službami. Pre obe skupiny užívateľov bude Access-eGov poskytovať dve základné kategórie služieb. Ako prvé bude Access-eGov identifikovať – podľa potrieb a kontextu situácie používateľa (lokalizáciu a pod.) – tradičné a/alebo elektronické služby verejnej správy (ak sú k dispozícii) relevantné ku danej životnej situácii (daného používateľa občana) alebo obchodný prípad (v prípade podnikateľov). Táto úloha znamená nájdenie preddefinovaného cieľa, ktorý bude na základe sémantického porovnania s so nahradený príslušnou elektronickou službou. Ako druhé, po tom čo boli relevantné služby identifikované, bude Access-eGov generovať „scenár“ pozostávajúci z elementárnych služieb verejnej správy. Vo väčšine prípadov bude mať tento scenár hybridnú podobu – napr. kombinácia elementárnych tradičných a elektronických služieb – ktoré povedú ku požadovanému výstupu. Access-eGov bude tiež poskytovať virtuálneho personálneho asistenta, ktorý

bude prevádzať užívateľa cez scenár (pripomínať mu termíny, poskytovať podporné informácie, inicializovať elektronické služby, atď.). Pre garanciu, že Access-eGov bude prístupný a podstatne užitočný aj skupinám znevýhodnených užívateľov, je špeciálna pozornosť v projekte venovaná kritériám elektronickej inklúzii (e-Inclusion).

3.2 Inštitúcie samosprávy

Technologické riešenie projektu Accessegov [2] je zamerané na podporu sémantickej interoperability medzi novými a existujúcimi elektronickými službami. Pre inštitúcie verejnej správy (na všetkých úrovniach) poskytne projekt možnosť jednoduchého zavedenia nových elektronických služieb. Verejná služba, ktorá bude registrovaná v systéme Access-eGov, bude môcť byť lokalizovaná, koncentrovaná a prostredníctvom agentov a iných IT komponentov aj automaticky používaná. Výhoda takéhoto riešenia spočíva najmä vo zvýšení dostupnosti služieb, zvýšenej transparentnosti poskytovaných služieb, znížení prevádzkových nákladov pri poskytovaní služieb a uľahčení práce zamestnancov inštitúcií verejnej správy. Je zrejme, že posledne menované je vo všeobecnosti ťažko preukázateľné v číslach. Ide však o problém celého e-Governmentu, ktorý odbúraním jedného druhu práce prináša iný. Projekt Access-eGovernent prinesie pridanú prácou v podobe nutnosti sémantického popisu služieb ako aj v tvorbe základných skeletonov scenárov pre procesné modely typických životných situácií a údržbu oboch. Tieto aktivity, však budú prinášať ovocie mnohým občanom a podnikateľským subjektom a môžu tak odbúrať značnú časť inak nevyhnutných interakcií.

4 Záver

V tomto príspevku je prezentovaný projekt Access-eGov a jeho vzťah ku elektronizácií verejnej správy. Projekt má zvýšiť dostupnosť služieb pre všetky skupiny obyvateľov a podnikateľské subjekty a bude zlepšovať transparentnosť týchto služieb. Access-eGov systém je navrhnutý tak, aby bolo možné využiť už existujúce elektronické služby pričom má podporovať aj tradičné služby verejnej správy. Z týchto služieb bude možné vytvárať služby zložené. Existujúce presvedčenie, že elektronizáciou, pre ktorú projekt vytvára vhodné prostredie, budú znížené prevádzkové náklady pri poskytovaní služieb verejnej správy. To môžme v našom prípade spojiť so zjednodušením prístupu služieb, ich interoperabilitou a uľahčením získavania potrebných informácii pre občanov a podnikateľov po zavedení systému do praxe. Tým sa odbúra časť príslušnej práce inštitúcií. Na druhej strane však pribudne práca zviazaná so sémantickým popisom služieb verejnej správy. Zámerom projektu je, aby táto činnosť v konečnom dôsledku znamenala zjednodušenie a uľahčenie práce zamestnancov inštitúcií verejnej správy.

5 Poďakovanie

Tento príspevok vznikol vďaka podpore projektu FP6-2004-27020 Access-eGov (Access to e-Government Services Employing Semantic Technologies) a projektu č. 1/4074/07 Metódy anotovania, vyhľadávania, tvorby a sprístupňovania znalostí s využitím metadát pre sémantický popis znalostí.

Referencie

1. Význam projektu pre občanov/podnikateľov - http://www.accessegov.org/acegov/web/sk/index.jsp?id=50146
2. Význam projektu pre samosprávy – http://www.accessegov.org/acegov/web/sk/index.jsp?id=50145
3. Access-eGov User requirement analysis & development / test recommendations. Deliverable D2.2, Access-eGov Project, 2006
4. Access-eGov State of the Art Report. Deliverable D2.1, Access-eGov Project, 2006

Service-based architecture of Access-eGov system

Martin Tomášek, Karol Furdík
InterSoft, a. s., Floriánska 19,
040 01 Košice, Slovakia
{Martin.Tomasek, Karol.Furdik}@intersoft.sk

Abstract. The paper describes functional architecture of Access-eGov system – a platform for integration of e-Government services on a semantic level. Design follows principles of peer-to-peer and service-oriented architectures, with implementation of Web Services. The overview of main system components is presented, together with the brief functional description.

Keywords: e-Government, Semantic Web, ontologies, annotation, service-oriented architecture.

1 Access-eGov system – objectives and functionality

The *Access to e-Government Services Employing Semantic Technologies* (Access-eGov) is an international EU IST project No. FP6-2004-27020. It has started in January 2006 and it is planned for 36 months. The main objective of the project is to develop a common platform for integration of e-Government services, based on Semantic Web technologies and distributed architectures.

The two basic categories of services will be provided for citizens and business users of the Access-eGov system. Firstly, it is a meta-service, which will identify proper e-Government services relevant to the given life event or business episode, according to the (semantically expressed and described) user's needs and requirements [1]. Secondly, after finding relevant services, the Access-eGov system will generate a scenario consisting of elementary government services. These services can be of a "hybrid" nature, i.e. a combination of traditional and electronic e-Government services. A specialized interface, named as Virtual personal assistant, is intended to guide users in the space of available services.

2 Architecture design

The requirement of integration of the existing e-Government services implies a modular, extensible, distributed, and flexible architecture composition. A matrix approach was chosen for designing the architecture in a structured and formalized way [2]. The matrix consists of four views (columns) and four abstraction levels (rows) in order to compose a coherent description framework. The framework addresses four distinct views: *Security*, *Functionality*, *Data* (as well as metadata), and

Structure. There are strong interdependencies between them, since they represent complementary views. Four levels of abstraction are addressed, namely: *Context*, *Conceptual*, *Logical*, and *Physical* levels. Based on this framework, the requirements were specified for each of matrix fields – a description of an abstraction level based on each of views. The key decision made at each of levels was to select the most adequate solution alternative that delivers the required services, and best addresses the guiding principles.

Concerning the interoperability between different types of legacy systems used within public service backend, an implementation of Web Services in conjunction with service-oriented and peer-to-peer architectures was chosen as the most promising approach. The approach based on these technologies will prove far more feasible, but will be challenging since research is still necessary in this field of distributed application architecture. The figure below (Fig. 1) gives an overview of the system components.

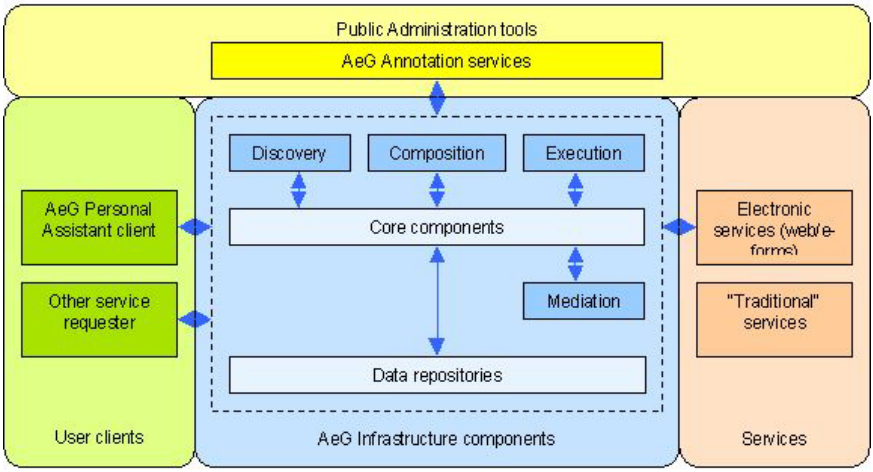


Fig. 1. Overview of the Access-eGov system components [2].

The overall Access-eGov system may be sub-divided into three major component groups:

- the AeG Infrastructure itself,
- the AeG Personal Assistant client and corresponding end-user interfaces,
- AeG Administration and Management Tools (e.g. Annotation services), which are not integral parts of the AeG Infrastructure, but are affiliated to it.

The actual services are still hosted under responsibility and on the premises of users, which are participating public agencies or their respective data centres. They are simply made available through Access-eGov system, and thus do not form an integral part of the system itself. The services are either electronically available (directly via web service interfaces or web forms) or represent “traditional” office

services that may merely be described and registered within AeG. Only executable services will dispose of an electronic XML-interface to the AeG Infrastructure.

Public agencies are supposed to annotate those services that they are willing to expose to the public. These kinds of service-related metadata will be transferred to the Persistence layer via executable Core components. Therefore, domain experts may use a generic Annotation service component, available as web-based application.

The Core components consist from in-memory object model for semantic web services ontologies, web service entry point that makes infrastructure components available to Personal Assistant and Annotation services (Connection manager), security and notification services. These components are mediated on the data level to avoid possible data heterogeneity problems [3]. The mediation will be based on mappings expressed with the mapping ontology and designed for the mapped domain ontologies.

All the necessary system data are stored in the persistence layer and are accessible by unified API. Several repositories were identified according to the type of stored data, namely:

- *life events / goals repository* for managing goals and generic scenarios associated with the life events,
- *service repository* for storage of descriptions of the e-Government services registered for the specific already orchestrated scenarios,
- *ontology repository* containing domain ontologies and associated mappings for mediators,
- *process context repository* for storage of the context (state) of the processes executed for the orchestrated life event scenarios,
- *security data repository* containing user login information and access rights.

Discovery, Composition, and Execution modules are responsible for semantic matching of goals and services, its dynamic composition into complex sequences, and invoking composed services of orchestrated scenarios.

On the administration side, the Annotation services are included into a web-based tool providing functionality for semantic description of particular services, for management of life events and goals.

The set of specific domain ontologies will be used to represent the functional and non-functional properties of a particular service [4]. Currently it is assumed that the system will need ontologies for *Fees*, *Forms*, *Input and output artefacts*, and *Administration*. Each domain ontology may have mappings to other ontologies stored in the Ontology repository (a part of Data repository). This way, an ontology can have M:N-relations to other entries in Ontology repository. All the ontologies used by Access-eGov platform will be stored in persistent repositories that are accessible to all peer-instances. The ontologies will consist of a core set of public service concepts to sufficiently describe services. The actual ontologies that are used for annotation process and for lookup during automated service retrieval will be provided by the respective public service provider.

The AeG Personal Assistant accesses AeG Infrastructure functionality via standardised interfaces and communicates with executable Core components that are charged with Discovery, Composition and Execution of registered public services. All communication goes through these Core components in order to gain access to persistently held data. The Personal Assistant client is a thin web-based client that

provides a functionality of user and profile management, scenario execution management, and life events / goals discovery.

3. Conclusion, future work

The modular architecture of Access-eGov system, based on web services, was briefly described in the paper. After the detailed analysis of available resources, the WSMO / WSMX was chosen as the most promising platform and environment for implementation. In the next phase of the project, this choice will be proven on the three pilot applications – in the Public Administration organizations from Germany, Poland, and Slovakia.

Acknowledgments. The Access-eGov IST project No. FP6-2004-27020 is co-funded by European Commission.

References

1. Klischewski, R., Ukena, S.: D2.2 User requirement analysis & development / test recommendations. Public deliverable of the Access-eGov project, FP6-2004-27020. Germany University in Cairo (2006)
2. Schillinger, R., Duerbeck, S., Mach, M., Bednar, P., Hreno, J.: D3.1 Access-eGov Platform Architecture. Technical report of the Access-eGov project, FP6-2004-27020. University of Regensburg (2006)
3. Tomasek, M., Paralic, M., Furdik, K., Schillinger, R., Duerbeck, S., Mach, M., Bednar, P., Hreno, J.: D3.2 Access-eGov Components Functional Descriptions. Technical report of the Access-eGov project, FP6-2004-27020. InterSoft, a.s., Kosice (2006)
4. Ukena, S., Klischewski, R.: D7.2 Guidelines for Semantic Mark-Up of e-Government Resources. Technical report of the Access-eGov project, FP6-2004-27020. Germany University in Cairo (2006)

Knowledge management

Aké charakteristiky jednotlivca pri hľadaní vo veľkom informačnom priestore dokážeme zachytiť automaticky? Koľko ich potrebujeme?

Michal Barla and Mária Bieliková

Institute of Informatics and Software Engineering,
Faculty of Informatics and Information Technologies,
Slovak university of Technology in Bratislava, Slovakia
{barla,bielik}@fiit.stuba.sk

1 Jednoduché vs. komplexné modely používateľa

Aktuálne trendy v modelovaní používateľa pre adaptívne webové systémy môžeme rozdeliť do dvoch skupín. Na jednej strane vznikajú stále komplikovanejšie modely, ktoré sa snažia pokryť mnoho aspektov používateľa čoraz komplexnejšími formalizmami (napr. ontológia GUMO [3]) a na druhej strane je viacero (nasadených) systémov, ktoré dokážu svojim používateľom poskytnúť služby personalizácie, odporúčania bez potreby rozsiahlych modelov (napr. s využitím profilov). V mnohých prípadoch je tento model dokonca latentný, vyjadrený implicitne. Nevýhodou oboch skupín systémov je ich uzavretosť – iba sám systém vie interpretovať použitý model používateľa, resp. vie ako personalizovať svoj obsah. Komplikovanejšie modely sa používajú v menšej miere najmä kvôli problému ich automatického napĺňania.

2 Správanie skupín používateľov

Často sa ako základ dát o používateľoch zoberie záznam webového servera, nad ktorým sa potom aplikujú techniky dolovania označované ako *web usage mining* [1,2]: objavovanie znalostí – zhlukovanie, klasifikácia, objavovanie asociácií alebo sekvenčných vzorov v záznamoch o správaní sa skupín používateľov. Výsledky, ktoré tieto metódy dávajú však nemôžeme pokladať za charakteristiky používateľa. Predstavujú vzory správania, ktoré môžu však poskytnúť dobrý odhad toho čo používateľ hľadá, aké sú jeho záujmy, najmä vzhľadom na väčšiu skupinu používateľov.

Tieto prístupy často slúžia ako podklad pre profiláciu typického návštevníka webu (webového informačného systému) a lepšie prepojenie stránok. Existujú prístupy, ktoré takto objavené znalosti používajú pre odporúčanie (napr. stránok, ktoré najčastejšie navštevujú používatelia patriaci do toho istého zhluku ako aktuálny používateľ). V tejto oblasti sa dajú využiť napríklad aj umelé neurónové siete či genetické algoritmy a iné tzv. *soft computing* techniky [4]. Systém však väčšinou nie je zameraný na jednotlivca, ale na skupinu používateľov.

3 Od charakteristík skupiny k jednotlivcovi

Pre napĺňanie modelu používateľa môžeme použiť aj heuristické metódy, ktoré odhadujú charakteristiky používateľa na základe vykonaných akcií [5] a so znalosťami o interpretácii týchto akcií (napr. ak používateľ vyhodnotí dve pracovné ponuky veľmi rozdielne a tieto sa líšia iba v mieste vykonávania práce, možno z toho usudzovať, že je miesto výkonu práce pre používateľa dôležitá charakteristika a vieme aj povedať, ktorá je vhodná a ktorá nevyhovuje). Tento spôsob nie je celkom možné úplne abstrahovať od použitej domény, ale pridaním znalostí o doméne sa dokážeme dostať na úroveň jednotlivca a jeho doménových preferencií. V prípade takéhoto modelu sa však komplikuje aj jeho využitie pri personalizácii. V špecifických doménach možno použiť aj ďalšie spôsoby. Napr. v doméne pracovných ponúk je zaujímavým spôsobom získania charakteristík používateľa multikriteriálna analýza vykonaná na základe vyhodnotenia určitého počtu pracovných ponúk používateľom na definovanej škále, pričom sa následne analyzujú hodnoty jednotlivých atribútov týchto ponúk.

V systémoch pre vzdelávanie sa získavajú charakteristiky aj vykonaním testov, čím sa odhaduje úroveň vedomostí prezentovaných používateľovi/študentovi. Túto metódu však nemožno považovať za automatické získanie charakteristík, keďže je dopredu dané čo sa ktorou odpoveďou zistí.

Keďže model používateľa je len modelom, nie je možné očakávať, že bude zhodný s reálnymi charakteristikami používateľa. Proces odhadovania cieľov a charakteristík používateľa je náročný a zrejme bez zásahov samotného používateľa ani nie je možný. Otázkou je do akej miery sme schopní automaticky odhadovať a koľko vlastne potrebujeme preto, aby informačný systém poskytoval informácie v súlade s očakávaniami používateľa (je toto vôbec možné?).

4 Zhrnutie

Čiastočné odpovede na položené otázky vznikajú v rámci riešenia projektu NAZOU [6], časti, ktorá sa zaoberá personalizovanou prezentáciou dát a ponúk. Ukazuje sa, že pomerne jednoduché heuristiky (napr. analyzovanie charakteristík ponúk, na ktoré používateľ klikol ako prvé) zabezpečia veľmi dobré správanie. Pridávanie ďalších heuristík ho už zrejme nezlepší natoľko. Považujeme za dôležité diskutovať tieto otázky aj v kontexte vyhodnotenia objavených charakteristík a získania spätnej väzby o kvalite odhadu.

Ukazuje sa, že prístup založený na heuristikách, pravidlách môže byť s minimálnym úsilím znovupoužiteľný vo viacerých doménach (menia sa iba pravidlá, nie ich interpretácia). Predpokladom na znovupoužitie je:

- vhodný navigačný model – model musí poskytovať používateľovi určitú voľnosť a slobodu výberu, aby sa v navigácii mohli prejaviť používateľove preferencie, záujmy a pod. Navigačný model musí umožňovať definíciu zmysluplných pravidiel správania sa používateľov;
- konzistentný prístup k tvorbe modelu používateľa – interpreter je znovupoužiteľný vo viacerých doménach vtedy, keď sa v týchto doménach zvolí

podobný prístup k reprezentovaniu charakteristík používateľa. V projekte NAZOU oddeľujeme doménovo špecifický a doménovo nezávislý model používateľa. Jednotlivé charakteristiky vznikajú ako inštancie príslušných tried (typov) charakteristík. Interpreter teda môže pracovať s modelom používateľa na vyššej úrovni pričom má k dispozícii opis, ktorý mapuje všeobecné koncepty modelu na aktuálne použitý doménovo špecifický model.

Poďakovanie

Táto práca vznikla za čiastočnej podpory Agentúry na podporu vedy a techniky (APVT-20-007104) a štátneho programu pre výskum a vývoj "Budovanie informačnej spoločnosti" (1025/04).

Referencie

1. Pierrakos D. et al.: Web Usage Mining as a Tool for Personalization: A Survey. In User Modeling and User-Adapted Interaction, 13(4), pp. 311–372,Springer, Netherlands, 2003
2. Eirinaki, M., Vazirgiannis, M.: Web Mining for Web Personalization. In ACM Transactions on Internet Technology (TOIT), 3(1) pp. 1–27. ACM Press, New York, NY, USA, 2003.
3. Heckmann D et al.: GUMO – the general user model ontology. In Ardis-sono L. et al. (Eds) 10th Int. Conf. on User Modeling, Edinburgh, UK, Springer, LNCS 3538, pp 428-–432, 2005.
4. Friaz-Martinez, E. et al.: Modeling human behavior in user-adaptive systems: Recent advances using soft computing techniques In Expert Systems with Applications 29 (2) 320–329.
5. Pohl, W., Kobsa A., Kutter, O.: User Model Acquisition Heuristics Based on Dialogue Acts, In 4th International Conference on User Modeling, pp. 225–226, MITRE Corporation, Bedford, MA, USA, 1994.
6. Návrat, P. et al. (Eds.). Tools for Acquisition, Organization and Presenting of Information and Knowledge. Proceedings. Research Project Workshop, Bystrá Dolina, Low Tatras, Slovakia, Sept. 2006, 256p.

Personalizovaná navigácia pre podporu vyhľadávania a pochopenia informácií v priestore webu so sémantikou

Michal Tvarožek, Mária Bieliková

Ústav informatiky a softvérového inžinierstva,
Fakulta informatiky a informačných technológií,
Slovenská technická univerzita
Ilkovičova 3, 842 16 Bratislava
bielik,tvarozek@fiit.stuba.sk

1 Existujúce problémy

Práca s informáciami predstavuje v súčasnosti významnú náplň práca mnohých ľudí – používateľov informačných systémov. Množstvo prístupných a spracúvaných informácií exponenciálne narastá, pričom už v súčasnosti samotná veľkosť dostupného informačného priestoru spôsobuje mnohé problémy. Tieto nastávajú na rôznych úrovniach spracovania a práce s informáciami.

Z pohľadu vyhľadávania je náročné modelovať informácie v mnohých informačných doménach a efektívne vyhľadať tie, ktoré zodpovedajú dopytu, ktorý často nemusí byť celkom presný. Na druhú stranu, z pohľadu používateľov, je napr. náročné formulovať dopyt do niektorého z vyhľadávacích strojov a následne nájsť požadované informácie vo výslednom dokumente, resp. na webovej stránke, ktorá je výsledkom vyhľadávania.

V praxi sa problémy prejavujú rôzne. „Stratenie sa“ používateľa v hyperpriestore, časová náročnosť vyhľadávania a práce s informáciami, formulácia dopytov pre vyhľadávacie stroje alebo nájdenie požadovaných informácií v dokumentoch predstavujú len niekoľko problémov, ktoré je nevyhnutné riešiť.

2 Aktuálne prístupy

Klasický web tvoria dokumenty s informáciami, ktoré opisujú najmä spôsob ich prezentácie používateľovi. S cieľom umožniť automatické spracovanie informácií, ktoré sa stáva s nárastom ich množstva nevyhnutnosťou, vznikla vízia webu so sémantikou, ktorý opisuje aj význam – sémantiku jednotlivých informácií. Nadstavbu webu so sémantikou predstavuje web dôvery, ktorý k významu informácií pridáva aj znalosti o pravdivosti, resp. dôveryhodnosti zdrojov týchto informácií.

Proces vyhľadania, prístupu a spracovania informácií sa typicky začína formulovaním dopytu, pomocou ktorého sa z celého informačného priestoru získa relevantná časť informácií (napr. dokumentov alebo webových stránok). Následne pokračuje navigáciou vo výsledkoch vyhľadávania a výberom vhodných dokumentov [1]. Po nájdení výsledného dokumentu proces pokračuje hľadaním

cieľových informácií v ňom, pričom záverom procesu je až nájdenie a spracovanie cieľových informácií.

Samotná dostupnosť technických prostriedkov (napr. vyhľadávacích strojov, formalizmov, algoritmov), ktoré budú využívať nové metódy vyhľadávania a práce s informáciami neznamená automatické vyriešenie uvedených problémov. Nutným predpokladom vyriešenia problémov je až schopnosť používateľov jednoducho a efektívne využívať dostupné prostriedky na vykonanie jednotlivých častí procesu vyhľadania a práce s informáciami:

- Formulácia dopytu pre vyhľadávací stroj, ideálne so sémantikou.
- „Spracovanie" výsledkov vyhľadávania a nájdenie množiny cieľových zdrojov informácií (dokumentov, webových stránok).
- Nájdenie a následné spracovanie cieľových informácií vo výslednom dokumente [2].

Riešenie uvedených problémov v kontexte webu so sémantikou, resp. webu dôvery vyžaduje nové metódy v prístupe k informáciám, kedy navigácia iba pomocou explicitne uvedených odkazov v texte je nepostačujúca. Úlohu vyhľadávania informácií zjednodušujú vyhľadávače a meta-vyhľadávače, ktoré nájdu webové stránky, ktoré považujú za relevantné vzhľadom na daný dopyt a ďalej sa znovu pokračuje navigáciou v explicitne uvedených odkazoch v dokumente (webovej stránke).

Keďže vízia webu so sémantikou predpokladá opis informačných zdrojov ontológiami, hľadanie metód efektívnej navigácie v ontológiách a jej personalizácia predstavuje perspektívny prístup k riešeniu uvedených problémov vo všetkých troch častiach procesu práce s informáciami. Využitie ontológií za účelom sémantického vyhľadávania [3,4,5] vyžaduje formuláciu dopytov do ontológie s využitím sémantiky informácií, čo pre „bežných" používateľov prestavuje netriviálnu úlohu. Vhodná vizualizácia informačného priestoru a navigácia v ňom s podporou pre personalizáciu môže výrazne uľahčiť tvorbu komplexných dopytov.

3 Navrhnutá metóda

Identifikovali sme niektoré metódy ako realizovať efektívnu navigáciu v rámci spomínaných „nových metód" v prístupe k informáciám, ktoré vychádzajú z opisu prezentovaných informácií ontológiou, kedy okrem siete informácií máme aj sieť metadát a hľadáme efektívne metódy navigácie v takomto priestore. Navigáciu vo výsledkoch vyhľadávania a prípadné zjemnenie dopytu je možné realizovať pomocou fazetového prehliadača [6] s pokročilým rozhraním pre definovanie dopytov, rozšíreného o podporu ontológií a personalizácie na základe modelu používateľa [7].

Táto metóda vychádza z myšlienky ohraničovania priestoru na základe atribútov prezentovaných inštancií. Fazety a ohraničenia vyberá používateľ a v existujúcich prístupoch sú definované vopred a to zvyčajne na základe priamych vlastností jedného typu inštancií. Príspevkom k navigácii je dynamické generovanie faziet na základe dopytu, údajov z doménovej ontológie a ontológie

používateľa. Keďže fazety sú generované počas práce používateľa nie je potrebné ich vopred definovať, čo zlepšuje prácu v dynamicky sa meniacom prostredí obsahujúcom vopred neznáme typy informácií. Náš prístup súčasne podporuje generovanie faziet aj na základe nepriamych vlastností inštancií, t.j. vlastností inštancií iného typu, ktoré sú s hľadanými inštanciami v nejakom vzťahu, čo umožňuje používateľovi efektívnejšie formulovať vlastnosti hľadaných inštancií.

Pri procese generovania faziet identifikujeme priame a nepriame vlastnosti inštancií, určíme ich relevanciu napr. na základe modelu používateľa alebo pomocou techník kolaboratívneho filtrovania. Následne skonštruujeme opis novej fazety na základe typu konkrétnej vlastnosti a zaradíme ju do množiny používaných faziet.

Pre efektívnejšiu identifikáciu a výber cieľových dokumentov z výsledkov vyhľadávania navrhujeme použitie adaptácie rozhrania tak, aby prehľad výsledkov vyhľadávania obsahoval informácie, ktoré používateľovi pomôžu pri výbere cieľových dokumentov. Adaptívne zobrazenie pre používateľa relevantných atribútov na základe modelu používateľa a anotácia jednotlivých výsledkov umožnia používateľovi lepšie sa rozhodnúť pri výbere smeru ďalšej navigácie.

Identifikovanie tých častí výsledného dokumentu, ktoré obsahujú požadované informácie a navigácia v nich predstavuje samostatný problém. Dôležitou je nielen formulácia dopytu (prebraná z predchádzajúceho vyhľadávania), ale najmä spôsob prezentácie výsledkov a navigácie v samotnom dokumente. Zaujímavé je automatické identifikovanie vybraných častí dokumentu, ich zobrazenie spolu s prípadnou anotáciou a navigácia medzi nimi na základe odvodených odkazov, resp. odporúčanie súvisiacich alebo relevantných častí dokumentu.

4 Zhrnutie

Opísali sme existujúce problémy v oblasti vyhľadávania a spracovania informácií ako aj problém navigácie vo veľkých informačných priestoroch. Medzi aktuálne prístupy k riešeniu uvedených problémov v oblasti patria web so sémantikou a ontológie, fazetové prehliadače ako aj personalizácia a adaptívne webové systémy.

Navrhli sme rozšírenie a kombináciu existujúcich prístupov na zlepšenie práce používateľa s veľkými informačnými priestormi pomocou rozšírenia klasického fazetového prehliadača o práce s ontológiami a adaptáciu na základe modelu používateľa.

Pomocou dynamického generovania faziet a rozšíreného režimu tvorby dopytov umožňujeme používateľovi jednoduchú a efektívnu vizuálnu tvorbu dopytov do ontológie prostredníctvom navigácie v nej. Zlepšenie prístupu a pochopenia informácií dosahujeme personalizáciou faziet a pohľadov na výsledky vyhľadávania ako aj identifikáciou a zvýraznením hľadaných informácií v cieľovom dokumente.

Poďakovanie. Táto práca vznikla za čiastočnej podpory Agentúry na podporu vedy a techniky na základe zmluvy č. APVT-20-007104 a štátneho programu

pre výskum a vývoj „Budovanie informačnej spoločnosti“ na základe zmluvy č. 1025/04.

Referencie

1. Lavene, M., Wheeldon, R.: Navigating the World-Wide-Web. In Lavene, M., Poulovassilis, A., eds.: Web Dynamics, Springer (2003)
2. Domingue, J., Dzbor, M., Motta, E.: Magpie: Supporting Browsing and Navigation on the Semantic Web. In: Intelligent User Interfaces. (2004) 191–197
3. Guha, R., McCool, R., Miller, E.: Semantic Search. In: The 12th International Conference on World Wide Web. (2003) 700–709
4. Zhang, L., Yu, Y., Zhou, J., Lin, C.X., Yang, Y.: An Enhanced Model for Searching in Semantic Portals. In: WWW 2005, ACM Press (2005) 453–462
5. Shah, U., Finin, T., Joshi, A.: Information retrieval on the semantic web. In: CIKM '02: Proceedings of the eleventh international conference on Information and knowledge management, New York, NY, USA, ACM Press (2002) 461–468
6. Instone, K.: How user interfaces represent and benefit from a faceted classification system. In: SOASIS&T. (2004)
7. Tvarožek, M.: Personalized Navigation in the Semantic Web. In Wade, V., Ashman, H., Smyth, B., eds.: AH 2006, LNCS 4018, Springer-Verlag Berlin Heidelberg (2006) 467–471

Ontology based knowledge system for administrative workflows management

Ivana Budinská, Zoltán Balogh, Emil Gatial, Michal Laclavík, Igor Mokriš, Radoslav Forgáč, Ladislav Hluchý, Viktor Oravec, Martin Šeleng

Institute of Informatics SAS, Dubravska cesta 9, Bratislava, Slovakia

budinska@savba.sk

Abstract. The paper describes an ontology approach to the modeling of administrative workflow processes. The workflow process is described as a set of activities. The administrative character of workflow enables to describe each activity via input and output documents. The activity is invoked when all required input documents are completed. All documents' descriptions are also stored in the ontology. The ontological approach is applied on the RAPORT system. The description of architecture and basic principles of the RAPORT system is provided in this paper.

1 Introduction

Recent research in workflow management for organizations with administrative type of processes is focused on the application of intelligent information technologies on the basis of workflow management systems. In the systems, intelligent knowledge base is used to represent knowledge within organizations; thus, response to environmental changes is allowed on the basis of historical situations, and also according to employees' knowledge and experiences captured within the modeled organization. Such approach enables to develop systems that can resolve current problems and offer expert view on the current problem solution. There exist a lot of software tools to support workflow management. Many organizations with administrative processes have couple of works to carry out that can be formalized and structured according to internal directives and instructions. It is not required to use the complex workflow management system in such organizations, because there are just very few well defined processes. The development of IT allows employees of many organizations to use the Internet, intranet, e-mails, etc. during their work. The paper describes a system based on e-mail communication and a web portal that is suggested for management of workflow processes in administrative organizations. The core of the system is an ontology that contains organization model, activity model and data model. The sequence of activities is ontology driven. The next activity is invoked when all input conditions that are represented by approved or finalized documents, are completed. Big advantage of such solution is in relatively easy implementation on the assumption that the organization has e-mail

communication and internet access established. Employees are not required to learn working with another type of software. Web portal provides all necessary information about workflow processes and e-mail communication enables automatic or semi-automatic notification about deadlines, required actions, and changes in plans, etc.

The ontology based workflow management improves administrative processes in several ways, i.e. improves the quality of work, saves the processing cost, proposed system can advice user with actions to be taken and suggest user with solutions which are related to current user's working process and time savings. This kind of administrative work requires the broad knowledge of rules (laws, internal instruction, etc.) as well. The system based on ontology can encapsulate such knowledge; moreover it can capture the solutions of past processes and propose the most proper one. The users also will have overview upon the personal assignments of current working processes, so the users of system evade the processing of the same thing twice. Moreover, the proposed system will inform user about the progress of current workflow activity and about the pending processes.

2 Architecture

The architecture of the knowledge system is designed in the generic way that can work for arbitrary administration process. It also takes care about administrative process in the CST. The system architecture comes from the following requirements:

1. collect experience from users and present useful experience to other users that works in the same or similar work context,
2. keep an eye on current training plans that contain of important deadlines, alarm users about that;
3. prepare for users at deadlines necessary information such as predefined emails, documents, formulas and let users know about that,
4. support user's experience exchange and collaboration.

A 3-tier layer architecture consists of: data layer, process layer and presentation layer. As in several current projects, important background of the system design is its ontology, which defines structure and relationships among experience entities; ontology is the main mechanism used for the representation of information and knowledge, definition of the meaning of the terms used in the content language and the relation among these terms.

Data layer: Data layer ensures functionality of the database and the file system repository. Information and knowledge are organized and maintained for the use of the process and presentation layer. The generic, domain-specific and user-model ontology are also stored in the data layer in OWL format and in database. Predefined structure of such as documents, formulas, templates and predefined plans of training activities are stored here. Such information and data can be managed and modified by the system administrators.

Process layer: is the heart of the system. It functionalities are designed to fulfill the system requirements. The functions of the process layer can be briefly listed as follows:

1. ensure event monitoring and notification,
2. information analysis,
3. creating of ontological data based on accessible resources and context, storing them through Data organization and maintenance as experience for future use.
4. creating active notes based on the current working context and previous experiences in the data layer and providing them to users through the presentation layer
5. preparing required information (documents, emails) related to the current working activity and working context according to the predefined templates, structures and accessible knowledge

Presentation layer: provide transparent, user-friendly and adaptable middleware for presentation knowledge and information to users. To fulfill this purpose, the presentation layer describes the user model with user ontology in the background. The user ontology is generic personal ontology that is firm extended for the army environment. The user ontology is already well-integrated with the domain-specific ontology and is accessible from the data layer.

Based upon these requirements the personal, communication, function, data and process model of workflow management system was developed [1, 2].

Conclusion

The approach of the knowledge management for the design of the system architecture in the administrative process is applied in a process of military exercise preparation within a project RAPORT [3]. Research is build based on the long-term work in this research area within EU RTD IST project “A Platform for Organizationally Mobile Public Employees” [4, 5], etc. The idea of active notes/advice comes from the DÉCOR [6]. The difference and new in the presented system is the combination of workflow management system in the administrative processes with email processing [7, 8], that is more lightweight and has found wider usage in small and medium sized enterprises and in administrative organizations as well.

This work is supported by Slovak national projects APVT-51-024604, APVV LPP-0231-06 SPVV 1025/2004, VEGA 2/6103/6, VEGA 2/7098/27 and EU RTD IST project K-Wf Grid FP6-511385.

References

1. Mokriš, I., Forgáč R.: Utilization of Knowledge System for Modeling of Administrative Processes. Proc. of Conf. „Communication and Information Technologies 2005“, ISBN 80-8040-269-8, Tatranské Zruby, Nov.23. – 25, 2005, pp. 95-98, (in Slovak).

2. Nguyen G., Balogh Z., Laclavik M., Gatial E., Hluchy L., Arenas, A.: Ontology-Based Experience Management for Public Organizations. 8th Int. Conf. on Business Information Systems BIS'2005, ISBN 83-7417-094-8, April 2005, Poznaň, pp. 217-227
3. Research and Development of a Knowledge Based System to Support Workflow Management in Organizations with Administrative Processes (APVT-51-024604) http://raport.ui.sav.sk.
4. Laclavik M., Balogh Z., Hluchý L., Krawczyk K., Dziewierz M., Kitowski J., Majewska M.: Knowledge Management for Administration Processes. Proc. of Znalosti ´4, VŠB-TU Ostrava, ISBN 80-248-0456-5, 2004, pp.248-255.
5. Laclavík, M., Gatial, E., Balogh, Z., Habala, O., Nguyen, G., Hluchý, L.: Experience Management based on Text Notes (EMBET). In. Cunnigham "Innovation and the Knowledge Economy: Issues, Applications, Case Studies". Amsterdam, IOS Press, 2005. ISBN 1-58603-563-0, pp. 261-268.
6. Laclavík, M., Gatial, E., Balogh, Z., Habala, O., Nguyen, G., Hluchý, L.: Semantic Annotation based on Regular Expressions. In. ITAT 2005, UPJŠ University Košice, ISBN 80-7097-609-8, 2005, pp. 305-306.
7. Martin Seleng, Michal Laclavik, Zoltan Balogh, Ladislav Hluchy
   Automated Content-based Message Annotator - ACoMA
   In: Proceedings of ITAT 2006 Information Technologies - Applications and Theory, Peter Vojtas (Ed.), Department of Computer Science, Faculty of Science, Pavla Jozef Safarik University, Kosice, 2006, pp.195-198, ISBN 80-969184-4-3

Yet Undiscovered Potential of the e-mail Communication*

Michal Laclavík[1], Martin Šeleng[1], Ladislav Hluchý[1], Jacek Kitowski[2,3]

[1] Institute of Informatics, Slovak Academy of Sciences, Dubravska cesta 9
845 07 Bratislava, Slovakia
[2] Institute of Computer Science, AGH University of Science and Technology,
Krakow, Poland
[3] Academic Computer Centre CYFRONET-AGH, Krakow, Poland
laclavik.ui@savba.sk
http://ikt.ui.sav.sk

Abstract. Email repositories and an email activity are valuable assets for any modern internet based business organization. In order to achieve organizational objectives or to successfully run a business process, goals, tasks or actions in any organization need to be communicated. Communication is important part of a business process and collaboration. When storing knowledge provided by a user, the context of this information or knowledge has to be detected. The context of the user can be represented by the current user task/activity, business process, and other related parameters. In most of internet oriented businesses, email is considered as a primary medium for information exchange, and email also plays a major role in an SME business process. According to statistics, email is the second most used service of the internet after WWW. Therefore email can be considered as a good medium for detection of the user context / problem, business process, task, customer or other related data and thus Email can be a medium for active information and knowledge provision. We also briefly describe the ACoMa tool (Automated Content-based Message Annotator), which provide context sensitive knowledge on EMBET platform. If application domain ontology is available and an email message belongs to described application domain, the tool can find application context in email message and attach relevant information and knowledge into it.

1 Introduction

Email repositories and an email activity are valuable assets for any modern internet based business organization. In order to achieve organizational objectives or to successfully run a business process, goals, tasks or actions in any organization need to be communicated. Communication is important part of a business process and collaboration. According to Habermas's work [1], the categories of communication

* This work is supported by projects K-Wf Grid EU RTD IST FP6-511385, NAZOU SPVV 1025/2004, RAPORT APVT-51-024604, VEGA 2/6103/6, VEGA 2/7098/27.

goals in organizations are: Commanding a specific action; Managing Collective Action; Influencing; Providing Information for Future Action; and Seeking Information for Future Action. Therefore business email communication is action oriented, communication is clear and short and thus partially understandable for computers after text processing and an analysis.

Building a knowledge management system (KM), the users need and want to have knowledge in a situation when solving a problem, without extensive searching in organizational knowledge. In addition, the user does not have to know whether the organization has some knowledge in the KM system for the current problem, and when collaborating and sharing knowledge, it is necessary to know the current user needs. For this reason the user's current context must be detected. When storing knowledge provided by a user, the context of this information or knowledge has to be detected. The context of the user can be represented by the current user task/activity, business process, and other related parameters. In most of internet oriented businesses, email is considered as a primary medium for information exchange, and email also plays a major role in an SME business process. According to statistics, email is the second most used service of the internet after WWW. Therefore email can be considered as a good medium for detection of the user context / problem, business process, task, customer or other related data.

In the KM system it is important to answer the question "How can we best keep knowledge dynamic, use it in action-oriented situations, and make it the backdrop for creativity?" The answer is through e-mail, the quintessential Internet application. Consider the following [2]:

- Any organization, without exception, will have an e-mail infrastructure before it reaches the stage of developing an organizational memory (OM).
- E-mail communication in a modern organization is over 78% action-oriented, according to a recent study [3]. Organizations must converge to action, and communication is perhaps the foundation for most organizational action [4].
- Managers, and knowledge workers of all kinds, interact with their e-mail systems on a daily basis - it is a standard operating procedure. This means that using email as the window into an organizational memory gives us the smallest change in an organization's daily activities.
- Managers are motivated to achieve successful communication. They want their instructions to be understood and their answers to queries to be effective.

When building a solution on top of email communication, an organization does not have to change the way of doing its business, when such a solution is installed and set up in the organization. Users simply receive emails in the same way as before, but with attached relevant knowledge to the problem which the email represents.

Some work to connect knowledge with emails has been done in several projects such as the kMail system [2] which integrates e-mail communication with organizational memories, but also forces users to use a special email client and lacks a closed knowledge cycle loop or another project Gmail[1], a new kind of webmail

[1] http://gmail.com

developed by Google, which shows content sensitive advertisements with the email or offer some content sensitive actions such as adding events to calendar (see Figure 1). By contrast to Gmail, this approach intends to connect emails with similar hints based on organizational knowledge, which are linked to organizational resources and systems.

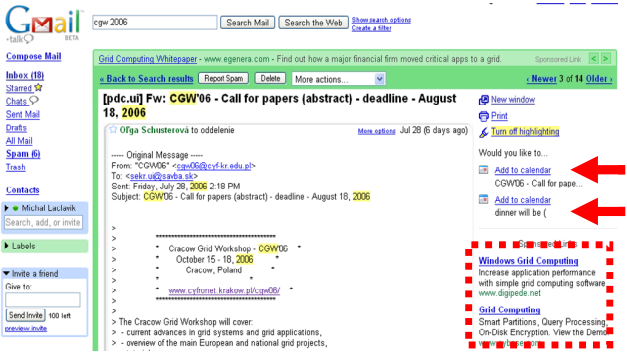


Fig. 1. Gmail webmail offers content sensitive actions and advertisements.

1.1 Objectives

The overall objective is to propose an adaptable platform for e-Collaboration and knowledge sharing in an SME. In order to meet the needs arising from the motivation just outlined, it will be necessary to provide:

- a continuously growing shared memory for the information, knowledge and experience, which is provided to a user in a pro-active manner
- email processing for context acquisition and return of knowledge;
- Mechanisms for continuous update of the knowledge base, so as to 'close the loop' in the cycle of knowledge generation and use.

2 Overview of the Approach

To find the innovative contribution the email processing system can give to e-collaboration it is necessary to consider several points:

- collaboration of workers with different expertise in an SME, needs not only tools to provide technical assistance to the networking job, but also a way to share expertise;
- communication is therefore a crucial point in collaboration;
- communication among geographically dispersed groups can be achieved through the use of email;

- email is one of the most powerful conveyors of knowledge as found by the COMMORG[2] project: impact of introduction, adoption and diffusion of email communication on the evolution and change of organizations' structures and processes;

The major innovative aspect of the system can be the introduction of a platform for e-communication that allows communicating with other members of a group with a common ontology and organizational memory thus allowing a smart and fast way of collaborating together.

The aim is also to develop an active knowledge sharing system based on the knowledge embodied in emails. It will thus be a form of Electronic Performance Support System, but going much further by supporting a worker by bringing together workers knowledge in an organization and enhancing their performance and reactiveness through knowledge sharing.

The main innovations can be the following.

- The central role of emails. These are ubiquitous and strongly linked to business processes, though their content is unstructured. The innovation lies in tying email content to business context, so as to analyze and understand the current context and relate it to knowledge in the organizational memory.
- Clustering of organizations. Most work on knowledge sharing has concentrated on enabling it within single organizations. In such system, the use of a shared cluster organizational memory will enable sharing across members of the cluster. This raises challenges of modeling different views on a common but distributed business process.
- A use of email enables to have an "active" knowledge sharing channel, since a user does not have to search extensively for needed knowledge. Shared knowledge is delivered within the email – the current problem/activity being solved by the user.

A user can receive email with additional information (text attachments) at the end or beginning of the email message. This information can contain relevant categories, hints or links (similar as Gmail) to related cluster resources such as document repositories, databases or information systems. It can also suggest the most appropriate reply text for this email e.g. when replying to received order or problem resolution. Furthermore some suggestions for a user concerning possible next activities to be taken could be presented to the user.

Please note that text attachments are directly displayed in most of email clients and they appear as part of an email message, however they do not change the email message itself. Text attachments can appear as an addition which does not reduce user overhead because users have to read such attachment to have information, but this information should mainly provide links between relevant organizational resources. A similar idea, with email annotation and hints, was used in the Pellucid IST project with a different type of a hint presentation than intended in this paper. The real challenge is to create simple tools which a regular user can use for updating a knowledge model – hints, rules for matching hints, annotation patterns, similarity matching etc.

---

[2] www.commorg.net

The potential of the proposed system is to boost knowledge management in and among SMEs by providing a push oriented, pro-active solution reducing necessary efforts for knowledge management and which can be customized in the same form for different processes in one organization. The system can be innovative also by taking advantage of the central role of emails linked to business processes, by enabling e-collaboration and knowledge sharing in organizations, and by active and context-sensitive delivery of shared knowledge.

Good results can be achieved by deep semantic analyses of emails, including email data, standards, formats and attachments, as well as jointly analyzing email headers and email bodies according to reference business models and organization knowledge using semantics, and linking to organizational resources and systems. These analyses can be supported by statistical analysis of messages. The results can reflect a kind of experience, which extends knowledge step by step.

3 ACoMA System

Use of emails gives us possibility to provide knowledge in a context. Email serves as context provider. By attaching organizational or task knowledge to email, we give user possibility to access knowledge directly when needed. On Figure 2 we can see architecture of the system as well as example of an email with attached knowledge in form of notes with associated links to resources. Knowledge is provided in form of HTML attachment inside of email message. The ACoMA tool is developed in scope of Raport project[3]. More details on ACoMA can be found in [5]. ACoMA use also EMBET tool [6] to get relevant knowledge from organizational memory.

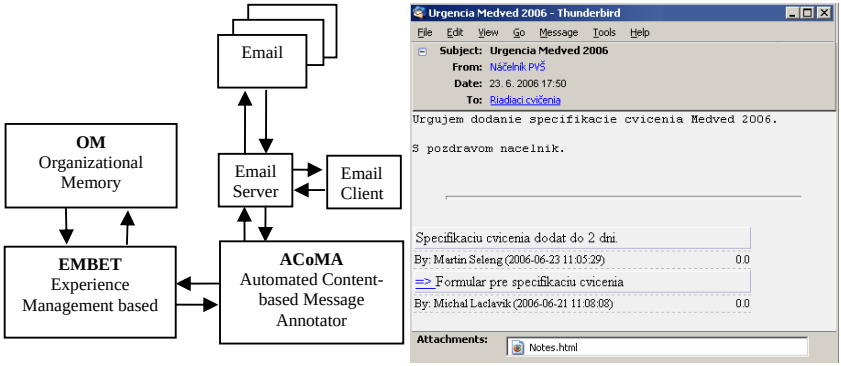


Fig. 2. ACoMA Architecture and Email with attached knowledge.

[3] http://raport.ui.sav.sk/

4 Conclusion and Future Work

Knowledge sharing platforms and tools are mainly “passive” and provide static knowledge sharing functionality. The possibility of “active” and “pro-active” intervention on communication channels, to provide real “interaction” with the experience formation and use process, is not widely considered.

The paper proposes also an active e-Collaboration approach to support, and even enable, SMEs to carry out their business overcoming geographical, technological and organizational barriers. The direct intervention on communication channels – based on e-mail message processing – is the distinguishing aspect, which addresses knowledge extraction and discovery as well as proactive provision of support to users. The generic e-Collaboration platform envisaged includes shared knowledge representation, business services provision and e-mail analysis and processing to extract knowledge and provide support in terms of annotation, information integration and suggestions.

Three main barriers hampering ICT adoption by SMEs can be addressed by such platform: economic, by delivering a lightweight platform able to interact with simple networked environments (e-mail); shortage of skill, by addressing self-adaptive and automatic organization knowledge base creation and maintenance; and cultural, by integrating with normal ways of working.

We also present briefly ACoMA and EMBET system which are being developed in this direction.

References

1. J. Habermas, The Theory of Communicative Action, Beacon, Boston, 1981.
2. David G. Schwartz and Dov Te’eni, Bar-Ilan University, Tying Knowledge to Action with kMail, MAY/JUNE 2000 , IEEE Knowledge Management, p 33-39
3. D. Te’eni and D.G. Schwartz, “Contextualization in Computer-Mediated Communication,” Information Systems—The Next Generation, L. Brooks and C. Kimble, eds., McGraw-Hill, New York, 1999, pp. 327–338
4. C.A. O’Reilly and L.R. Pondy, “Organisational Communication,” Organisational Behavior, S. Kerr, ed., Grid, Columbus, Ohio, 1979, pp. 119–150.
5. Martin Seleng, Michal Laclavik, Zoltan Balogh, Ladislav Hluchy: Automated Content-based Message Annotator – AcoMA; In: Proceedings of ITAT 2006 Information Technologies - Applications and Theory, Peter Vojtas (Ed.), Department of Computer Science, Faculty of Science, Pavla Jozef Safarik University, Kosice, 2006, pp.195-198, ISBN 80-969184-4-3; http://laclavik.net/publications/Itat_2006_seleng.pdf
6. Laclavík, M., Gatial, E., Balogh, Z., Habala, O., Nguyen, G., Hluchý, L.: Experience Management Based on Text Notes (EMBET); Proc. of eChallenges 2005 Conference, 19 - 21 October 2005, Ljubljana, Slovenia, Innovation and the Knowledge Economy, Volume 2, Part 1: Issues, Applications, Case Studies; Edited by Paul Cunnigham and Miriam Cunnigham; IOS Press, pp.261-268. ISSN 1574-1230, ISBN 1-58603-563-0.

Information Retrieval for Voice Operated Information System in Slovak

Marián Trnka

Institute of Informatics of the Slovak Academy of Sciences, Bratislava, Slovakia
trnka@savba.sk

Abstract. Speech communication interfaces (SCI) are nowadays widely used in several domains. Automated spoken language human-computer interaction can replace human-human interaction if needed. In this paper we describe development of the first Slovak spoken language dialogue system. In this paper we focus mainly on the information retrieval for the described system. The dialogue system is based on the DARPA Communicator architecture. The functionality of the SLDS is demonstrated and tested via two pilot applications, „Weather forecast for Slovakia“ and „Timetable of Slovak Railways“.

Key words: information system, dialogue system, Galaxy, speech recognition, speech synthesis, html parsing, web wrapper

Introduction

Due to the progress in the technology of speech recognition and understanding, the spoken language dialogue systems (SLDS) have emerged as a practical alternative for the conversational computer interface. They are more effective than the Interactive Voice Response (IVR) systems since they allow for more free and natural interaction. In this paper we describe the development of the first SLDS which is able to interact in the Slovak language. The system has been developed in the period from July 2003 to June 2006. The SLDS enables multi-user interaction in the Slovak language through telecommunication networks to find information distributed in computer data networks such as the Internet.

Contractors of the project are the Ministry of Education of the Slovak Republic and the Technical University of Košice. Collaborative organizations are the Institute of Informatics, the Slovak Academy of Sciences Bratislava, the Slovak University of Technology in Bratislava and the University of Žilina.

The paper is organized as follows. The first chapter gives a brief overview of the system architecture. Information server and it structure is described in the second chapter. Chapter three addresses two pilot applications.

System architecture

The architecture of the developed system is based on the DARPA Communicator [1, 2]. Our system consists of the hub and six system modules: telephony module (Audio server), automatic speech recognition (ASR) module, text-to-speech (TTS) module, transport module, backend module (Information server) and module of dialogue management. The communication among the dialogue manager, the Galaxy hub, and the other system modules is represented schematically in Figure 1.

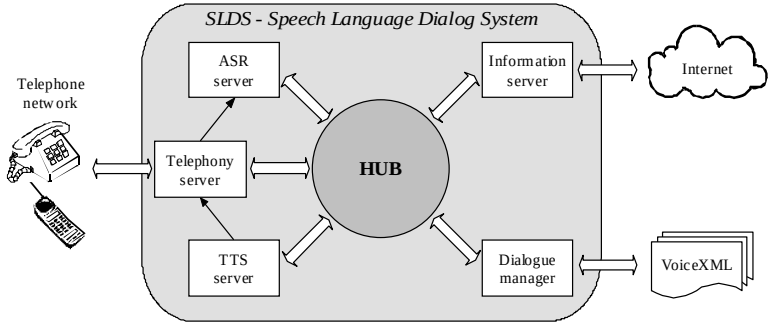


Fig. 1. The architecture of the Galaxy/VoiceXML based spoken Slovak dialogue system

The telephony module connects the whole system to a telecommunication network. It opens and closes telephone calls and transmits the speech data to/from the ASR/TTS modules. The automatic speech recognition server performs the conversion of incoming speech to a corresponding text. Context dependent HMM acoustic models trained on SpeechDat-Sk and MobilDat-Sk speech databases [3, 4] and ATK/HTK and Sphinx IV based speech recognition engines [5, 6] were used in this task. The dialogue manager [7] controls the dialogue of the system with the user and performs other specified tasks. The heart of the dialogue manager is the interpreter of VoiceXML mark-up language. The information server connects the system to information sources and retrieves information required by the user. The server of text-to-speech (TTS) synthesis [8, 9] converts outgoing information in text form to speech, which is more user-friendly.
We have designed the system to support „Windows-only" as well as mixed Windows/Linux platform solutions.

Information server

Having analyzed various existing approaches of information retrieval from the web, and the task to be carried out by the information server in our pilot applications, we came to the decision that a simple retrieval technique, such as a rule based ad-hoc application searching only several predefined web-servers with a relatively well

known structure of pages is sufficient to fulfill the task. Selection of eligible web servers was preceded by proper examination for stability and information reliability during one month period.
The information server (backend server) is capable of retrieving the information contained on the suitable web-pages according to the Dialogue Manager (DM) requests, extracting and analyzing the desired data. If the data are taken for valid, it returns them in the XML format to the DM. If the backend server fails to get valid data from one web source, it switches to the second wrapper retrieving the information from a different web-server.
The information server communicates with the HUB via the GALAXY interface. This module accomplishes its own communication with HUB, receives input requests, processes them and makes decisions to which web-wrapper (WW) the request should be sent. Then it receives an answer and sends it back to the HUB.

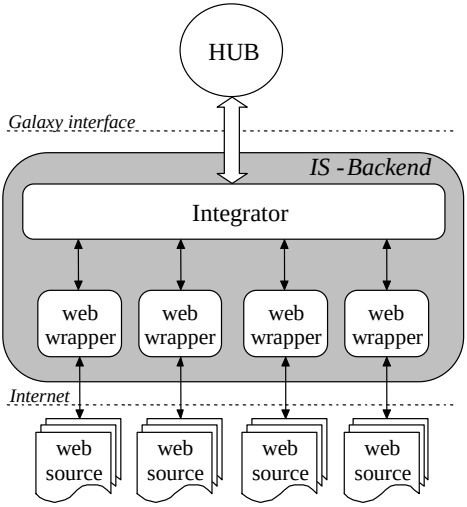


Fig. 2. Information server (Backend) architecture

Web wrapper architecture

The purpose of the web wrapper is to convert information implicitly stored as an HTML document into information explicitly stored as a data-structure for further processing. [10] The various components of our web wrapper and their interaction within the information flow are presented in Figure 1.
The web-wrapper is responsible for the navigation through the web-server, data extraction from the web-pages and their mapping onto a structured format (XML). The wrapper is specially designed for one source of data; thus to combine data from different sources, several wrappers are required.

The wrappers are designed to be as robust as possible against changes in the web-page structure. Nevertheless, in the case of substantial changes in the web-page design, the adaptation of the wrapper would probably be inevitable.
An automatic periodic download and cashing of the web-page content is performed to speed up the system (to eliminate the influence of long reaction times of the web-pages) and to assure robustness against drop-out while simultaneously keeping the information as current as possible.
The server allows for extension with additional web-wrappers thus it is open for future applications and services. The information on the wrappers currently accessible is stored in the system configuration file.

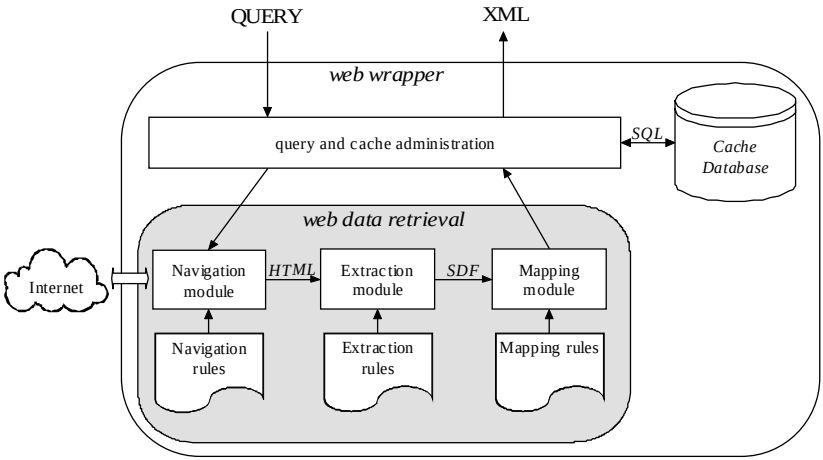


Fig. 3. Web wrapper architecture

Pilot applications

Weather forecast service

Based on the available web sites www.meteo.sk and www.shmu.sk the proposed telephone-based Weather Forecast Service enables to get weather forecast for about 80 Slovak district towns and the most popular tourist localities. Callers are able to access continuously updated weather information simply by uttering the names of cities or holiday localities and date.
It was necessary for every entity to define their keys or key words, i.e. main obligatory items, which must be obtained from user (data necessary to query the database and to obtain the information requested):

- Place: District town or holiday locality
- Date: relative date / accurate date

Timetable of Slovak Railways

The second service running on our SLDS provides information about Slovak railways timetable. Callers have direct access to the online timetable on webpage during this service.
In the first step of the dialogue design we specified necessary items of domain database. These items represent required information which needs to be obtained from the user:
• Starting place: railway station in Slovakia
• Destination place: railway station in Slovakia
• Date: relative date (today, tomorrow etc.)/absolute date ("the twentieth of December" etc.)
• Time: departure time (hour, minute)

Implementation of services

Both services are available at one access point, which is accessible from several telephone numbers. One can try them by calling +421 55 602 2297 (T-com), +421 911 650 038 (T-Mobile), +421 918 717 491 (Orange).
At the beginning, the system provides an initial message, where it welcomes the user, introduces itself and notices the user about the key words/expressions and its use. Afterwards, user is prompted to choose one of the available services - Weather forecast or Railway timetable. Then the dialog switches to the particular sub-dialog of the selected service.
A typical dialogue between the user (U) and the system (S) looks as follows:
S: Welcome to the IRKR portal. Would you like to play the introduction?
U: No.
S: Choose one of the services: Weather forecast or Railway's timetable.
U: Weather forecast
S: Please, name a city and assign a day, for which you want to get the weather forecast.
U: Bratislava, tomorrow.
S: Did you say Bratislava, tomorrow?
U: Yes
S: The weather forecast for Bratislava for tomorrow is: sunny, 32 degrees centigrade...
Notice: IRKR means „Inteligentné rečové komunikačné rozhranie", which is a Slovak expression for Intelligent Speech Communication Interface.

Conclusions

In this paper we have described the development of the first Slovak spoken language dialogue system. Our main goal was to develop a dialogue system that will serve as a starting platform for further research in the area of spoken Slovak engineering. The work on this project was finished in the year 2006 [11]. We successfully combined up

to date free resources with our own research into functional system that enables a multi-user interaction through telephone in Slovak language to retrieve required information from Internet resources. The functionality of the SLDS is demonstrated and tested by means of two pilot applications, „Weather forecast for Slovakia“ and „Timetable of Slovak Railways“. Applying new findings we continue in further development and improvement of the system.

Acknowledgements

This work was supported by Slovak Grant Agency VEGA under grants No. 1/1057/04, 1/3110/06, 2/2087/22 and by the Ministry of Education of Slovak Republic under research project No. 2003 SP 20 028 01 03.

References

1. http://www.sls.csail.mit.edu/sls/technologies/galaxy.shtml
2. http://communicator.sourceforge.net/
3. Pollak, P., Cernocky, J., Boudy, J., Choukri, K., Rusko, M., Trnka, M. et al. “SpeechDat(E) „Eastern European Telephone Speech Databases“, in Proc. LREC 2000 Satellite workshop XLDB - Very large Telephone Speech Databases, Athens, Greece, May 2000, pp. 20-25.
4. Rusko M., Trnka M., Darjaa S.: MobilDat-SK – a Mobile Telephone Extension to the SpeechDat-E SK Telephone Speech Database in Slovak, Proceedings of the XI. International Conference SPECOM 2006, St. Petersburg, Russia, 2006, pp. 449 - 454, (ISBN 5-7452-0074-x)
5. Lamere, P., Kwok, P., Walker, W., Gouvea, E., Singh, R., Raj, B., Wolf, P., “Design of the CMU Sphinx-4 decoder”, in Proc. Eurospeech 2003, Geneve, Switzerland, September 2003, pp. 1181–1184
6. Mirilovič, M., Lihan, S., Juhár, J., Čižmár,A., “Slovak speech recognition based on Sphinx-4 and SpeechDat-SK”, in Proc. DSP-MCOM 2005, Košice, Slovakia, 2005, pp. 76-79.
7. Ondáš, S., Juhár, J., “Dialogue manager based on the VoiceXML interpreter”, in Proc. DSP-MCOM 2005, Košice, Slovakia, Sept. 2005, pp.80-83.
8. Rusko M., Trnka M., Darjaa S.: Three Generations of Speech Synthesis systems in Slovakia, Proceedings of the XI. International Conference SPECOM 2006, St. Petersburg, Russia, 2006, pp. 449 - 454, (ISBN 5-7452-0074-x)
9. Čepko, J., Rozinaj, G., "Corpus synthesis of Slovak speech", Proc. of the 46th International Symposium Electronics in Marine – ELMAR 2004, pp.313-318, Zadar, Croatia. 2004.
10. Chun-Nan Hsu, Chia-Hui Chang, Harianto Siek, Jiann-Jyh Lu, Jen-Jie Chiou: Reconfigurable Web Wrapper Agents for Web Information Integration, IJCAI-03 Workshop on Information Integration on the Web, Mexico, 2003
11. Juhár J., Ondáš S., Čizmár A., Rusko M., Rozinaj G., Jarina R., „Development of Slovak GALAXY/VoiceXML Based Spoken Language Dialogue System to Retrieve Information from the Internet“, Proceedings of the Ninth International Conference on Spoken Language Processing (Interspeech 2006 — ICSLP), Pittsburgh, Pennsylvania, USA, 2006, ISSN 1990-9772, pp. 485-488.

Riadiaci agentový systém na báze umelých imunitných systémov•

Tomáš Kasanický

Ústav informatiky, Slovenská akdemia vied, Dúbravska cesta 9, Bratislava
tomas.kasanicky@tuke.sk

Abstrakt. Funkcionalita mnohých autonómnych aplikácií, pracujúcich v reálnom prostredí, si vyžaduje vytvorenie špeciálnych postupov, ktoré ju zabezpečujú v prípade zlyhania, či už programového, alebo hardverového vybavenia. Tento príspevok poukazuje na možnosť využiť pri riešení tohto problému generickú schopnosť algoritmov umelých imunitných systémov, aplikovaných ako agentový systém v procese odstraňovania neznámych a neočakávaných zlyhaní sledovaného systému. Práca sa pokúša zachovať princíp redundantnosti, ktorý je zakladným prístupom pri tvorbe zlyhaniu odolávajúceho systému (fault-tolerant systems)

Kľúčové slová: Fault-tolerance, graceful degradation, artificial immune systems, multi-agent systems

Úvod

Častou požiadavkou na súčastné aplikácie používané v reálnych podmienkach je zabezpečenie funkcionality systému (hardvéru a softvéru) aj počas zlyhania. Za zlyhanie sa pritom považuje akákoľvek skutočnosť, ktorá znemožní fungovanie systému. Táto práca sa zameriava na popis systému, ktorý je určený prevažne pre ochranu riadiacich algoritmov proti zlyhaniu hardverových súčasti sledovaného systému, pričom porucha môže byť spôsobená zmenou prostredia, v ktorom zariadenie operuje, alebo poruchou na samotnom zariadení.

Tvorba zlyhaniu odolávajúcich (fault-tolerance) výpočtových zariadení sa datuje do počiatku samotného vzniku týchto zariadení. Medzi prvé aplikácie postupov zabraňujúcich znefunkčneniu výpočtového systému možno označiť počítač SAPO (SAmočiný POčítač), ktorý vytvoril Antonin Svoboda (1907-1980). Konštrukčne bol počítač založený na rele a bubnovej pamäti. Procesor využíval metódu triplifikácie, kde sa daná operácia vykonala paralelne na troch jednotkách a za správny výsledok sa prehlásil ten, ktorý sa vyskytol na najmenej dvoch jednotkách. Nasledoval významný

• Popisovaná metodológia bola vyvinutá za podpory VEGA 2/7101/27 a APVV LPP-0231-06

rozvoj tejto oblasti ku, ktorému značne prispeli okrem iných AT&T (system ESS) a NASA (Self-Testing-And Repair).

V súčastnej dobe sa čoraz častejšie objavujú práce, ktoré sa pomocou evolučných princípov snažia vytvoriť zlyhaniu odolne zariadenia [3, 6, 1], ale tak isto práce, ktoré sa zaoberajú možnosťou vytvoriť systém, ktorý dokáže generovať redundantnosti, ktoré následne umožnia fungovanie celku aj počas vplývania chyby. Táto praca zavadza do tejto problematiky, podobne ako [7, 5, 4], generickú shopnosť algoritmov umelej imunitnej odozvy .

Umele imunitné systémy

Umelé imunitné systémy sa pokúšajú riešiť niektoré paradigmy výpočtových systémov na základe paralely medzi imunitnými systémami živočíchov a výpočtovými systémami. Existuje viacero definícií umelých imunitných systémov, avšak jednou z najvšeobecnejších, a teda aj najpoužívanejšou je definícia Johna Timmisa:

Umelý imunitný systém je výpočtový systém založený na metaforách prirodzených imunitných systémov.

Jedná sa teda o prístupy, ktoré sa motivujú imunitným systémom (prevažne cicavcov) a snažia sa hľadať similarit medzi teóriou imunitných systémov a riešenými problémami, prípadne simuláciou biologických procesov za účelom štúdia.

Základné modely a algoritmy umelej imunitnej odozvy:

- Model kostnej drene
- Model týmusu
- Algoritmus klonálnej selekcie
- Model imunitnej siete
- Alternatívne prístupy

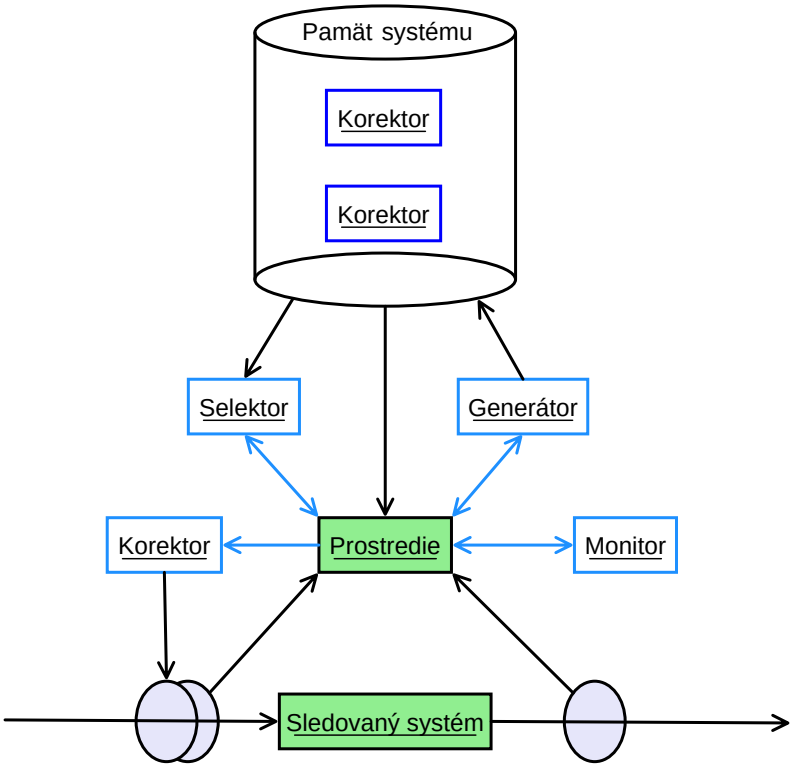
Táto práca si za funkčný model zvolila predlohu algoritmu klonálnej slekcie. Navrhovaný agentový systém simuluje funkcionalitu aktivácie B-imunitných buniek cicavcov. Algoritmus, ktorý uskutočňuje túto simuláciu (simulácia kolonálnej selekcie) sa nazýva Clonalg [2]. Jedná sa o jeden zo základných algoritmov umelej imunitnej odozvy. Jeho základná štruktúra sa dá zhrnúť do nasledujúcich bodov:

1. Náhodná inicializácia populácie indivíduí M.
2. Zistenie afinity medzi prvkom P a každým jedincom populácie M .
3. Výber n1 jedincov z populácie M, ktorých afinita je najvyššia, generovanie kópii z týchto jedincov, pričom počet generovaných kópií je závislý na miere afinity jednotlivca z populácie a prvku P.
4. Mutácia všetkých kópií s pravdepodobnosťou mutácie závislej od miery afinity prvku P a jedincom z populácie.
5. Pridanie týchto klonov do populácie M a následný výber n2 indivíduí z tejto populácie a uchovanie ich ako pamäť systému.

Opakovaniu bodov 2 až 5 pokiaľ nie je splnené požadované kritérium.

Agentový systém

Agentový systém je založený na reaktívnych agentoch. Komunikácia medzi jednotlivými agentamy je realizovaná nepriamo cez prostredie. Rovnakým spôsobom prebieha hlavná komunikácia, aj medzi bunkami imunitného systému. V dôsledku unifikácie prostredia bol zavedený agent, ktorý zabezpečuje normalizáciu prostredia na požadované hodnoty. Tento agent tak isto prijíma jednotlivé komunikačné správy a cyklicky ich rozposiela všetkým agentom, teda simuluje prostredie, v ktorom sa všetky agenty nachádzajú.



Obr. 1. Navrhovaný agentový systém.

Pokiaľ agent *monitor* spozoruje zmenu chovania sledovaného systému, vyšle do prostredia správu o tejto zmene. Na základe tejto zmeny *selektor* hľadá v pamäti systému odpovedajúci korekčný člen v podobe agenta *korektora*. Ak bol nájdený, vykoná sa náhrada. V prípade, že v pamäti systému sa nenachádza potrebný *korektor*, *generátor* sa pomocou princípov klonálnej selekcie pokúsi vygenerovať vhodný *korektor*. Ak vyhovuje požiadavkám, je zaradený do riadenia a uložený do pamäte systému.

Zhodnotenie

Popisovaný systém sa pokúša nájsť riešenie v situáciách, kedy by iné konvenčné metódy riadenia neboli schopne bez zásahu človeka pokračovať v činnosti. Avšak je nutná podotknúť, že popísaný prístup nezaručuje nájedenie riešenia za každej situácie, a tak isto nájdené riešenie nemusí byť nutne optimálne. Preto v ďalšom vývoji tohto satému je nutné sa zamerať na interaktívnu evolúciu, ktorá umožní v prípadoch, kde je to dostupné, zasiahnuť obsluhe a usmerniť výber pri hľadaní riešenia.

Referencie

1. Avizienis A., Bondavalli A., Thevenod-Fosse P. An immune system paradigm for the design of fault tolerant systems, Lecture notes in computer science (Lect. notes comput. sci.) ISSN 0302-9743
2. De Castro, L.N., Timmis J. Artificial Immune Systems: A New Computational Intelligence Approach", Springer-Verlag ISBN- 978-1852335946, 2002
3. Hartmann M.,Haddow P. C.. Evolution of fault-tolerant and noise-robust digital designs. Computers and digital techniques, IEE proceedings. Stevenage, 2004. ISSN 1350-2387.
4. Jun, J.-H., Lee, D.-W. & Sim, K.-B. Realization of Cooperative and Swarm Behavior in Distributed Autonomous Robotic Systems Using Artificial Immune System, In Proc. IEEE SMC pp. 614-619 1999
5. Krishna Kumar, K. & Neidhoefer, J.. An Immune System Framework for Integrating Computational Intelligence Paradigms with Applications to Adaptive Control. In Computational Intelligence A Dynamic System Perspective, IEEE Press, pp. 32-45 1999.
6. P. Jedrzejowicz, I. Czarnowski, H. Szreder and A. Skakowski. FaultTolerant Programs on Multiple Processors. Parallel and Distributed Processing. Lecture Notes in Computer Science. Springer, 1999.
7. Richard O. Canham and Andy M. Tyrrell. A Hardware Artificial Immune System and Embryonic Array for Fault Tolerant Systems. Genetic Programming and Evolvable Machines. Springer Netherlands, 2004. ISSN 1389-2576.

Semantic web

Scripting the Semantic Web

Marian Babik, Ladislav Hluchy *

Intelligent and Knowledge-based Technologies Group,
Department of Parallel and Distributed Computing, Institute of Informatics,
Slovak Academy of Sciences
Marian.Babik@saske.sk, Ladislav.Hluchy@savba.sk

Abstract. The Semantic Web is a vision for the future of the Web in which information is given explicit meaning, making it easier for machines to automatically process and integrate information available on the Web. Semantic Web will build on the well known Semantic Web language stack, part of which is the Web Ontology Language (OWL). Python is an interpreted, object-oriented, extensible programming language, which provides an excellent combination of clarity and versatility. The deep integration of both languages is one of the novel approaches for enabling free and interoperable data [1].
In this article we present a metaclass-based implementation of the deep integration ideas. The implementation is an early Python prototype supporting in-line class and properties declaration, instance creation and simple triple-based queries. The implementation is backed up by a well known OWL-DL reasoner Pellet [3]. The integration of the Python and OWL-DL through meta-class programming provides a unique approach, which can be implemented with any metaclass enabled scripting language.

1 Introduction

The deep integration of scripting languages and Semantic Web has introduced an idea of importing the ontologies directly into the programming context so that its classes are usable alongside classes defined normally. This can provide a more natural mapping of OWL-DL than classic APIs, reflecting the set-theoretic semantics of OWL-DL, while preserving the access to the classic Python objects. Such integration also encourages separation of concerns among declarative and procedural and encourages a new wave of programming, where problems can be defined by using description logics [8] and manipulated by dynamic scripting languages [1]. The approach represents a unification, that allows both languages to be conveniently used for different subproblems in the software-engineering environment.

* Acknowledgments: The research reported in this paper has been partially financed by the EU within the project IST-2004-511385 K-WfGrid and Slovak national projects, NAZOU SPVV 1025/2004, RAPORT APVT-51-024604, VEGA 2/6103/6, VEGA 2/7098/27.

In this article we would like to introduce an early prototype, which implements some of the ideas of the deep integration in Python language [6]. It supports in-line declaration of OWL classes and properties, instance creation and simple triple-based queries [2]. We will emphasize the notion of modeling intensional sets (i-sets) through metaclasses. We will also discuss the possible drawbacks of the approach and the current implementation.

2 Intensional Sets and Metaclasses

Intensional sets as introduced in [1] are sets that are described with OWL DL's construct and according to this description, encompass all fitting instances. A sample intensional set can be defined by using Notation3 (N3) [10] as, e.g. ":Person a owl:Class; rdfs:subClassOf :Mortal". This simply states that Person is also a Mortal. Assuming we introduce two instances, e.g. ":John a :Person and :Jane a :Mortal", the instances of Mortal are both John and Jane. Please note, that N3 is used only for demonstration purposes, the current implementation can also support NTriples and RDF/XML.

Terminology-wise, a metaclass is simply "the class of a class". Any class whose instances are themselves classes, is a metaclass. A metaclass-based implementation of the intensional sets is based on the core metaclass Thing, whose constructor accepts two main attributes, i.e. default namespace and N3 description of the i-set. The instance of the metaclass is then a mapping of the OWL class to the intensional set. Following the above example class Person can be created with a Python construct:

```
Person = Thing('Person', (),
{defined_by: 'a owl:Class; rdfs:subClassOf :Mortal',\
 namespace: 'http://samplens.org/test#'})
```

This creates a Python class representing the intensional set for Person and its namespace. In the background it also updates the knowledge base with the new assertion. The individual John can then be instantiated simply by calling $John = Person()$. This statement calls the default constructor of the class Person, which provides support for asserting new OWL individual into the knowledge base. A similar metaclass is used for the OWL property except that it can not be instantiated. The constructor is used here for different purpose, i.e. to create a relation between classes or individuals. The notion of importing the ontology into the Python's namespace is then a matter of decomposing the ontology into the groups of intensional sets, generating Python classes for these sets and creating the instances.

This kind of programming is also called metaclass programming and can provide a transparent way how to generate new OWL classes and properties. Since metaclasses act like regular classes it is possible to extend their functionality by inheriting from the base metaclass. It is also simple to hide the complex tasks needed for accessing the knowledge base, reasoner and processing the mappings between OWL-DL's concepts and their respective Python counterparts.

3 Implementation and Drawbacks

One of the main drawbacks of the current implementation is the fact, that it doesn't support open world semantics (OWA)[1]. Although the reasoner in the current implementation can perform OWA reasoning and thus it is possible to correctly answer queries, the Python's semantics are based on the boolean values. One of the possibilities is to use epistemic operator as suggested in [1], however this is yet to be implemented. Another problem when dealing with ontologies are namespaces. In the current prototype we have added a set of namespaces that constitute the corresponding OWL class or property description as an attribute of the Python's class. This attribute can then be used to generate the headers for the N3 description. This approach needs further extension to support management of different ontologies. One of the possibilities would be to re-use Python's module namespace by introducing a core ontology class. This ontology class would serve as a default namespace handler as well as a common importing point for the ontology classes.

The other drawback of the approach is the performance of the reasoner, which is due to the nature of the JPype implementation (the conversions between virtual machines imposes rather large performance bottlenecks). This can be solved by extending the support for other Python to Java APIs and possible implementation of specialized client-server protocols.

4 Related Work

To our best knowledge there is currently no python implementation of the deep integration ideas. However there are similar projects written in other languages such as ActiveRDF and PHPHomepage [18, 19]

Since Java is a frame language its notion of polymorphism is very different than in RDF/OWL. This is usually solved by incorporating design patterns, which make the APIs quite complex and sometimes difficult to use. The dynamic nature of the scripting languages can support OWL/RDF level of polymorphism and thus it is possible to directly expose the OWL structures as Python classes without any API interfaces. One of the interesting Java projects, which tries to automatically map OWL ontologies into Java through Java Beans is based on the ideas shown in [15]. This approach tries to find a way how to directly map the ontologies to the hierarchy of the Java classes and interfaces.

There is quite a number of python libraries supporting RDF and OWL such as cwm, pychinko and rdflib [11, 12, 20]. Among the existing Python libraries, which support RDF and OWL, the most interesting in terms of partial integration are Sparta and Tramp, which bind RDF graph nodes to Python objects and RDF arcs to attributes of such objects [17, 4]. The projects however doesn't clearly address the OWL and are mainly considered with RDF. It is thus difficult to evaluate what is the level of support for the inferred OWL models.

[1] A prototype implementation is available under MIT license at: http://seth-scripting.sourceforge.net

5 Conclusion

We have described a metaclass-based prototype implementation of the deep integration ideas. We have discussed the advantages and shortcomings of the current implementation. We would like to note, that this a work in progress, which is constantly changing and this is just a report of the current status. At the time of writing authors are setting up an open source project, which will host the implementation of the ideas presented. There are many other open questions, that we haven't covered here including integration of query languages (possibility to re-use ideas from native queries [16]); serialization of the ontologies; representation of rules, concrete domains, etc. We hope that having an initial implementation is a good start and that its continuation will contribute to the success of the deep integration of the scripting and Semantic Web.

References

1. Vrandecic, D., Deep Integration of Scripting Languages and Semantic Web Technologies, In Soren Auer, Chris Bizer, Libby Miller, 1st International Workshop on Scripting for the Semantic Web SFSW 2005 , volume 135 of CEUR Workshop Proceedings. CEUR-WS.org, Herakleion, Greece, May 2005. ISSN: 1613-0073
2. Web Ontology Language (OWL), see http://www.w3.org/TR/owl-features/
3. Pellet OWL Reasoner, see http://www.mindswap.org/2003/pellet/index.shtml
4. TRAMP: Makes RDF look like Python data structures http://www.aaronsw.com/2002/tramp, http://www.amk.ca/conceit/rdf-interface.html
5. Bechhofer, S., Lord, P., Volz,R.:Cooking the Semantic Web with the OWL API. 2nd International Semantic Web Conference, ISWC, Sanibel Island, Florida, October 2003
6. G. van Rossum, Computer programming for everybody. Technical report, Corporation for National Research Initiatives, 1999
7. Java-Python Extension, http://sourceforge.net/projects/jpe
8. Baader, F., Calvanese, D., McGuinness, D.,L., Nardi, D. and Patel-Schneider,P., F. editors. The description logic handbook: theory, implementation, and applications. Cambridge University Press, New York, NY, USA, 2003.
9. Jython, Java implementation of the Python, http://www.jython.org/
10. Notation 3, see http://www.w3.org/DesignIssues/Notation3.html
11. Closed World Machine, see http://www.w3.org/2000/10/swap/doc/cwm.html
12. Katz, Y., Clark, K. and Parsia, B., Pychinko: A native python rule engine. In International Python Conference 05, 2005.
13. Euler proof mechanism, see http://www.agfa.com/w3c/euler/
14. JPype, Java to Python integration, see http://jpype.sourceforge.net/
15. Kalyanpur, A., Pastor, D., Battle, S. and Padget, J., Automatic mapping of owl ontologies into java. In Proceedings of Software Engg. - Knowledge Engg. (SEKE) 2004, Banff, Canada, June 2004.
16. Cook, W. R. and Rosenberger, C., Native Queries for Persistent Objects, Dr. Dobb's Journal, February 2006
17. Sparta, Python API for RDF, see http://www.mnot.net/sw/sparta/
18. ActiveRDF putting the semantic web on rails, http://www.activerdf.org/
19. RDFHomepage project, http://rdfhomepage.opendfki.de/
20. RDFLib, http://rdflib.net/

Towards Semantic Enterprise Vision*

Michal Laclavík, Ladislav Hluchý, Marián Babík, Zoltán Balogh,
Ivana Budinská, Martin Šeleng, Marek Ciglan

Institute of Informatics, Slovak Academy of Sciences, Dubravska cesta 9,
Bratislava, 845 07, Slovakia
laclavik.ui@savba.sk
http://ikt.ui.sav.sk/

Abstract. By "Semantic Organization" or "Semantic Enterprise" we understand applying semantic web ideas & technologies in organizations. Semantic web research brought new ideas and promising results. However results and applications are still far from original objectives. We believe that one of the reasons is problem of upper level ontologies and ontology translation and matching. While in semantic web it is extremely important to solve such problem, in organizational level we can use domain or application specific semantic description with better success. We believe that organizational applications such as knowledge management, semantic based information processing, semantic based workflow and services can be successfully applied in organizations. In this article we discuss such Semantic Enterprise Vision and provide some examples of related research in the Institute of Informatics SAS in this area.

1 Introduction

Semantic Enterprise or Organization as explained in our paper is not mentioned in many resources, however many researcher are aware of possible success of semantic web research and technologies in organizational context. Semantic organization can be understood as availability of information and knowledge as understood in Knowledge Management [1]. Term "Semantic Enterprise" or "Semantic Organization" is mentioned only in several sources in this context [2] or as a way of enterprise integration [3] or also as a way for enterprise information portal [4].

Organizations need to manage information and knowledge and make it available for its employees in pro active way. We discuss this problem in the second chapter. Organization needs also to offer its product and services, manage resources needed or providing of services and product and manage workflows in organization as well as intra organization workflows, which is related to intelligent semantic information search or knowledge supported workflows and services as in semantic service oriented architectures which we discuss in 3rd chapter.

* This work is supported by projects K-Wf Grid EU RTD IST FP6-511385, NAZOU SPVV 1025/2004, RAPORT APVT-51-024604, VEGA 2/6103/6, VEGA 2/7098/27.

2 Processing of Information for Knowledge Management

An area related to Knowledge Management or intelligent information search can be understood as to get right information and knowledge in right context. So far most of the systems allow searching information using full text search. The common problem is information relevance and overload. Other search solutions exist, which uses some semantic data and categories, however user has to actively search for the information. In our work we focus on active knowledge provision where we have created EMBET architecture [5, 16]. In EMBET we try to detect context from computerized tasks and match information and knowledge from organizational memory for such context. We identify 4 main problems which we believe need to be solved in any active knowledge provision system:

- Information context detection
- User context detection
- Information versus user context matching
- Displaying the knowledge

It is clear that we solved these only very partially, but important is also to identify the challenges. For information context detection we use semi-automatic ontology based semantic annotation where we developed OnTeA tool [6], which semi automatically annotates text or documents. For database data annotation we have also developed RDB2Onto tool [26]. Information and knowledge can be also annotated manually. Other application specific tools such as KAA-WXA [7] or ACoMA [8] can be used to provide information and knowledge with assigned context as well. Common approach is also to use document classification or clustering. To identify and extract important information in document we also use ExPoS tool [9] to process HTML or text documents. For information processing and retrieval it is important to use also solutions such as indexing and full text search, where we developed RFTS tool [27] offering special functionalities supporting semantic annotation as well.

For user context detection, we use interface which listens for events relevant to user environment. Here we developed event based model [10]. Such events can be sent for example from workflow management systems as we have used in Pellucid project [11], where workflow management system informed Pellucid about a user activity or a process instance. Workflow systems notify the EMBET system similar way in Raport [12] or K-Wf Grid projects [13]. In Raport project, we additionally detect user context from user's electronic communication [8].

Even when we correctly detect user and information knowledge, there is problem with context matching. So far we have used and created several simple context matching algorithms such as: matching based on context subset or superset; similarity based context matching based on weights or concrete implementation [14]; ontology concepts similarity [15]; or some combinations of previous. Some new ideas we also introduced in [16], which are based on intersection of user and knowledge context.

When needed information and knowledge is identified we have to display it to the user. In most of the cases we have used web based interfaces based on XSL [5, 14], but we have identified other possibilities such as graph or tree based visualization [17] as well. Personalization and knowledge navigation [18] plays important role and should not be overcome. Important is not only visual aspect but also the way of using

knowledge based user interface. We found quite interesting displaying information and knowledge in email messages [8].

2.1 Organizational Memories

In order to manage and store semantic information & knowledge we need to create organizational memories. Several solutions are available such as Jena[1], Sesame[2] or KAON[3], however each application requires different approach [19, 20]. We also focus on problem of distributed knowledge bases [21].

3 Semantic based Workflows and Services

Workflow processes are adopted in most of organizations. In some organizations, processes are supported by Workflow Management Systems. Computerized workflow processes can provide important part of user context in which relevant information and knowledge can be provided. Workflow processes can support basic administration processes [12, 22] but can be also applied on workflow of services or web services [23]. Information & knowledge can be provided to user or can help execute workflows automatically by supporting such workflow and services with semantic information [24, 25, 7]. Many organizations can provide their product and services via web services. Semantic based web services and Service Oriented Architectures is becoming popular research field due to organization and internet evolution and interest.

4 Conclusions and Future Work

In this paper we briefly touch vision of Semantic Enterprise or Semantic Organization and explain how our research is related to this vision. We identify some of challenges in such research and try to describe our position and our related research. We describe briefly tools and systems[4] designed and developed at Institute of Informatics SAS and gave a list of references to related publications. Many of mentioned tools are presented in other articles of these proceedings as well.

References

1. Thomas H. Davenport, Laurence Prusak, Working Knowledge, ISBN:1578513014, 2000

[1] http://jena.sourceforge.net/
[2] http://www.openrdf.org/
[3] http://kaon.semanticweb.org/
[4] http://www.ikt.ui.sav.sk/?page=software.php

2. Michael Wacey: The Semantic Organization: Knowing What You Know; 2005, http://xml.sys-con.com/read/136230.htm
3. Leo Obrst: Toward a Standard Rule Language for Semantic Enterprise Integration, http://www.mitre.org/news/events/tech05/briefings/2148.pdf, MITRE's Technology Symposium, 2005
4. Emanuele Della Valle, Maurizio Brioschi: Towards a Semantic Enterprise Information Portal, Sundial Resort, Sanibel Island, Florida, USA - ISWC 2003
5. Laclavik M., Gatial E., Balogh Z., Habala O., Nguyen G., Hluchy L.: Experience Management Based on Text Notes (EMBET); Proc. of eChallenges 2005 Conference, 19 - 21 .2005, Ljubljana, Slovenia, Innovation and the Knowledge Economy, Volume 2, Part 1: Issues, Applications, Case Studies; Edited by P.Cunnigham and M. Cunnigham; IOS Press, pp.261-268. ISSN 1574-1230, ISBN 1-58603-563-0.
6. Michal Laclavik, Martin Seleng, Emil Gatial, Zoltan Balogh, Ladislav Hluchy: Ontology based Text Annotation – OnTeA; Information Modelling and Knowledge Bases XVIII. IOS Press, Amsterdam, Marie Duzi, Hannu Jaakkola, Hannu Kangassalo, Yasushi Kiyoki (Eds.), Vol. 154, February 2007, ISBN 978-1-58603-710-9
7. Martin Šeleng, Michal Laclavík, Zoltán Balogh, Ladislav Hluchý: Semantic Analysis of Grid Workflows WXA (Workflow Xml Analyzer); In: Proc. of the 2nd Int. Workshop on Grid Computing for Complex Problems - GCCP'2006, Bratislava, 27-29 November 2006;
8. M. Seleng, M. Laclavik, Ladislav Hluchy, Jacek Kitowski: ACoMA: Towards E-mail based Collaborative Working Environments in SMEs; In: P. Mikulecky, J. Dvorsky, M. Kratky (Eds.): Znalosti 2007, Proceedings of 6th annual conference. VSB-Techická universita Ostrava, pp.356-359. February 2006, Ostrava, Czech Republic. ISBN 978-80-248-1279-3.
9. Oravec V., Nguyen G.: Offer Extraction and Separation for Internet Documents. In: Tools for Acquisition, Organisation and Presenting of Information and Knowledge. P.Navrat et al. (Eds.), Vydavatelstvo STU, Bratislava, 2006, pp.22-30, ISBN 80-227-2468-8. Workshop 26-28 September, Nizke Tatry, Slovakia.
10. Laclavik, M., Babik, M., Balogh, Z., Hluchy, L.:AgentOWL: Semantic Knowledge Model and Agent Architecture; In Computing and Informatics. Vol. 25, no. 5 (2006), p. 419-437. ISSN 1335-9150
11. Lambert S., et all: Knowledge management for organisationally mobile public employees. In: Proc. of KMGov 2003, Rhodes Island, Greece, LNCS 2645, Springer-Verlag, pp. 203-212, ISSN 0302-9743, ISBN 3-540-40145-8.
12. Budinska I.: Ontology based knowledge system for administrative workflows management; WIKT 2006
13. Habala O., Babík M., Hluchý L., Laclavík M., Balogh Z.: Semantic Tools for Workflow Construction In: Proc.of International Conference on Computational Science, Part III, LNCS 3993, Springer-Verlag, 2006, pp. 980-987, ISSN 0302-9743, ISBN 3-540-34383-0, May 28-31, Reading, UK
14. Laclavik M.: Ontology and Agent based Approach for Knowledge Management; PhD Thesis submitted for the degree philosophiae doctor; Institute of Informatics, Slovak Academy of Sciences, Applied Informatics, Submitted June 2005; Defended 12th January 2006
15. Balogh Z., Budinská I.: OntoSim - Ontology-based Similarity Determination of Concepts and Instances. In: Tools for Acquisition, Organisation and Presenting of Information and Knowledge. P.Navrat et al. (Eds.), Vydavatelstvo STU, Bratislava, 2006, pp.64-70, ISBN 80-227-2468-8. Workshop 26-28 September, Nizke Tatry, Slovakia.
16. Michal Laclavik, Martin Seleng, Ladislav Hluchy: EMBET: Towards User Assistance, Collaboration and Knowledge Sharing; In: P. Mikulecky, J. Dvorsky, M. Kratky (Eds.): Znalosti 2007, Proceedings of 6th annual conference. VSB-Techická universita Ostrava, Fakulta elektrotechniky a informatiky, 2007, pp.356-359. February 2006, Ostrava, Czech Republic. ISBN 978-80-248-1279-3.

17. Laclavík M., Balogh Z., Nguyen G.T., Gatial E., Hluchý L.: Methods for Presenting Ontological Knowledge to the Users; In: L.Popelinsky, M.Kratky (Eds.): Znalosti 2005, Proceedings, VSB-Techicka universita Ostrava, Fakulta elektrotechniky a informatiky, 2005, pp.61-64. ISBN 80-248-0755-6. February 2005, Vysoke Tatry
18. Navrat, P.—Bielikova, M.—Rozijanova, V.: Methods and Tools for Acquiring and Presenting Information and Knowledge in the Web. In: Proc. International Conference on Computer Systems and Technologies – CompSysTech 2005, Varna 2005, http://ecet.ecs.ru.acad.bg/cst05/Docs/cp/SIII/IIIB.7.pdf.
19. Kryza B., Pieczykolan J., Babík M., Majewska M., Slota R., Hluchý L., Kitowski J.: Managing Semantic Metadata in K-Wf Grid with Grid Organizational Memory. In: Bubak, M., Turala, M., Wiatr, K. (editors): Proceedings of the Cracow Grid Workshop '05, Cracow, November 2005, pp. 66-73, published by ACC CYFRONET AGH, Poland, April 2006. ISBN 83-915141-5-3.
20. Ciglan M., Babík M., Laclavík M., Budinská I., Hluchý L.: Corporate Memory: A framework for supporting tools for acquisition, organization and maintenance of information and knowledge. In: Proc. of 9-th Intl. Conf. ISIM'06 "Information Systems Implementation and Modelling", Brno, April, MARQ Ostrava, 2006, pp. 185-192, ISBN 80-86840-19-0.
21. Babík M., Hluchý L.: A Scalable Distributed Ontology Repository In: L.Popelínský, M.Krátký (Eds.): Znalosti 2005, Proceedings, VŠB-Techická universita Ostrava, Fakulta elektrotechniky a informatiky, 2005, pp.8-17. ISBN 80-248-0755-6. February 2005, Vysoke Tatry, Slovakia.
22. Laclavík M., Balogh Z., Hluchý L., Krawczyk K., Dziewerz M., Kitowski J., Majewska M.: Knowledge Management for Administration Processes In: Proceedings of Znalosti 2004, February 2004, VŠB-Techická universita Ostrava, pp.248-255. ISBN 80-248-0456-5.
23. Habala O., Babík M., Hluchý L., Laclavík M., Balogh Z.: Semantic Tools for Workflow Construction In: Proc.of International Conference on Computational Science, Part III, LNCS 3993, Springer-Verlag, 2006, pp. 980-987, ISSN 0302-9743, ISBN 3-540-34383-0, May 28-31, Reading, UK
24. Babík M., Hluchý L., Kitowski J., Kryza B.: WSRF2OWL-S: A Framework for Generating Semantic Descriptions of Web and Grid Services. In: Bubak, M., Turala, M., Wiatr, K. (editors): Proceedings of the Cracow Grid Workshop '05, Cracow, November 2005, pp. 49-56, published by ACC CYFRONET AGH, Poland, April 2006. ISBN 83-915141-5-3.
25. Balogh Z., Gatial E., Laclavik M., Maliska M., Hluchy L.: Knowledge-based Runtime Prediction of Stateful Web Services for Optimal Workflow Construction. Proc. of 6-th Intl. Conf. on Parallel Processing and Applied Mathematics PPAM'2005, R.Wyrzykowski et.al. eds., 2006, LNCS 3911, Springer-Verlag, pp. 599-607, ISSN 0302-9743, ISBN 3-540-34141-2. Poznan, Poland.
26. Michal Laclavik: RDB2Onto: Relational Database Data to Ontology Individuals Mapping; In: Tools for Acquisition, Organisation and Presenting of Information and Knowledge. P.Navrat et al. (Eds.), Vydavatelstvo STU, Bratislava, 2006, pp.86-89, ISBN 80-227-2468-8. Workshop 29-30 September, Nizke Tatry, Slovakia.
27. Ciglan M.: Documents Content Indexing for Supporting Knowledge Acquisition Tools. In: Tools for Acquisition, Organisation and Presenting of Information and Knowledge. P.Navrat et al. (Eds.), Vydavatelstvo STU, Bratislava, 2006, pp.101-104, ISBN 80-227-2468-8. Workshop 26-28 September, Nizke Tatry, Slovakia.

Semantic-based Groupware System for SAKE

Peter Butka, Ján Hreňo

Technical University of Košice, Letná 9, 04200, Košice, Slovakia
Peter.Butka@tuke.sk, Jan.Hreno@tuke.sk

Abstract. In this article we propose the Semantic-based Groupware System (GWS) for SAKE project. SAKE (Semantic Agile Knowledge-based E-government) is a STREP Project sponsored by the European Union starting March in 2006. The overall objective of SAKE is to specify, develop and deploy a holistic framework and supporting tools for an agile knowledge-based e-government that will be sufficiently flexible to adapt to changing and diverse environments and needs. We give a brief overview of the SAKE semantic-based GWS which will be provided by us in the project.

Keywords: semantic-based groupware system, e-government, agile knowledge management

1 Introduction

Existing approaches for knowledge management in e-government focus mainly on the efficient management of a particular, isolated knowledge resource and on supporting only message-based communication between public administrators. However, the demands for knowledge-based e-government are much higher:

- First, the existing approaches do not take into account the increased granularity of informational resources and the manifold semantic
- Second, due to complexity of the decision making processes, effective knowledge management requires the creation of a supportive, collaborative culture while eliminating traditional rivalries.
- Third, the usage of existing knowledge resources is indeed a valid aspiration, but for realizing a learning e-government, the crucial is creation of *new* knowledge.
- Finally, ad hoc management of the changes in e-government systems might work in the short term, but to avoid unnecessary complexity and failures in the long run, management must be done in a systematic way.

Whole SAKE approach will provide tools and methodologies to address these problems. In this article we will concentrate on semantic-based groupware system.

2 Semantic-based Groupware System

Semantic-based Groupware system supports more efficient knowledge sharing by developing:
- methods and tools for ontology-based tagging the interaction between public administrators;
- methods and tools for enabling building community of practice from interaction log and their specific vocabularies by social tagging;
- methods and tools for collaborative knowledge creation methods and tools for pushing of knowledge and for searching for experts.

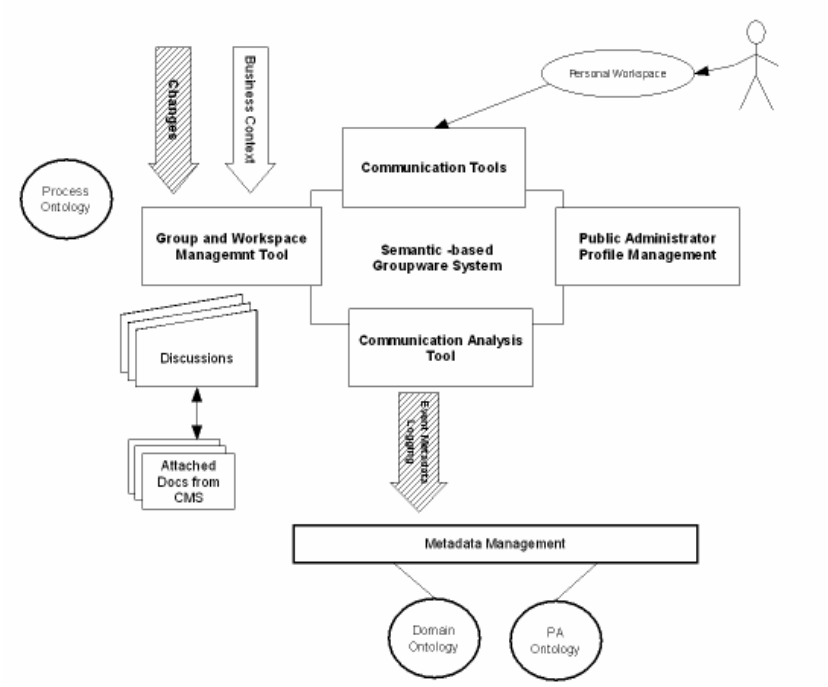


Fig. 1. Semantic-based Groupware System in SAKE

Figure 1 shows basic components of semantic GWS, it consists of four modules, with the following responsibilities:

Group and workspace management

- Creates new groups according to given processes/activities
- Changes groups during their lifetime according to the changes in the PA profiles
- Extract knowledge from failure of communication (metadata logging)

Communication tools

- Set of tools enabling users to communicate in needed form

Communication analysis tool

- Extract knowledge from successful communication (metadata logging)

Public administrator profile management

- PA profiles in form of concepts from PA ontology
- Helps to find proper PA for a business context/activity

3 Conclusion

The importance of supporting better management of continually changing knowledge is nowadays more important due to the evolution of Europe towards a multicultural, more open and international society with changing common values, increasing levels of education, demographic involvement and adoption of new technologies (EU report, 2004). This is especially true for the New Member States and Candidate Countries, since European integration has paved the way for new legislation, regulations and corresponding changes that affect the way Public Administrations in the Enlarge Europe are organized and operate.

SAKE (Semantic Agile Knowledge-based E-government) is a STREP Project sponsored by the European Union starting March in 2006. The overall objective of SAKE is to specify, develop and deploy a holistic framework and supporting tools for an agile knowledge-based e-government that will be sufficiently flexible to adapt to changing and diverse environments and needs. Indeed, it will provide an innovative framework, supported by an integrated platform, for realizing agile knowledge-based e-government system, based on semantic technologies. In this article we proposed the Semantic-based Groupware System (GWS) for SAKE platform, which will be provided by us in the project.

Acknowledgment

The research presented in this paper was partially funded by the EC in the project “IST PROJECT 027128 - SAKE”.

References

1. Nonaka, I., Takeuchi, H.: The Knowledge-Creating Company: How Japanese Companies Create The Dynamics of Innovation. Oxford: Oxford University Press (1995)
2. Stojanovic, N., Mentzas, G., Apostolou, D.: Semantic-enabled Agile Knowledge-based e-government. Semantic Web meets eGovernment 2006 AAAI Spring Symposium Series Stanford University, California, USA, March 27-29 (2006)
3. SAKE: *Annex 1: Description of Work*. 18 October 2005.

Workflow based orchestration model for WSMO

Peter Bednár[1], Ján Hreňo[2]

[1] Centre for Information Technologies, [2] Faculty of Economics, Letna 9, 040 01 Kosice, Slovakia
{Peter.Bednar, Jan.Hreno}@tuke.sk

Abstract. For the services providing applications in e-Government the orchestration model where both electronic services and "traditional" services provided by the public administration offices can be combined to the hybrid complex scenario. The proposed scenario process model will specify additional high level structure for processes and will extend the WSMO orchestration to guide citizens to achieve specific goals and to coordinate activities performed by citizen, traditional public administration services and web services.

Keywords: Semantic Web Services, WSMO, orchestration.

1 Introduction

The current proposal for WSMO orchestration model is intended for automatic orchestration of Web services and cannot be used directly in the context of e-Government. Today, situation in the e-Government requires the orchestration model where both electronic services and "traditional" services provided by the public administration offices can be combined to the hybrid complex scenario. The following sections will describe current proposal for the WSMO orchestration and choreography based on the Abstract State Machines and extended model based on the workflow process modelling which will provide support for generic hybrid scenarios.

2 WSMO orchestration and choreography

A WSMO choreography description consists of the states represented by ontology, and the if-then rules that specify (guarded) transitions between states. The ontology that represents the states provides the vocabulary of the transition rules and contains the set of instances that change their values from one state to the other. The concepts of an ontology used for representing a state may have specified the grounding mechanism which binds service description to the concrete message specification (e.g. WSDL).Like for the choreography; an orchestration description consists of the states and guarded transitions. In extension to the choreography, in an orchestration can also appear transition rules that have as postcondition the invocation of a mediator that links the orchestration with the choreography of a required web service.

3 Workflow based WSMO orchestration

The proposed scenario process model will specify additional high level structure for processes and will extend the WSMO orchestration to achieve the following requirements:

a) guide citizens to achieve specific goals and to
b) coordinate activities performed by all actors - citizen, traditional public administration services and web services.

Proposed scenario process model distinguishes between atomic, abstract, and composite activities. Atomic activities can be invoked, have no sub-activities, and are executed in a single step from the requester's point of view. The abstract activities are specified as sub-goals used as elements of abstraction, they are viewed as executed in a single step, but they are not invocable. Abstract activities are resolved to the atomic activities in the composition phase or during the execution phase. Composite activities consist of the simple activities and define their workflows using control constructs, such a sequence, flow, if-then-else or iterate. Specification of the composite activity includes its data flow and variable bindings which will specify how its inputs are accepted by particular sub-activities, and how its various outputs are produced by particular sub-activities.

Important aspect in our design will be the support for activities assigned to the human actors (i.e. citizens, officers etc.). Human activities are typically associated with the "traditional" services but can be used to model citizen activities which are related to a specific life event but are out of the scope of the public administration services.

Besides supporting the described activity types, the following facilities were identified as useful for a process model to provide support for modelling orchestrated scenarios:

- the proposed model should support common workflow and dataflow patterns [2]
- compatibility with the proposed standard workflow modelling languages (i.e. WS-BPEL), which will allow to reuse existing workflow models for modelling of the semantic services
- compatibility with the standard process modelling notation (i.e. BPMN or UML) in order to visualize scenarios to the users and to use standard tools for modelling,

Acknowledgments. The work presented in the paper is supported by the EC within the FP6-2004-27020 Project “Access to e-Government Services Employing Semantic Technologies”

References

1. Roman et al.: Web Service Modeling Ontology (WSMO), WSMO deliverable D2 version 1.1. available from http://www.wsmo.org/TR/d2/v1.2/, [2005]
2. W.M.P. van der Aalst, A.H.M. ter Hofstede, B. Kiepuszewski, and A.P. Barros.: Workflow Patterns. Distributed and Parallel Databases, 14(1):5-51, [2003].

Applications

Znalostmi řízený průchod kurzem

Zdeněk Velart[1], Petr Šaloun[2], and Markéta Kalinská[1]

[1] Katedra informatiky, FEI, VŠB-TU Ostrava
{zdenek.velart.fei, marketa.kalinska.fei}@vsb.cz
[2] Katedra informatiky a počítačů, PřF Ostravská univerzita v Ostravě
petr.saloun@osu.cz

Abstrakt Tradiční sekvenční model uspořádání kurzu nemusí vyhovovat každému. Jednou z možností, jak toto omezení obejít je využití adaptivních hypermédií a personalizovat navigaci a předkládaný materiál. Rozšířením této idee o slovník pojmů odpovídající znalostem definovaným v konceptech je možné definovat navigaci v kurzu pomocí znalostí a prerekvizit jednotlivých konceptů. Důležitou součástí systému, který by takovýto formát kurzu realizoval je logování a to z důvodu adaptace ale také z důvodu úpravy slovníku pojmů, znalostí a prerekvizit na základě vyhodnoceného průchodů studentů.

Tradiční model uspořádání kurzu je založen na sekvenční prezentaci informace studentovi, takovým způsobem, jak autor kurzu zamýšlel. Toto uspořádání však nemusí vyhovovat každému, především studentům, kteří daný kurz již neúspěšně absolvovali a opakují jej nebo také studentům se znalostmi překrývajícími se z jiného kurzu, např. v oblasti programování základní řídící struktury jsou velmi podobné pro jazyky Java a C++.

Jedním ze způsobu jak obejít toto omezení je využití metod a technik definovaných v oblasti nazývané adaptivní hypermédia a personalizovat navigaci a předkládaný materiál konkrétním studentům například podle dosažené úrovně a rozsahu znalostí.

Tato idea personalizace pomocí adaptivních hypermédií může být dále rozšířena vytvořením slovníku pojmů odpovídající znalostem definovaným v konceptech. Koncept (stránka, část stránky) pak bude definovat nebo využívat konkrétní pojmy a jejich zpracování v rámci celého kurzu umožní vytvořit uspořádání konceptů podle definovaných a používaných pojmů (znalosti a prerekvizity). Takto uspořádané koncepty pak budou nabídnuty studentům jako alternativní průchody kurzem.

V souvislosti s předchozími úvahami se začínáme zabývat možností nevycházet pouze ze sekvenčního uspořádání kurzu a založit pruchod kurzem pomocí snad vhodnějšího a lépe adaptovatelného přístupu. Odpověď na tuto otázku není jednoznačná a tento text se bude snažit alespon částečně na ni odpovědět a i na semináři budeme v diskuzi hledat odpověď.

S potřebou definovat kurz, ne pomocí sekvenčního uspořádání jednotlivých konceptů, ale pomocí vazeb mezi prerekvizitami a novou informací popisující příslušný koncept vyvstává otázka jakým způsobem je možné tyto pojmy pro

jednotlivé koncepty manuálně či automatizovaně definovat. Sekvenční, autorem vytvořené pořadí konceptů, bude základem pro automatizovanou tvorbu slovníku pojmů. Očekáváme, že srovnáním množin pojmů mezi koncepty s využitím původního sekvenčního pořadí konceptů vznikne neúplné uspořádání konceptů z pohledu slovníku pojmů. Toto uspořádání ukáže možné vstupní/startovní body pro úrovně znalostí dosažené v jiných kurzech nebo pro jiné pořadí průchodu koncepty než bylo autorem původně zamýšleno. Například v kurzu C++ vcelku tradiční čelní umístění aritmetických výrazů a základních matematických operací může být nahrazeno nezávisle rozvíjenou větví začínající například manipulaci s řetězci či znaky. Tento nový pohled na obsah kurzu bude zajímavý z pohledu tvůrce mapy pojmů i pro autora obsahu kurzu. Doufáme zejména že výsledek bude zajímavý pro cílovou skupinu uživatelů, tedy pro studenty. V oblasti, kterou se zabýváme – výuka programování – je toto možné pomocí následujícího způsobu.

Existující koncepty a také ukázkové příklady zdrojového textu asociované s konkrétními koncepty se automatizovaně zpracují pomocí nástroje, který bude vyhledávat v textu výskyt klíčových slov jazyka a dalších pojmů, které navíc autor kurzu manuálně dodefinoval jako klíčová slova kurzu. Získanou množinu pojmů z jednoho konceptu je nutné upravit a to rozdělením na dvě skupiny, kde jedna reprezentuje prerekvizity a druhá představuje znalosti naučené z tohoto konceptu, které zároveň mohou představovat prerekvizity jiného konceptu. Tímto způsobem se vytvoří, za přispění autora kurzu, graf kurzu založený na prerekvizitách a znalostech.

Navíc by autor ke každému takto rozšířenému konceptu měl přiřadit hodnotu, určující úroveň znalostí získaných naučením se příslušného učebního materiálu. Definováním tohoto atributu autor kurzu může určit, jaká úroveň naučených znalostí má odpovídat výsledné známce z kurzu. Dává tím také možnost studentům na začátku kurzu si určit jakou výslednou známku z kurzu by rádi získali a tím i jaké množství informace se chtějí naučit, čímž by se i na základě tohoto nastavení přizpůsobovalo zobrazované menu jednotlivým studentům. Volba výsledné známky však nesmí fungovat jako omezení, ale spíše jako doporučení, který materiál je nutné znát na danou úroveň.

Velmi důležitou součástí systému, ve kterém by byl takovýto kurz realizován je logování. A to jednak z důvodu aktuálního přizpůsobování se studentovi a jeho potřebám, ale také z důvodu zpětného vyhodnocení všech nasbíraných informací o studentech a jejich průchodem kurzem. Tyto logovaná data pak mohou být po patřičném vyhodnocení využity k opravě dosavadního uspořádání konceptů, pokud bude zjištěno, že některý z konceptů je větší skupinou studentů naprosto opomíjen, a tudíž je jeho výskyt v průchodu nadbytečný, nebo že po jeho navštívení studenti přecházeli na stránku s konceptem, jež nebyl pro daný průchod doporučen a tudíž by měl být do tohoto průchodu kurzem zařazen.

Přínosem tohoto řešení je personalizace (skupinová či individuální) pro studenty přicházející z příbuzných kurzů (tzn. kurzů, jejichž průnik znalostí je neprázdný, například kurz programování C++ a kurz programování v Javě), kde si studenti sami definují znalosti, které již ovládají nebo pomocí mapování mezi

metadaty jednotlivých kurzů budou tyto znalosti určeny. Následně již studentům nemusí být předkládán celý kurz, ale jen takové koncepty, které jim přinesou nějaké nové informace nebo znalosti.

Reference

1. Brusilovsky P., Sosnovsky S., Yudelson M., Chavan G. *Interactive Authoring Support for Adaptive Educational Systems.*
2. Brusilovsky P., Sosnovsky S., Yudelson M. Accessing *Interactive Examples with Adaptive Navigation Support.* In Fifth IEEE International Conference on Advanced Learning Technologies (ICALT'04) . 2004. s. 842-843.
3. Šaloun P., Velart Z. *Adaptive Hypermedia as a mean for learning programming.* In Workshop proceedings of the sixth international conference on Web engineering 2006. ACM Digital Library, 2006, ISBN 1-59593-435-9
4. Velart Z., Šaloun P. *Výuka programování C++ v adaptivním systému AHA!.* In Technologie pro e-vzdělávání 2006, Praha:ČVUT FEL, 2006
5. Velart Z., Šaloun, P. *Využití adaptivního hypermediálního systému AHA! při výuce programovacího jazyka C++.* In Objekty 2005, 2005, 280-288, ISBN 80-248-0595-2

Model of Military Training – Knowledge Approach*

Igor Mokriš, Radoslav Forgáč

Institute of Informatics, Slovak Academy of Sciences, Bratislava, Slovakia
mokris@aoslm.sk, forgac@aoslm.sk

Abstract. The paper deals with the development of the ontology-based knowledge management system for organization of the military training. Model goes out from a detail analysis of the workflow processes associated with the military exercise preparation.

Keywords: military training, military exercise preparation model, workflow process, ontology-based knowledge system

1 Introduction

The ontology based knowledge management system [1] is developed as a pilot application for organization of military exercise preparation at Centrum of Simulation Technologies, National Academy of Defense, Liptovský Mikuláš, Slovakia (CST NAO). CST NAO organizes and performs the training and education of officers for headquarters' staffs and commanders. On the basis of assigned tasks, CST staff realizes the needed activities and account for the exercise organization which nowadays is realized manually using office software.

2 Model of military exercise preparation description

Process of military exercise preparation can be divided into three parts (Fig.1). There is an exercise preparation, exercise execution and exercise evaluation [3].

The most important for the system development is first part. During this part the work meetings of employees of CST and officers of Slovak army are performed. The aim of the work meeting is to determine the content, topics, tasks and targets of military exercise. There are prepared a lot of documents as the scenarios of military exercise, simulation and technical plans, choice of localities for military actions by digital maps of terrain, etc.

* This work was supported by Slovak Science and Techn. Assist. Agency under the contract No. APVT-51-024604 and Slovak Science Agency VEGA No. 2/7098/27

Second part is exercise execution. The officers of headquarters staffs and military groups, which take part in military training process, take place in CST. Theirs role is to solve the determined targets of military exercise, learn to use the needed documents, react on arise military situations, etc. Exercise performs by computer simulation software, which simulates military actions of officers, military groups and military techniques during simulated computer fight under virtual reality.

Third part is exercise evaluation after realization of military fight training activities. There are evaluated the activities of each commander and group during simulated fight and at the same time is evaluated exercise preparation by CST, too. The documents and information about exercise preparation is archived.

The employees of CST participate especially on first part (mainly exercise planning) and marginally on second part (exercise execution) and on third part (exercise evaluation) of military exercise process. Officers of Slovak army are responsible for military exercise preparation, execution and exercise evaluation from point of military fight activities.

During of the military exercise planning process it is needed to perform large amount of work for preparation of exercise requirements, collection of documents (so called Exercise Directives) and needed tasks for every employee of CST under consideration actual situation in personally staff of CST. This documents and control flow will be used for representation, modeling and evaluation of CST activities and at the same time it can be used also for verification of the preparation of CST for planned activities (Fig.2).

However, CST organizes several overlapping military exercises during a training year which complicate the organization of each exercise preparation. Thus, overlapping of information about the organization of individual exercises can occur, and this results in further difficulties. There are in general problems with information management, knowledge management and time management, which can be solved.

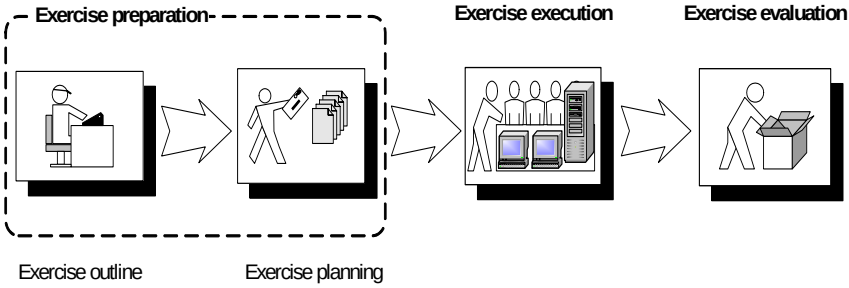


Fig. 1 Organization of a military training in CST NAO LM

Above mentioned military exercise preparation was analyzed and its model was proposed. In this model the data flow, process flow, organization, communication and personnel structure was considered.

The organization changes in modeled military exercise preparation may cause time delays and increase the risk of losing information, which is needed for qualified management. The knowledge system offers solution for such situations and enables simplify the exercise organization for employees of CST [2,4]. In the case of changes of conditions from reason of absence of employees, or arrival of new employees, etc., the knowledge system can minimize the time consuming and the risk of information loss. Utilization of knowledge system prevents mistakes in realization of exercise preparation from point of planning of needed activities, documents preparation, etc. The contribution of the knowledge system in CST task organization is to define the field of activity and relations between employees of CST so as after whatever change were possible, without more serious time loss, verify all needed activities and continue in given activity without detriment of complexity, competence and qualification.

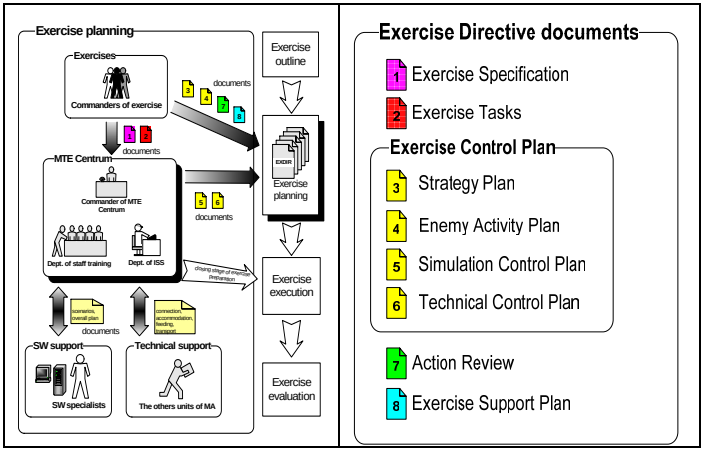


Fig. 2 Organization of a military exercise preparation in CST NAO LM

The structure of the knowledge system is designed in the generic way that can work for arbitrary administration process. It also takes care about administrative processes in CST. The system architecture comes from the following requirements as collect experience from users and present useful experience to other users that works in the same or similar work context; keep an eye on current training plans that contain of important deadlines, alarm users about that; prepare for users at deadlines necessary information such as predefined emails, documents, formulas and let users know about that and support user's experience exchange and collaboration.

Based upon these requirements the personal, communication, function, data and process model of workflow management system was developed (Fig.3) [1, 3, 5]. Important background of the system design is its ontology, which defines structure and relationships among experience entities. Ontology is the main mechanism used

for the representation of information and knowledge, definition of the meaning of the terms used in the content language and the relation among these terms.

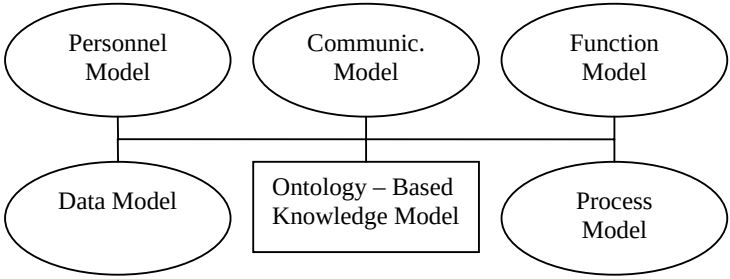


Fig. 3 Model of military exercise preparation in CST NAO LM

.3 Conslusion

The approach of the knowledge management for the design of the system architecture in the administrative process in army training centers is build based on Slovak national project APVT-51-024604 [5].

References

1. Hluchý L., Budinská I., Balogh Z., Forgáč R., Gatial E., Laclavík M., Mokriš I., Nguyen G.: Modeling of Military Training Activities by Ontology-based Knowledge System. Proc. of 6. Int. Conf. “Multimedia in Bussiness”, ISBN 83-7251-673-1, Kielce, Oct. 25- 27, 2006, pp. 192-198.
2. Laclavík, M., Gatial, E., Balogh, Z., Habala, O., Nguyen, G., Hluchý, L.: Experience Management based on Text Notes (EMBET). In. Cunnigham “Innovation and the Knowledge Economy: Issues, Applications, Case Studies”. Amsterdam, IOS Press, 2005. ISBN 1-58603-563-0, pp. 261-268.
3. Mokriš, I., Forgáč, R.: Utilization of System Pellucid for Training Organization of Slovak Officers in Military Academy at Liptovský Mikuláš. Proc. of Conf. „Simulation and Modeling in Slovak Army“, Liptovský Mikuláš, ISBN 80-8040-235-3, 2004, pp. 25-31, (in Slovak).
4. Návrat, P., Bieliková, M.: Tools for Knowledge Acquisition, Organization and Maintenance in the Environment of Heterogeneous Information Sources. Proc. of Znalosti ´6, Hradec Králové, February 2006, pp. 237-242, (in Slovak).
5. Research and Development of a Knowledge Based System to Support Workflow Management in Organizations with Administrative Processes (APVT-51-024604) http://raport.ui.sav.sk.

In memory object server based application for railway companies

Miloš Budinský

SIEMENS PSE
milos.budinsky@siemens.com

Abstract:n this paper it is briefly presented a Java based technique used in the real time in memory system of an Intranet/Internet real time Client/Server application for Railway companies. It belongs to the product family of Siemens PSE Austria called ROMAN (ROute MANagement), see (1, 2)

1 Application Domain

iRoman is intended for Railway companies mainly for short term train planning and/or train dispatching including train conflict detection and resolving in real time. It is a hot candidate to become a common basis of 3 new members of ROMAN applications family:
ROMAN Solver (Dispatching & Conflict Resolution)
ROMAN Cross Border (Cooperative Planning)
ROMAN Anywhere (Web Timetable Editor)

"Anywhere" is already sold to Italian and Austrian railways. "Solver" is currently being prepared for Tunisian railways.

2 Architecture

The technique is based on an in memory object server with a graph of objects prelinked during the server startup after all data is loaded into the operation memory. (See Fig 2, on the right)

This high performance in memory server uses a Track-Topology-Graph as a special distributed container keeping other business objects (like trains, runtime tables, design areas, etc.). It uses a fact that the most of business objects in this application domain are either graphs/sub-graphs (e.g. entire track topology, train design areas, blocking areas…) or oriented paths in the graph (trains, train timetable patterns…). Both graph like objects and graph-path like objects consist of graph-node like objects and graph-edge like objects. All of them are pre-linked together representing so a skeleton on which all other non-geographic data objects are "hanged" (e.g. time information, train conflict information…). A simplified example is shown in Fig 3.

Fast traversing in this pre-linked object graph leads to a high velocity of queries (like e.g. train conflicts search and resolving). Since all data are present all the time in the memory as Java objects, relatively easy and straightforward implementation of even complicate business rules is possible.

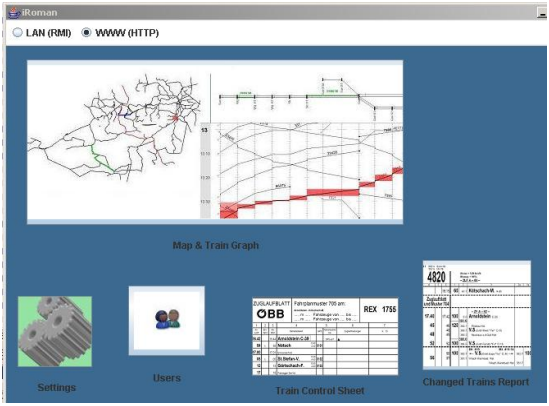


Fig.1 The thin client is a modular Java Webstart desktop application

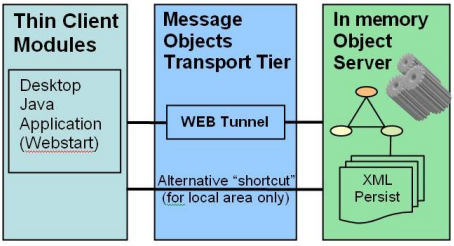


Fig.2 The thin client, middle tier, graphical output

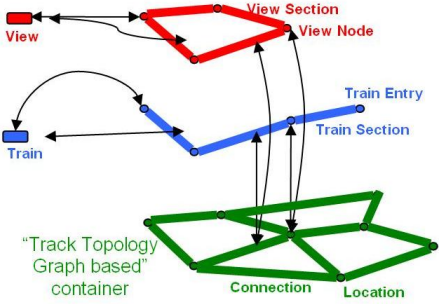


Fig.3 A simplified example

The thin client is a modular Java Webstart desktop application providing with a lot of graphical outputs. (See Fig 2, on the right and Fig 1).

A possible configuration of Client Application Startup Panel is shown in Fig 1. The icons shown in the picture can give an idea about the variety of client modules. The most important client modules are Train Graph Editor and Track Map based Editor.

The Train Graph screens displaying trains and conflicts of all clients are synchronized in real time via the automatic client refresh capability.

The client modules are loosely coupled with the above described services tier via sending message objects (locally or over the internet) thanks to the middle tier shown in the center of Fig 2.

4 Conclusion

The above described unconventional approach which takes into the account also the specific semantic of the data together with the in memory configuration leads to a very high performance (many times faster in comparison with a traditional relational database server). It represents a good platform for train conflict resolving and timetable optimization algorithms.

References

1. http://w2.siemens.at/roman
2. http://www.transportation.siemens.com/ts/en/pub/products/ra/products/control_tec/roman.htm
3. Architectural design specification, Product iRoman, Siemens PSE, 2007

Znalostný manažment v Ozbrojených silách Slovenskej republiky

Petr Všetečka

Akadémia ozbrojených síl gen. M. R. Štefánika, Demänová 393, 031 01 Liptovský Mikuláš, Slovak republic
http://www.aoslm.sk/, vsetecka@aoslm.sk

Abstrakt. Príspevok popisuje súčasný stav využitia znalostného manažmentu v podmienkach Ozbrojených síl Slovenskej republiky, rozsah vzdelávania profesionálnych vojakov v tejto oblasti a možnosti implementácie znalostného manažmentu do vojenských organizácií.

1 Pozadie problematiky

Znalostný manažment patrí medzi fenomény súčasnej doby. Veľa podnikov, firiem, inštitúcií a organizácií hľadá možnosti uplatnenia znalostného manažmentu s cieľom dosahovania vyššej prosperity a efektívnosti vynaložených prostriedkov. Jeden z koreňov vzniku základných myšlienok znalostného manažmentu je možné nájsť aj v americkej armáde a v snahe jej veliteľov uchovať drahocenné a krvou zaplatené skúsenosti z bojových akcií v zahraničí. Preto znalostný manažment má svoje miesto aj vo vojenských organizáciách vrátane Ozbrojených síl Slovenskej republiky (OS SR).

2 Výučba predmetu znalostný manažment

V podmienkach Slovenskej republiky v súčasnosti existujú 2 vzdelávacie inštitúcie, ktoré sa podieľajú na vzdelávaní, výchove a výcviku profesionálnych vojakov pre potreby SR. Akadémia ozbrojených síl gen. M. R. Štefánika v Liptovskom Mikuláši, ako vojenská vysoká škola univerzitného typu, zabezpečuje vzdelávanie kadetov v 4-ročnom bakalárskom štúdiu. Národná akadémia obrany maršala A. Hádika zabezpečuje vojenský program týchto kadetov a odborné kurzy v rámci kariérneho rastu profesionálnych vojakov.

V súčasnosti iba jeden študijný odbor bakalárskeho štúdia má vo svojom študijnom programe zahrnutý predmet „Znalostný manažment“. Tento celkom nový predmet je voliteľný a vyučovaný v anglickom jazyku v rozsahu 30 hodín. Jeho prípravu sprevádzali chýbajúce skúsenosti pedagógov a obmedzená dostupnosť informačných zdrojov. Na odstránení týchto skutočností sa intenzívne pracuje a obsahová náplň predmetu, ako aj informačné zabezpečenie sa postupne

zdokonaľujú. Aj keď skúsenosti s výučbou predmetu Znalostný manažment sú malé, sme si vedomí jeho dôležitosti najmä pre vojenskú organizáciu.

3 Práca so znalosťami v Ozbrojených silách Slovenskej republiky

V ozbrojených silách SR sa vždy pracovalo so znalosťami, no tejto práci chýba systematickosť a stanovené pravidlá. Prenos znalostí bol v minulosti zúžený iba na osobné predávanie skúseností medzi staršou a mladšou generáciou profesionálnych vojakov a s tým spojené časovo náročné vyhľadávanie možných nositeľov potrebných znalostí.

Vstupom do NATO sa podstatne rozšírilo spektrum použitia našich vojenských jednotiek. V súčasnosti má SR svojich profesionálnych vojakov vo viacerých misiách (vrátane Kosova, Afganistanu, Iraku, Cypru a i.). Pre prípravu príslušníkov do zahraničných vojenských misií bola zriadená špeciálna výcviková jednotka. Jej inštruktori sú vojaci, ktorí slúžili vo viacerých zahraničných misiách a sú nositeľmi potrebných znalostí. Ich úlohou je okrem prípravy a výcviku aj priama spolupráca a pomoc pri striedaní vojenských kontingentov v zahraničí.

Stále je to však človek, ktorý je nositeľom znalostí. V súčasnosti chýbajú pravidlá na získavanie znalostí, nie je vybudovaný elektronický znalostný systém pre zachytávanie, ukladanie a sprístupnenie potrebných znalostí. V niektorých prípadoch chýba motivácia a dôvera, ktorá by viedla jedinca k zdieľaniu svojich tacitných znalostí.

4 Perspektívy rozvoja

Vedecko pedagogický zbor Akadémie ozbrojených síl si je vedomý nutnosti rozvíjania znalostného manažmentu v kontexte s celoživotným, dištančným a elektronickým vzdelávaním a budovania inštitucionálnej pamäti. Za týmto účelom bola v roku 2006 zahájená príprava projektového zámeru Vojenskej virtuálnej univerzity. Výsledky tohto projektu majú byť určené najmä pre profesionálnych vojakov a zamestnancov verejnej a štátnej správy v oblasti obrany a bezpečnosti.

Rozhodujúcim partnerom pri príprave a realizácii projektu je Ústav informatiky Slovenskej akadémie vied. Hlavný prínos tohto pracoviska bude najmä pri tvorbe znalostného systému pre podporu riadenia výučbového procesu vo Vojenskej virtuálne univerzite. Hlavným cieľom znalostného systému bude podrobná analýza znalostí v elektronickom vzdelávacom procese, návrh metód na zachytávanie, uchovávanie a vytváranie znalostí súvisiacich s elektronickým vzdelávaním a vytvorenie rámcového modelu znalostí, ktorý bude reprezentovať znalosti pedagóga, ako aj študenta.

Realizácia projektu umožní kvalitatívne novú úroveň distribúcie informácií, učiva a znalostí podstatne širšiemu okruhu záujemcov, bez ohľadu na dennú dobu a vzdialenosť. Tým sa Vojenská virtuálna univerzita stane ďalším krokom ku sprístupneniu dištančného vzdelávania odbornej verejnosti v podmienkach Slovenskej republiky.

Záver

Implementácia znalostného manažmentu do Ozbrojených síl Slovenskej republiky bude dlhodobý a veľmi náročný proces. V súčasnosti iba niektorí rozhodujúci funkcionári OS SR si uvedomujú potrebu využitia znalostí v prospech celej organizácie. Potrebu zdieľania znalostí si však uvedomujú najmä príslušníci jednotiek vysielaných do zahraničných misií a vedecko pedagogický pracovníci Akadémie ozbrojených síl.

Problematika učiacej sa organizácie a inštitucionálnej pamäti má veľké perspektívy uplatnenia vo vojenských štruktúrach. Je iba otázkou času, kedy tejto tematike bude prisúdená taká vážnosť, ktorú si zaslúži. Projekt Vojenskej virtuálnej univerzity by mohol byť prvým významným krokom na ceste implementácie znalostného manažmentu do vojenských organizácií.

Literatúra

1. Všetečka, P. – Svetlík, M.: Vedomostný manažment v procese informačných operácií. In: Zborník vojenskej akadémie. – Roč. 11, č. 1 (2004), s. 98-103 – ISSN 1335-0935
2. Všetečka, P. – Svetlík, M.: Vedomostný manažment. In: Zborník vojenskej akadémie. – Roč. 11, č. 2 (2004), s. 111-117 – ISSN 1335-0935
3. Hluchý L., Budinská I., Balogh Z., Forgáč R., Gatial E., Laclavík M., Mokriš I., Nguyen G.: Modeling of Military Training Activities by Ontology-based Knowledge System. Proc. of 6. Int. Conf. “Multimedia in Bussiness”, Kielce, Oct. 25- 27, 2006.
4. Forgáč, R..- Budinská, I.- Gatial, E.- Nguyen, G.- Laclavík, M.- Balogh, Z.- Mokriš I.- Hluchý, L.- Ciglan, M.- Babík, M.: Ontology Based Knowledge Management for Organizational Learning. Proc. of 9th Int. Conf. ISIM'06 – Information Systems Implementation and Applictions

Processing of information resources in Slovak language

Dostupné zdroje a výzvy pre počítačové spracovanie informačných zdrojov v slovenskom jazyku*

Michal Laclavík[1], Marek Ciglan[1], Stanislav Krajči[2],
Karol Furdík[3], Ladislav Hluchý[1]

[1] Ústav informatiky Slovenskej akadémie vied, Dúbravská cesta 9,
Bratislava, 845 07, Slovakia
laclavik.ui@savba.sk
http://ikt.ui.sav.sk/
[2] Ústav informatiky, Prírodovedecká fakulta, UPJŠ, Park Angelinum 9,
040 01 Košice, Slovakia
stanislav.krajci@upjs.sk
[3] Centrum pre informačné technológie, FEI TU v Košiciach, Letná 9,
040 01 Košice, Slovakia
Karol.Furdik@tuke.sk

Abstrakt. Článok je základnou informáciou o skupinách, ktoré sa zaoberajú počítačovým spracovaním slovenského jazyka. Takisto pojednáva o ich výsledkoch a výzvach na výskum a vývoj v tejto oblasti v budúcnosti. V krátkosti tiež rozoberá potreby projektu NAZOU pre spracovanie slovenčiny, kde sa ako základný problém javí lematizácia a stemming slovenčiny.

1 Kto sa zaoberá slovenčinou

Na Slovensku ale aj v Čechách funguje niekoľko skupín, ktoré sa zaoberajú spracovaním slovenčiny. Určite najvýznamnejším je pracovisko v Jazykovednom ústave Ľ. Štúra SAV (JÚĽŠ), kde sa tvorí korpus slovenského jazyka[1]. Korpus textov [1] predstavuje špecifický súbor jazykových dát, ktorý sa buduje v elektronickej podobe a spracováva sa na vedecko-výskumné a učebné ciele. Lingvisti na základe autentického jazykového materiálu opisujú predovšetkým významy a funkcie slov i ďalších jazykových prostriedkov. Bežným používateľom môže korpus poslúžiť ako zdroj poznania reálneho fungovania jazykových prostriedkov, nenahrádza však kodifikačné ani gramatické príručky. Popri korpuse sa pracuje aj na morfológii a tiež lematizácii [2] [11] slovenčiny, kde je aktívny najmä Radovan Garabík.

V minulosti bolo najvýznamnejším pracoviskom v tejto oblasti Laboratórium počítačovej lingvistiky na Pedagogickej Fakulte UK pod vedením Vladimíra Benka.

* This work is supported by projects K-Wf Grid EU RTD IST FP6-511385, NAZOU SPVV 1025/2004, RAPORT APVT-51-024604, VEGA 2/6103/6, VEGA 2/7098/27.

[1] http://korpus.juls.savba.sk/

V súčasnosti sa ďalej pokračuje v práci na JÚĽŠ SAV, kde boli odovzdané aj predchádzajúce výsledky.

Ďalším významným pracoviskom je Fakulta elektrotechniky a informatiky Technickej univerzity v Košiciach, kde paralelne pôsobí niekoľko skupín zaoberajúcich sa počítačovým spracovaním slovenčiny. Lingvistická dielňa na Katedre počítačov a informatiky FEI TU, vedená Jánom Genčim, realizuje projekty „Porovnávanie adjektív na základe regulárnych výrazov“, „Morfologická databáza slovenčiny“ a „Tvorba synsetov“ [21].

Na Katedre kybernetiky a umelej inteligencie FEI TU v Košiciach sa skupina Dušana Krokavca zameriava na aplikačný výskum v oblasti počítačového spracovania reči, Peter Sinčák skúma použitie neurónových sietí na analýzu prirodzeného jazyka, Ján Paralič a jeho skupina sa sústreďuje na aplikácie v oblasti dolovania znalostí z textov, sémantické reprezentácie a znalostné technológie. Ako istý integrujúci a syntetizujúci prístup možno tiež spomenúť prácu Karola Furdíka [20], v ktorej je predstavený komplexný model počítačového spracovania slovenčiny, kombinujúci lingvistické, štatistické a znalostné prístupy.

Relatívne novým pracoviskom je Centrum pre informačné technológie[2] FEI TU v Košiciach, kde sa vyvíja knižnica JBOWL[3] (Java Bag-Of-Words Library) pre podporu aplikácií spracovania prirodzeného jazyka a objavovania znalostí v textoch [22]. Tento softvérový systém pre manipuláciu s textovými dokumentmi poskytuje funkcie a metódy pre podporu spracovania prirodzeného jazyka (značkovanie, morfologická analýza, lematizácia, dezambiguácia, syntaktická analýza na báze ATN sietí, zhlukovanie a identifikácia fráz, váženie termov, indexácia), získavania znalostí a dolovania v textoch. Knižnica JBOWL je vytvorená v programovom prostredí Java a je realizovaná ako Open source projekt pod GNU Lesser licenciou.

V Košiciach sa slovenčinou zaoberá aj skupina na UPJŠ, kde bol vytvorený indexovací a fulltextový vyhľadávací nástroj, ktorý využíva aj dáta zo slovenského slovníka pretransformovaného do elektronickej podoby na UPJŠ [7] [8] [9]. Pracuje sa aj na zapracovaní ďalších metapoznatkov[4], napríklad slovníka cudzích slov[5]. V rámci projektu NAZOU sa vytvára aj nástroj *Tvaroslovník* [12], ktorý slúži na jednoduchú lematizáciu slovenčiny.

V Prešove na Filozofickej fakulte Prešovskej univerzity sa dlhodobo zaoberá výskumom v oblasti spracovania reči skupina pod vedením Jána Sabola a Júliusa Zimmermanna. Spomenúť však treba aj nemalý podiel pracovníkov Prešovskej univerzity na budovaní korpusu slovenského jazyka, predovšetkým práce na morfologickej a syntaktickej anotácii. V oblasti syntaxe sú významnými práce Jolany Nižníkovej a Miloslavy Sokolovej [23] [24], ktoré predstavujú prvý koncepčne ucelený projekt slovníka slovesnej valencie a jeho následné využitie v typológii vetných vzorcov. V oblasti slovotvorby sa možno zmieniť o softvérovom riešení slovenského slovotvorného slovníka [25], ktorý bol budovaný na základe koncepcie Juraja Furdíka.

---

[2] http://www.tuke.sk/fei-cit/

[3] http://sourceforge.net/projects/jbowl/

[4] http://s.ics.upjs.sk/~gallova/Slovnik/index1.htm

[5] http://s.ics.upjs.sk/~gotthardova/swprojekt/

V oblasti morfológie sú významné práce Eduarda Kostolanského [26], v súčasnosti pôsobiaceho na Univerzite Cyrila a Metoda v Trnave. Ako pokračovanie týchto prác sa na Katedre informatiky tejto univerzity v súčasnosti realizuje druhá etapa projektu budovania viacjazyčného korpusu pre počítačom podporovanú výučbu cudzích jazykov[6].

Za veľmi zaujímavý pokus o algoritmické uchopenie slovenčiny možno považovať prácu bývalého matematika Emila Páleša realizovanú v rámci projektu Sapfo [10]. Jeho nosnou myšlienkou je vytvorenie pravidiel vo forme logického programu, ktoré by zachytávali všetky nepravidelnosti slovenského jazyka. Na túto prácu chcela pôvodne nadviazať skupina nadšencov z PF UPJŠ, dnes spoluriešiteľov projektu NAZOU. Ukázalo sa však, že cesta zachytiť slovenskú gramatiku do takýchto exaktných pravidiel je vzhľadom na obrovský počet výnimiek nerealizovateľná. Prijali preto úplne inú paradigmu, a to nehľadať pravidlá, ale získať všetky gramatické tvary všetkých slov (presnejšie slov zo Slovníka slovenského jazyka, teda našej oficiálnej slovnej zásoby) a spracovať ich do podoby tzv. *Tvaroslovníka*. Tu si treba uvedomiť, že počet gramatických tvarov nepresahuje milión, a uchovávať toľko údajov v dnešnej dobe už nie je problém. Obsah spomínaného (šesťdielneho) Slovníka slovenského jazyka (zahŕňajúceho okrem slov v základnom tvare aj ich významy a príklady použitia v dielach klasikov slovenskej literatúry) sa zmestí do textového súboru o veľkosti okolo 17,5 MB, samotné slová z toho zaberajú iba 1,5 MB. Samozrejme, získanie všetkých tvarov slov je úloha netriviálna, ale na druhej strane jednorazová. Takáto množina všetkých slovných tvarov by nielen automaticky ponúkala slovný základ, ale bola by aj dobrým základom pre lingvistický výskum vyšších lexikálnych jednotiek (vetný rozbor). Dodajme, že v súčasnosti už existuje elektronická verzia Slovníka slovenského jazyka[7], avšak vzhľadom na jeho veľkú chybovosť ju možno považovať len za verziu pracovnú.

Výpočtovou analýzou viet v slovenskom jazyku sa zaoberá diplomová práca Michala Čerešňu [13], ktorá ponúka aj lexikálny analyzátor slovenčiny.

Dôležitým je aj seminár SLOVKO[8] International Seminar Computer Treatment of Slavic and East European Languages, na ktorom sa stretáva komunita ktorá pracuje s počítačovým spracovaním jazyka v rámci Čiech, Slovenska ale aj ďalších susedných krajín.

Na Slovensku existujú aj ďalšie aktivity v komerčnej ale aj neakademickej sfére. V tejto oblasti pracuje napríklad Forma s.r.o.[9], ktorá dodáva spell check pre produkty Microsoft napr. Microsoft Word. Takisto má však samostatné produkty na fulltextové vyhľadávanie v slovenčine ako aj prevádzkuje niektoré stránky[10] ktoré by mali podporovať vyhľadávanie v slovenčine. Ďalej je tu tiež aktivita sk-spell[11] ktorá vytvára open source spell check ktorý sa využíva napr. v OpenOffice. Iniciatíva obsahuje aj anglicko-slovenský slovník a synonymický slovník.

---

[6] http://ki.fpv.ucm.sk/index.php?start=projekty
[7] http://dbserver.ics.upjs.sk/slovnik
[8] http://korpus.juls.savba.sk/~slovko/
[9] http://www.forma.sk/
[10] http://www.zbierka.sk/
[11] http://sk-spell.sk.cx/

2 Lematizácia a stemming

Slovenčina patrí medzi jazyky ktoré majú bohatú morfológiu, teda tvar slova sa mení podľa významu. Toto môže byť zjavnou nevýhodou pri počítačovom spracovaní. Je teda potrebné hľadať základný tvar slova (*lemma*) prípadne koreň slova (*stem*). V angličtine sa tiež používa lematizácia alebo stemming aby sa odstránilo množné číslo, minulý čas slovies a ďalšie. V angličtine je najbežnejší Porterov algoritmus ktorý však pre slovenčinu nefunguje. Keď si predstavíme napríklad koreň slova „rada“, koreň je „rad“, pričom tento koreň zahŕňa pri uvažovaní bez diakritiky nasledujúce slová: rada – podstatné meno, orgán; rád – podstatné meno, vyznamenanie; rád – sloveso; rad – podstatné meno, zoradenie; rada – podstatné meno, ponaučenie.

Problémom je aj rôzne kódovania slovenčiny v ktorých sú zapísané informačné zdroje (napr. web stránky). Základné sú win-1250, ISO-8859-2 alebo UTF, ale tiež špeciálne HTML značky začínajúce „&#“. Veľa informačných zdrojov je písaných aj bez diakritiky. Napríklad ak chceme spracovávať emaily, ktoré sa bežne píšu bez diakritiky, je potrebne riešiť lematizáciu aj bez diakritiky.

Lematizácia a stemming sa bežne používa najme pri indexácii a následnom fulltextovom vyhľadávaní. Treba však povedať, že pre slovenčinu zatiaľ niečo podobné neexistuje v uspokojivom rozsahu.

Firma Forma s. r. o. deklaruje, že má vyriešené indexovanie a vyhľadávanie v slovenskom jazyku, a teda aj stemming. Pri vyhľadávaní na stránke www.zbierka.sk však toto funguje iba v obmedzenom rozsahu. Napríklad pri dopytoch „Národná rada“, „Národnej rady“ a bez diakritiky „Narodna rada“ je výsledkom rôzny počet dotazov, ktorý nezahŕňa všetky formy slova.

Podobne vyhľadávač morfeo.sk[12] deklaruje vyhľadávanie v slovenčine, ale napríklad pre dopyty „Štefan Luby“ a „Štefanovi Lubymu“ vráti rôzny počet stránok. Samozrejme tvary slov nie sú vyriešené ani vo vyhľadávačoch Zoznam a Google.

Na lematizátore a stemmeri pracujú aj na niektorých pracoviskách v ČR [3], nepodarilo sa nám však zistiť, ako sú tieto výsledky úspešné v slovenčine.

Zaujímavou pre tvorbu slovenského stemmeru sa javí práca Lea Galamboša[13], ktorý vyvinul stemmer vhodný pre slovanské jazyky [4, 5, 6], na základe ktorého bol vyvinutý stemmer pre poľský jazyk[14]. Na lematizátore [11] pre slovenčinu sa pracuje na JÚĽŠ SAV, ktorý je založený na slovníkovom princípe. Beta verzia je dostupná na webe[15].

V rámci projektu NAZOU je na UPJŠ vytváraný nástroj Tvaroslovník [12], ktorý bude slúžiť na lematizáciu slovenského jazyka. Tento nástroj vie na základe vzorov vygenerovať lemmu, ktorá sa následne kvôli presnosti overuje v slovníku.

Tvorba algoritmického stemmeru pre slovenčinu je potrebná, pretože stemmery založené na slovníkoch nevedia zistiť lemmy alebo korene slov pre nové slová, priezviská, mená miest alebo mená firiem v inom ako základnom tvare. Lematizácia

[12] http://www.morfeo.sk/
[13] http://kocour.ms.mff.cuni.cz/~galambos/
[14] http://getopt.org/stempel/
[15] http://korpus.juls.savba.sk/~garabik/junk/mlv/

pritom nie je potrebná len pre správnu indexáciu a fulltextové vyhľadávanie, ale aj pre sémantickú anotáciu, identifikáciu informačných zdrojov v rámci domény, a ďalšie. Tento problém by mohol čiastočne vyriešiť nástroj Tvaroslovník [12], prípadne bude potrebné vyvinúť nový nástroj na stemming slovenčiny na základe existujúcich prístupov [4, 5, 6].

3 Slovenčina v nástrojoch projektu NAZOU

Problém spracovania informačných zdrojov v slovenskom jazyku sa rieši aj v projekte NAZOU [14] ako súčasť druhej pilotnej aplikácie. V projekte NAZOU sa spracúvajú informačné zdroje z internetu. Pilotnou aplikáciou je doména pracovných ponúk. Tieto sa vyhľadávajú a identifikujú na internete, sťahujú, indexujú, ako aj ďalej spracúvajú vo forme ontologických metadát a následne sa prezentujú užívateľom. Práve na tejto aplikačnej doméne je viditeľná potreba stemmingu, pretože názvy miest a obcí kde sa má práca vykonávať môžu byť vyskloňované. Taktiež podobný problém môže vzniknúť pri názvoch firiem, pracovných pozícií alebo kategórií pracovných ponúk. Tiež v prípade vyhľadávania informácií o uchádzačovi na internete môže nastať problém, že nenájdeme všetky dostupné informácie kvôli rôznym tvarom mena a priezviska.

Zatiaľ sme v NAZOU identifikovali potrebu ekvivalentu Porterovho algoritmu pre slovenčinu, teda algoritmus pre lematizáciu alebo stemming slovenčiny, ktorý by využil nástroje na indexovanie [15] [16], nástroj Ontea [17] slúžiaci na sémantickú anotáciu a pravdepodobne nástroj Erid [18], ktorý potrebuje zisťovať relevanciu všetkých tvarov relevantných slov. Ďalej je potrebný nástroj, ktorý identifikuje zdroj v slovenskom jazyku – aby nástroje vedeli, či majú použiť Porterov alebo iný algoritmus. Tento problém rieši nástroj NALIT [19]. Tiež pre vyhľadávanie a prípadne aj anotáciu je vhodné použitie synonymického slovníka.

4 Záver

V článku v krátkosti informujeme o tom, kto sa zaoberá počítačovým spracovaním slovenčiny a čo bolo v tejto oblasti spravené. V ďalšom sa zameriavame na problémy lematizácie a stemmingu, kde je načrtnutá potreba riešiť tieto problémy, ktoré nám vyplynuli aj v rámci projektu NAZOU.

V ďalšej práci by sme sa chceli zamerať na integráciu existujúcich riešení pre lematizáciu slovenčiny do nástrojov v projekte NAZOU. Tiež by sme sa chceli zamerať na možné riešenie problému stemmingu v slovenskom jazyku pre potreby počítačového spracovania zdrojov v slovenskom jazyku a ich následnej indexácie a fulltextového vyhľadávania.

Literatúra

1. Garbík, Radovan (2003): Štruktúra dát v Slovenskom národnom korpuse a ich vonkajšia anotácia. In: Slovenčina na začiatku 21. storočia. Ed. Mária Imrichová. Prešov: Prešovská univerzita, Fakulta humanitných a prírodných vied 2004, s. 164 – 173.
2. Garabík, Radovan – Gianitsová, Lucia – Horák, Alexander – Šimková, Mária (2004): Tokenizácia, lematizácia a morfologická anotácia Slovenského národného korpusu. Interný materiál. http://korpus.juls.savba.sk/publications/block2/tokenizacia-lematizacia-a-morfologicka-anotacia-slovenskeho-narodneho-korpusu/Tagset-aktualny.pdf
3. Institute of Formal and Applied Linguistics, Faculty of Mathematics and Physics, Charles University, Prague, Czech Republic: http://ufal.mff.cuni.cz/, http://www.elsnet.org/survey/InstituteofFormalandAppliedLinguistics11835Praha1CzechRepublic/resources.html
4. Leo Galambos: Multilingual Stemmer in Web Environment. PhD Thesis, Faculty of Mathematics and Physics, Charles University in Prague, 2004.
5. Leo Galambos: Semi-automatic stemmer evaluation. Mieczyslaw A. Klopotek, Slawomir T. Wierzchon, Krzysztof Trojanowski (Eds.): Intelligent Information Processing and Web Mining, Proceedings of the International IIS: IIPWM'04 Conference held in Zakopane, Poland, May 17-20, 2004. Advances in Soft Computing Springer 2004, ISBN 3-540-21331-7.
6. Leo Galambos: Lemmatizer for Document Information Retrieval Systems in JAVA. Leszek Pacholski, Peter Ruzicka (Eds.): SOFSEM 2001: Theory and Practice of Informatics, 28th Conference on Current Trends in Theory and Practice of Informatics Piestany, Slovak Republic, November 24 - December 1, 2001, Proceedings. Lecture Notes in Computer Science 2234 Springer 2001, ISBN 3-540-42912-3.
7. NAZOU Projekt SPVV 1025/04 , Faza 1 report, 2005, strany 98-99
8. NAZOU Projekt SPVV 1025/04 , Faza 2 report, 2005, strany 11-12
9. Krajči S., Novotný, R., Mati M., Pribolová J. (2003). Slovník slovenského jazyka. Dokumentácia skupinového projektu. http://s.ics.upjs.sk/~slovnik/ (October 2006).
10. Páleš, E.: Sapfo – parafrázovač slovenčiny. Veda vydavateľstvo SAV, 1994, Bratislava.
11. Radovan Garabík: Slovak morphology analyzer based on Levenshtein edit operations, WIKT 2006
12. Stanislav Krajči, Róbert Novotný: Hľadanie základného tvaru slovenského slova na základe spoločného konca slov, WIKT 2006
13. Čerešňa, M.: Výpočtový model na analýzu viet slovenského jazyka. Diplomová práca, vedúci diplomovej práce J. Šefránek, Fakulta matematiky, fyziky a informatiky Univerzity Komenského v Bratislave. http://www.dbai.tuwien.ac.at/staff/ceresna/ling/nl-parsing-model.pdf, 2002.
14. Návrat, P., Bieliková, M.: Tools for Acquiring, Organising and Presenting Knowledge in heterogeneous environment of information sources. In: Paralič, J., Dvorský J., Krátký, M. (Eds.): Proc. Znalosti 2006 5th Annual Conference, 2006, 237-242.
15. Lencses R.: Indexing for the Information Retrieval System supported with Relational Database. In Sofsem 2005 Communications (editors: Vojtas et al), Slovenska informaticka spolocnost, Bratislava, 2005
16. Marek Ciglan, Marian Babik, Michal Laclavik, Ivana Budinska, Ladislav Hluchy: Corporate Memory: A framework for supporting tools for acquisition, organization and maintenance of information and knowledge; In: Proc. of 9-th Intl. Conf. ISIM'06 "Information Systems Implementation and Modelling", Brno, April, MARQ Ostrava, 2006, pp. 185-192, ISBN 80-86840-19-0.
17. Michal Laclavik, Martin Seleng, Emil Gatial, Zoltan Balogh, Ladislav Hluchy: Ontology based Text Annotation – OnTeA; Information Modelling and Knowledge Bases XVIII. IOS

Press, Amsterdam, Marie Duzi, Hannu Jaakkola, Hannu Kangassalo, Yasushi Kiyoki (Eds.), 2007

18. Gatial E., Balogh Z.: Identyfying, Retrieving and Determining Relevance of Heterogenous Internet Resaources. In: Tools for Acquisition, Organisation and Presenting of Information and Knowledge. P.Navrat et al. (Eds.), Vydavatelstvo STU, Bratislava, 2006, pp.15-21, ISBN 80-227-2468-8. Workshop 26-28 September, Nizke Tatry, Slovakia.
19. Peter Vojtek, Vladimir Grlicky: Identification of Natural Language using N-grams and Markov Processes. In: Tools for Acquisition, Organisation and Presenting of Information and Knowledge. P.Navrat et al. (Eds.), Vydavatelstvo STU, Bratislava, 2006, pp.154-161, ISBN 80-227-2468-8. Workshop 26-28 September, Nizke Tatry, Slovakia.
20. Furdík Karol: Získavanie informácií v prirodzenom jazyku s použitím hypertextových štruktúr. Doktorandská dizertačná práca. Katedra kybernetiky a umelej inteligencie FEI, Technická univerzita v Košiciach, Košice, 2003.
21. Genči Ján: Contribution to Processing of Slovak Language at DCI FEEI TUKE. In: Slovanské a východoeurópske jazyky v počítačovom spracovaní, Bratislava, 10.-12. november 2005, Bratislava, VEDA, 2005, s. 67-72, ISBN 80-224-0895-6. Dostupné na: http://korpus.juls.savba.sk/~slovko/2005/proc/slovko.pdf
22. Bednár Peter, Butka Peter, Paralič Ján: Java Library for Support of Text Mining and Retrieval. In: Proceedings of the 4th annual conference Znalosti 2005. Eds. L. Popelínský, M. Krátký. VŠB TU Ostrava 2005, s. 162 – 169.
23. Nižníková Jolana: Vetné modely v slovenčine. 1. vyd. Prešov: Filozofická fakulta Prešovskej univerzity, 2001. ISBN 80-8068-052-3.
24. Nižníková Jolana - Sokolová Miloslava: Valenčný slovník slovenských slovies. 1. vyd. Prešov: Filozofická fakulta Prešovskej univerzity, 1998. ISBN 80-88885-53-1.
25. Furdík Juraj, Furdík Karol: Slovotvorný slovník slovenčiny - softvérové riešenie. In: Varia 9. Zborník materiálov z IX. kolokvia mladých jazykovedcov (Modra-Piesok, 1.-3. december 1999). Ed. M. Nábělková, M.Šimková. Bratislava, Slovenská jazykovedná spolocnosť pri SAV, 2002, s.305-316.
26. Kostolanský Eduard – Hašanová, Jana – Benko, Vladimír: Model morfologickej databázy slovenčiny (počítačové spracovanie jazyka). UCM Trnava, 2004. 188 s. ISBN 80-89034-70-5.

Hľadanie základného tvaru slovenského slova na základe spoločného konca slov *

Stanislav Krajči[1], Róbert Novotný[2]
[1]stanislav.krajci@upjs.sk
[2]robert.novotny@upjs.sk

[1,2]Ústav informatiky, Prírodovedecká fakulta, UPJŠ Košice

Abstrakt Na základe jednoduchého pozorovania, že na ohýbaní slovenského slova má najväčší vplyv jeho koniec, sme implementovali algoritmus, ktorého ambíciou je nájsť základný tvar daného tvaru slova. V prvej fáze sa spomedzi dopredu definovaných známych tvarov vybratých slov vyberie také, ktoré má s daným slovom čo najdlhší spoločný koniec, a stane sa tak jeho „predlohou“, v druhej fáze sa pomocou základného tvaru tohto už definovaného slova „podvojnou zámenou“ odvodí možný základný tvar daného slova. V (nepovinnej) tretej fáze sa skontroluje prítomnosť takto vzniknutého základného tvaru daného slova v zozname slov slovenského jazyka.

1 Východiská

V rámci projektu NAZOU (Nástroje na získavanie, organizovanie a udržiavanie znalostí v prostredí heterogénnych zdrojov) je pri spracúvaní nových ponúk do vektorového (resp. objektovo-atribútového) modelu, v ktorom sa eviduje viac či menej sofistikovaná štatistika výskytu termov (slov) v dokumente, užitočné združiť rôzne tvary toho istého slova do jednej skupiny a vybrať jej vhodného reprezentanta, či už ide o spoločný slovný koreň týchto tvarov alebo ich základný tvar.

V dvoch jazykoch, ktoré nás zaujímajú, – angličtine a slovenčine – je však tento proces diametrálne odlišný. V anglickom jazyku, ktorý pozná ohýbanie slov len vo veľmi prostej podobe, sa spoločný slovný základ dosiahne jednoduchšie, a to vzhľadom na pomerne dobre definovateľné pravidlá tvorby slov zo spoločného slovného základu (isteže, existujú aj výnimky (napr. nepravidelné slovesá), tých je však relatívne málo). Obvyklou metódou je tu tzv. stemming (podľa Porterovho algoritmu ([2])), ktorý spočíva v odstraňovaní niektorej z mála prípon (napríklad „-ed“, „-ing“, či „-s“).

Slovenčina však patrí k tzv. flexívnym jazykom, kde má väčšina slov niekoľko (desiatok) tvarov, ktoré sa vytvárajú storakými spôsobmi. Časová investícia do hľadania a aplikácie pravidiel ohýbania slov tak môže byť príliš veľká, ba väčšia

* Podporené štátnym projektom výskumu a vývoja Budovanie informačnej spoločnosti, Nástroje na získavanie, organizovanie a udržiavanie znalostí v prostredí heterogénnych zdrojov, 1025/04

než (časovo náročné, ale v podstate jednorazové) nájdenie všetkých tvarov týchto slov. (Tu spomeňme veľmi zaujímavú prácu E. Páleša ([1]), ktorá túto obavu nechtiac potvrdzuje.) Táto možnosť veľmi dlho neprichádzala do úvahy, veď celá oficiálna slovná zásoba – Slovník slovenského jazyka ([4]) má v papierovej podobe šesť hrubých zväzkov, pričom obsahuje v podstate len základné tvary slov. Ak si však uvedomíme, že fakticky ide asi „len“ o 150 000 slov, ktoré majú spolu pár miliónov tvarov, úloha uložiť ich všetky do databázy už prestáva byť v dnešnej dobe nepredstaviteľná.

S týmto cieľom prebehla na Ústave informatiky UPJŠ v Košiciach začiatkom tohto tisícročia elektronizácia spomínaného Slovníka slovenského jazyka, ale aj Veľkého slovníka cudzích slov ([3]). Po dôkladnom (hoci ešte stále neúplnom) vyčistení dát sme tak získali v podstate kompletný zoznam 150 000 doteraz oficiálnych slovenských slov doplnených o 60 000 slov cudzieho pôvodu. (Sme si, samozrejme, vedomí, že slovenský jazyk sa vyvíja a slovná zásoba sa od vydania Slovníka slovenského jazyka značne zmenila. Napriek tomu si však myslíme, že získaný zoznam slov je však určite dobrým východiskom pre náš ďalší výskum.)

2 Hľadanie základného tvaru

Ako sme už naznačili, bohaté skúsenosti s hľadaním tvarov slova v angličtine sú do slovenčiny neprenosné, pre slovenský jazyk jednoducho pendant spomínaného Porterovho algoritmu neexistuje. Na vytvorenie základného tvaru daného slova použijeme (zámerne) *veľmi jednoduchú* metódu. Je založená na triviálnom pozorovaní, že *ohýbanie slova závisí od jeho konca, nie od začiatku.*

Vstupom nášho algoritmu je slovo v ľubovoľnom tvare (z technických dôvodov sa obmedzme len na podstatné mená, uvádzanú myšlienku však možno rovnako aplikovať i na ostatné (ohybné) slovné druhy), výstupom zoznam jeho možných základných tvarov. Predpokladáme pritom, že máme k dispozícii jednak spomínaný zoznam (základných tvarov) slovenských slov a jednak zoznam všetkých tvarov už vyskloňovaných podstatných mien. (V ideálnom prípade, o ktorom hovoríme v predchádzajúcej stati, by tento zoznam obsahoval všetky tvary všetkých postatných mien.) Algoritmus má tri fázy:

1. hľadanie zodpovedajúcich *predlôh*, t. j. zodpovedajúcich tvarov slov zo zoznamu vyskloňovaných podstatných mien
2. vytvorenie možných základných tvarov pomocou „podvojnej výmeny“
3. overenie prítomnosti možných základných tvarov v zozname všetkých základných tvarov (včítane kontroly rovnosti rodu s rodom príslušnej predlohy)

Ilustrujme tento algoritmus na konkrétnom príklade. Prepokladajme, že máme nájsť základný tvar slova „ponúk“, pričom sa toto slovo v zozname predlôh nenachádza, zato sa tam nachádza slovo „ruka“, a to včítane všetkých jeho tvarov. Zdôraznime, že predlohou nemusí byť žiaden dobre známy „školský“ vzor skloňovania („chlap“, „hrdina“, „dub“, „stroj“, atď.). Všimnime si tiež, že tu vôbec nerešpektujeme slovné základy, pri skloňovaní totiž nie sú dôležité.

- slovo: $X =$ „ponúk“
- jedna z nájdených predlôh: $Y =$ „rúk“
- spoločný (neprázdny) koniec: $K =$ „úk“
- začiatok predlohy: $Y' =$ „r“ (teda $Y = Y' + K$)
- začiatok slova: $X' =$ „pon“ (teda $X = X' + K$)
- základný tvar predlohy: $Y =$ „ruka“
- koniec základného tvaru ponuky: $K' =$ „uka“ (teda $Y = Y' + K'$)
- základný tvar slova: $X = X' + K' =$ „ponuka“
- overenie existencie slova X v zozname základných tvarov

Rovnako dobrou predlohou (za predpokladu, že by sa nachádzalo v ich zozname) by mohlo byť trebárs slovo „oblúk“, algoritmom v predposlednom kroku odvodený príslušný základný tvar „obluka“ by však neprešiel kontrolou v poslednom kroku, takýto základný tvar totiž neexistuje.

Ak je predlôh viac, uprednostníme tú, ktorá má s daným slovom najdlhší spoločný koniec. Tento postup, samozrejme, nevylučuje, že základných tvarov daného slova môže byť viacero. Ak sa však žiadny nenájde, dané slovo je vhodne vyskloňovať a doplniť ním zoznam predlôh.

3 Záver

V tomto článku sme sa pokúsili ilustrovať jednoduchú metódu na nájdenie základného tvaru slovenského slova. Pre potreby projektu NAZOU sa pokúsime rozšíriť jej funkčnosť o tieto črty:

- možnosť ignorovať diakritiku
- rozšírenie funkčnosti na slová mimo slovníka (napr. vlastné mená) (t. j. vynechanie tretej fázy algoritmu)
- rozšírenie na ostatné ohybné slovné druhy (hlavne prídavné mená a slovesá)

Ľahko si môžeme všimnúť, že uvedenú metódu možno prirodzeným spôsobom rozšíriť na hľadanie všetkých tvarov slova, teda nielen základného. (Aj pomocou takto upravenej metódy) získaný spomínaný zoznam všetkých tvarov slovenských slov by bol nielen užitočnou pomôckou pri kategorizovaní ponúk (a, samozrejme, aj iných textov), ale i dobrým východiskom pre automatickú syntaktickú analýzu slovenských viet.

Reference

1. Páleš, E.: Sapfo, parafrázovač slovenčiny, Veda, Bratislava, 1994. ISBN 80-224-0109-9.
2. Porter, M. F.: An algorithm for suffix-stripping, Program, 14 (3), pp. 130–137, 1980.
3. Šaling, S., Ivanová-Šalingová, M., Maníková, Z.: Veľký slovník cudzích slov, SAMO, 2003
4. kol.: Slovník slovenského jazyka, Vydavateľstvo SAV, Bratislava, 1959–1968

Information Retrieval by means of Vector Space Model of Document Representation and Cascade Neural Networks*

Igor Mokriš, Lenka Skovajsová

Institute of Informatics, Slovak Academy of Sciences, Bratislava, Slovakia
mokris@aoslm.sk, skovajsova@aoslm.sk

Abstract. The paper describes the information retrieval neural network model which retrieves the information from the text documents in Slovak language by Cascade Neural Networks and document representation by Vector Space Model.**Keywords:** information retrieval, Slovak language, queries, keywords, text documents, vector space model, cascade neural networks

1 Introduction

The aim of this paper is to describe the information retrieval neural network model which retrieves the information from the text documents in Slovak language and which, for this purpose, uses the neural networks. This model comes from linguistic, conceptual and knowledge approach for the analysis of text documents in Slovak language. For representation of the text document collection in Slovak language uses the vector space model. The neural network model, based on feed - forward and spreading activation neural networks, accepts the structure of linguistically, conceptually and knowledge oriented model, where query representation, document database creation and document indexing for keyword and document determination is solved. Proposed structure of the neural network model solves the document retrieval on the base of user's question. However, learning algorithm and neural network invariance, come from utilization of the neural networks enables to decrease the computational complexity of the Slovak language analysis algorithm.

2 Information retrieval neural network model

Neural network model of information retrieval system comes from the model based on statistical, linguistic and knowledge approach, which expresses document content and document relevance [1, 2, 7]. User specifies the query in Slovak language for

* This work was supported by Slovak Science and Techn. Assist. Agency under the contract No. APVT-51-024604 and Slovak Science Agency VEGA No. 2/7098/27

that system and system returns a Slovak document subset relevant to his query. The structure of the system was proposed and it is modular. It consists of the query subsystem, indexation subsystem and document subsystem (Fig.1) [3]. Because this approach is very complicated, what comes from inflection of Slovak language [8], this model can be simplified and expressed by cascade neural network (Fig.2) [4, 6].

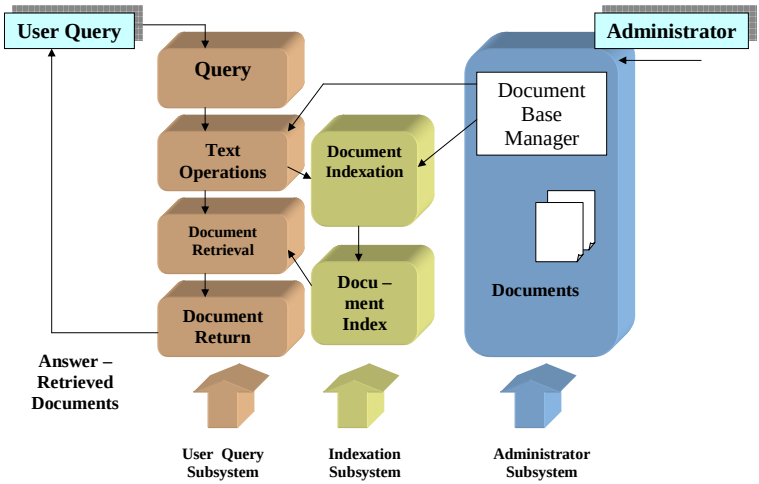


Fig. 1 Simplified structure of information retrieval system

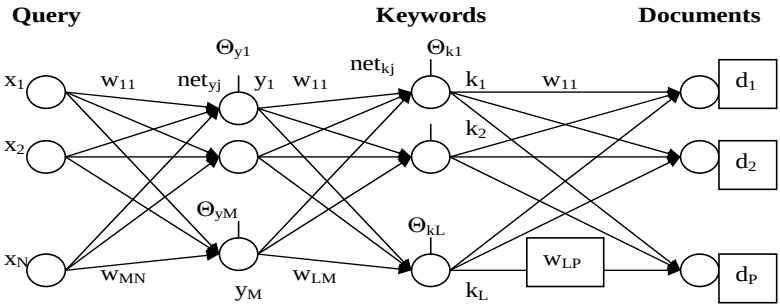


Fig. 2 Cascade neural network for determination of keywords and documents

This neural network model consists of two neural networks. 1st neural network is the feed – forward neural network of back propagation type, which solves the keyword determination by user question I Slovak language. It consists of 3 layers and its model is defined by equations (1–4), where x_i is the query representation and k_i is the keyword representation.

$$net_{yj} = \sum_{i=1}^{N} w_{ij} x_i(t) + \theta_{yj} \text{ , j=1...M} \quad (1)$$

$$y_j = f(net_{yj}) = \frac{1}{1+e^{-net_{yj}}}, \quad (2)$$

$$net_{kj} = \sum_{i=1}^{M} w_i y_i(t) + \theta_{kj} , \quad (3)$$

$$k_j = f(net_{kj}) = net_{kj} \quad (4)$$

2nd neural network is the spreading activation neural network, which solves the document retrieval. 2nd neural network expresses the document content and document relevance by vector space model [5, 9]. Vector space model is created by rows of keywords k_1, ..., k_L and columns of documents d_1, ... , d_p by relation (5), where k_i is i-th keyword, L is number of keywords, d_j is j-th document, P is number of documents. f_{ij} called also keyword weight is relative frequency of i-th keyword in j-th document and is used for creation of the matrix of vector space model by equation

$$F(L \times P) = \begin{pmatrix} k_1 \\ k_2 \\ \dots \\ k_L \end{pmatrix} = \begin{array}{c} \begin{matrix} d_1 & d_2 & & d_P \end{matrix} \\ \begin{pmatrix} f_{11} & f_{12} & \dots & f_{1P} \\ f_{21} & f_{22} & \dots & f_{2P} \\ \dots & \dots & \dots & \dots \\ f_{L1} & f_{L2} & \dots & f_{LP} \end{pmatrix} \end{array} \quad (5)$$

This neural network is not trained; its weights are set by matrix of vector space model by association of

$$\mathbf{W}_{ij} = \mathbf{F}_{ij} , \quad \text{i=1… P,} \quad \text{j=1... L} \quad (6)$$

and document relevance is determined by

$$d_j = f(net_{dj}) = net_{dj} \quad (7)$$

3 Experiments and results

Above mentioned model in MATLAB was programmed. In the query layer of 1st neural network there are 12 neurons, i.e. each neuron for one character of user query. In the keyword layer, there are 20 neurons, i.e. each neuron for one keyword. In the document layer, there are 90 neurons, i.e. each neuron for one document. The length of document is approximately about 50 words.

1st neural network was trained with a query training set, involves 164 queries and a keyword training set, which involves 20 keywords. Query testing set was created by chosen grammatical forms of the Slovak words. 2nd neural network uses keyword training set and document training set DS. This network was not trained because its weights were assigned to the matrix of vector space model (6).

Within the proposed model two experiments were made. First experiment was made with the first query test set, consisting of 185 queries, which contained different Slovak grammatical word forms. These forms were then used for the root bases of the keyword training set. For queries from the query testing set belonging keywords from the keyword training set were found and for them the documents from the document set with 0,9959 precision were found.

Second experiment was made with second testing set, where 100 queries were involved but no keyword belonged to it and this fact influences that no documents can be returned to the user correctly. From the results obtained, it follows, that the system reacted to chosen query training set with precision of 0.97.

From the whole assessment of the experiment, it follows, that the approach used has a perspective and provides next possibilities for its widespread.

References

1. Berg, J., Schuernie, M.: Information Retrieval Systems using Associative Conceptual Space. European Symposium on Artificial Neural Networks. 1999, ISBN 2-600049-9-X, pp. 351-356.
2. Cunningham, S.J., Holmes, G., Littin, J., Beale, R., Witten, I.H.: Applying Connectionist Models to Information Retrieval. In: S. Amari, and N. Kasobov (eds.), Brain-Like Computing and Intelligent Information Systems, Springer-Verlag, 1997, pp. 435-457.
3. Furdík, K.: Information Retrieval in Natural Language by Hypertext Structures. [PhD thesis], FEI TU Košice, 2003, (in Slovak).
4. Chen, H.: Machine learning for Information Retrieval: Neural Networks, Symbolic Learning, and Genetic Algorithms. Journal of the American Society for Information Science, 46 (3), 1995, pp. 194 - 216.
5. Vector Space Model (VSM). http://isp.imm.dtu.dk/thor/projects/multimedia/textmining/node5.html
6. Mokriš, I. - Skovajsová, L.: Neural Network Model of System for Information Retrieval from Text Documents in Slovak Language**.** Acta Electrotechnica et Informatica, No.3, Vol.5, ISSN 1335-8243, 2005, pp. 36-41.
7. Motta, E.: Reusable Components for Knowledge Models: Principles and Case Studies in Parametric Design. IOS Press, Amsterdam, 1999.
8. Páleš, E.: Sapfo - Slovak Para – Phraser. 1. issue. ISBN 80-224-0109-9, VEDA, Bratislava, 1994, (in Slovak).
9. Raghavan, V. V., Wong, S. K. M.: A Critical Analysis of Vector Space Model for Information Retrieval. Journal of the American Society for Information Science, Vol.37 (5), 1986, pp. 279-87.

Text Document Space Dimension Reduction by Latent Semantic Model*

Lenka Skovajsová, Igor Mokriš

Institute of Informatics, Slovak Academy of Sciences, Bratislava, Slovakia
lenka_skovajsova@post.sk, mokris@aoslm.sk

Abstract. This paper describes the Latent Semantic Model, which enables the information retrieval from Slovak documents based on the user query. This model comes out from the Vector Space Model, which for document set uses the full index representation. Main advantage of Latent Semantic Model in relation to the Vector Space Model is the great reduction of matrix dimension for document set representation.

Keywords: information retrieval, keywords, documents, Slovak language, Vector Space Model, Latent Semantic Model

1 Introduction

This paper describes the information retrieval from Slovak text documents based on Latent Semantic Model (LSI), which comes out from the Vector Space Model (VSM). The Latent Semantic Model uses the Singular Value Decomposition (SVD) algorithm and dimension reduction of the document set space representation in relation to original Vector Space Model.

2 Vector Space Model

The VSM is most often used for information retrieval [6,7]. The documents from document base are prepared at first. The stop-words are removed from documents and the other words are stemmed by Porter's algorithm. These stemmed words are then turned to keywords. It is then determined from keywords, how many times each keyword appears in the specific document, what is called frequency of the keyword in the document. From acquired information the VSM matrix is created. The element k_{ij} determines the frequency of keyword *i* in the document *j*. The VSM matrix has the following form:

* This work was supported by Slovak Science and Techn. Assist. Agency under the contract No. APVT-51-024604 and Slovak Science Agency VEGA No. 2/7098/27

$$K(m\times n)=\begin{pmatrix} k_{11} & k_{12} & \dots & k_{1m} \\ k_{21} & k_{22} & \dots & k_{2m} \\ \dots & \dots & \dots & \dots \\ k_{n1} & k_{n2} & \dots & k_{nm} \end{pmatrix} \quad \textbf{(1)}$$

The query *q* is brought as an input to the VSM

$$q=(q_1,...,q_n) \quad \textbf{(2)}$$

This query is represented by the keyword set and is multiplied with VSM matrix *K* and the vector of document relevance d is created

$$q\times K=d=(d_1,...d_m) \quad \textbf{(3)}$$

where each element d_j in this vector determines the relevance of document *j* to given query q.

The VSM could be very easily modeled and it is very easy work with it. For retrieval of relevant document set to the query it suffices to transform the query into the keyword vector, *q* to multiply it with VSM matrix K and arrange the output vector of document relevance d from the most relevant documents to the less relevant documents. Such arranged documents are sent to the user as an output.

One disadvantage of VSM is its high computing complexity by large size of VSM matrix, when here are lots of documents and/or keywords. Here comes the dimension reduction with LSI.

3 Latent Semantic Model

Latent Semantic Model (LSI) has as its input the VSM matrix [1,2,3,4,5,8]. On the VSM matrix, the Singular Value Decomposition (SVD) algorithm is applied, which decomposes the VSM matrix into three matrices:

$$K=U\times S\times V \quad \textbf{(4)}$$

where U is the matrix of orthogonal unit-length rows, S is the diagonal matrix of positive singular values and V is the matrix of orthogonal unit-length columns. The diagonal matrix of positive singular values elements S is positioned in the diagonal

of the matrix S and singular values are ordered from the highest value to the lowest value.

The dimension reduction of matrix K resides in elimination of a certain number of singular values positioned in the lowest part of the S matrix diagonal ($s_i \approx 0$) and their substitution them by the value $s_i=0$. From the matrix U in relation to the removed singular values s_i the last columns are removed and from the matrix V the last rows are removed. The approximated matrix K_r have the following form:

$$K_r = U_r \times S_r \times V_r \tag{5}$$

And the document relevance is acquired from the equation:

$$d_r = q \times K_r \tag{6}$$

When the user query (2) is entered as an input to equation (3), this query by the reduced matrix K_r (5) is multiplied and as a result the new document relevance vector d_r (6) is acquired. The Latent Semantic Model retains the most important relations between documents and the less important relations neglects.

LSI enables the reduction on the number of elements stored in memory. Besides the whole VSM Matrix are in the memory stored only three reduced matrices of LSI that have much less capacity as original Vector Space Model matrix.

4 Experiments

On the base of the upper approach, there was proposed model of 90 Slovak documents with 20 keywords. Over the documents there was created VSM matrix (1) for 20 keywords and 90 documents. The VSM matrix K by the SVD algorithm (4) was decomposed. From the matrix of positive singular values there was sequentially removed the lowest singular values. For each number of remaining singular values was calculated the absolute and the relative number of elements k_{ij} in the reduced VSM matrix K_r after approximation the number of singular value $s_i \approx 0$ in this matrix and the precision P and recall R with the comparison of the original VSM matrix K. Precision P and recall R were computed by the following formulas

$$P = n_{retrel} \,/\, n_{ret} \tag{7}$$

$$R = n_{retrel} \,/\, n_{rel} \tag{8}$$

where n_{retrel} is number of retrieved relevant documents, n_{ret} is number of retrieved documents and n_{rel} is number of relevant documents. The results in the

Tab.1 for approximated number of singular values are made as an average of precisions and recalls for each keyword.

Table 1. Precision, recall and number of elements in reduced matrices depending on number of singular values

			Number of elements k_{ij}	
Number of s_i	Precision P	Recall R	Absolute	Relative
1	0,7942	0,24	110	0,632184
3	0,95	0,4048	137	0,787356
5	0,975	0,5118	148	0,850575
7	0,9775	0,6342	161	0,925287
10	1	0,7542	165	0,948276
15	1	0,95	173	0,994253
20	1	1	174	1

5 Conclusion

From the results shown in Tab. 1 it follows that this approach is perspective for next investigation.

References

1. Ando, R. K.: Latent Semantic Space: Iterative Scaling Improves Precision of Interdocument Similarity Measurement. ACM SIGIR, Greece, 2000, pp. 216-224.
2. F.Y.Y. Choi, P.Wiemer, Hastings, J.More: Latent Semantic Analysis for Text Segmentation. Conf. of EM in NLP. 2001, pp. 109-117.
3. Cristianini, N. et. al.: Latent Semantic Kernels. Journal of Intelligent Information Systems, 2004, pp. 127-152.
4. Deerwester, S. et.al.: Information Retrieval using a Singular Value Decomposition Model of Latent Semantic Structure. ACM SIGIR, 1988, pp.465-480.
5. Deerwester, S. et. al.: Indexing by Latent Semantic Analysis. Journal of the American Society for Information Science, 41(6), 1990, pp.391-407.
6. Dominich, S. et. al.: Connectionist Interaction Information Retrieval. Modeling Vagueness and Subjectivity in Information Access. Vol 39(2), 2003, pp. 167-193.
7. Liu, G.Z.: Semantic Vector Space Model, Implementation and Evaluation. Journal of the American Society for Information Science, 1997, pp. 395-417.
8. Syu, I., Lang, S. D., Deo, N.: Incorporating Latent Semantic Indexing into a Neural Network Model for Information Retrieval. Proc. of the Fifth International Conference on Information and Knowledge Management, 1996, pp. 145-153.

Data Extraction from Documents – Emerging Problems and Solutions*

Viktor Oravec

Institute of Informatics, Slovak Academy of Sciences, Dubravská cesta 9,
845 07 Bratislava, Slovak Republic
viktor.oravec@savba.sk

Abstract. In nowadays, data extraction from documents is one of the most important tasks in an automatic information retrieval from internet documents. This paper addresses a state of the art, present problems and a future work in this field. Also it presents an algorithm for an estimation of a general document structure.

Keywords: Information extraction, document structure estimation

Introduction

Internet is a huge repository of documents with various structures which cover information value of documents. Retrieving of this information from an Internet document is an important task in knowledge retrieval. This problem has been addressed by recent research [1, 2], even on more specific level, where only the specific structure of a document is considered. For these purposes many wrappers have been developed [2], however a general approach has not been solved yet. The aim of this paper is to present actual state of the art of general information retrieval from Internet documents and emphasize main practical problems, such as hidden commercials removal or learning.

Recursive bounded frequent substring search

In this section, a novel recursive bounded frequent substring search is presented, which makes exhaustive search for frequent substrings in processed document converted into text using html-to-text converters. Frequent substring is substring which is located in all documents with similar structure. Afterwards, such frequent

* This work is supported by projects NAZOU SPVV 1025/2004, RAPORT APVT-51-024604 VEGA 2/7101/27 a APVV LPP-0231-06

substrings are marked as a part of the document's structure with no useful information. This type of search encounters several following problems: lost of html structure; complexity; frequent substrings may include useful information. However this algorithm introduces following advantage: general approach which results in bounded optimal solution.

Implementations

In this section full paper will describe architecture and implementation of recursive search method in JAVA language utilizing *htmlparser* library using UML diagrams. Also a description of a method's configuration based on XML document is proposed.

Emerging problems and future work

Resent research in field of information extraction from internet documents results in many problems that have to be solved to achieve high quality information retrieval. This paper addresses following problems:

- Lost of html structure: Loosing html structure of retrieved information can reduce an expressiveness of information.
- General solution: Algorithms for information retrieval from limited amount of structures is not an issue. General solution has to be found.
- Learning: Using intelligent approaches instead of traditional exhausted search.
- Hidden commercials: Usually commercials are not only included in banners and side bars of document, but also in a text part of a body of the document. Detection of such commercials can improve a quality of subsequent annotation.

References

1. Alberto H. F. Laender, et al. A brief survey of web data extraction tools. ACM SIGMOD Record, 31(2), pp. 84-93, 2002
2. Sugibucchi T., Tanaka Y. "Interactive web-wrapper construction for extracting relational information from web documents", Poster session on International World Wide Web Conference, 2005, Chiba, Japan, ISBN 1-59593-051-5
3. Oravec V., Nguyen G., "Offer Extraction and Separation for Internet Documents", Tools for Acquisition, Organisation and Presenting of Information and Knowledge (Editors: Návrat P., Bartoš P., Bieliková M., Hluchý L., Vojtáš P.), 2006, Vydavateľstvo STU, ISBN 80-227-2468-8

Semantic web technologies

RDF Suite – Case Study

Peter Smatana, Peter Bednár

Centre for Information Technologies
Technical University of Košice, Boženy Němcovej 3, 042 00 Košice
{Peter Smatana, Peter Bednár}@tuke.sk

Abstract. RDF (Resource Description Framework) is a family of World Wide Web Consortium (W3C) specifications originally designed as a metadata model using XML but which has come to be used as a general method of modeling knowledge. We could use many open source tools to effective manipulate with RDF. One of them is RDF Suite, developed at FORTH-ICS, which will be described in this paper. Basic information will be supported by our experience.

Introduction

Huge amount of data is the greatest problem of nowadays. These data could be stored on digital medium in different formats, such as video, audio or mainly text. Metadata (e.g. keywords, category, language and author) is needed to effective browsing of stored data. Resource Description Framework[1] (RDF) is a family of World Wide Web Consortium[2] (W3C) specifications originally designed as a metadata model using XML but which has come to be used as a general method of modeling knowledge. We could use many open source tools to effective manipulate with RDF. One of them is RDF Suite, developed at FORTH-ICS[3]. The goal of this paper is present some information about that tool and our experience with its.

RDF Suite

RDF Suite are the high – level scalable tools for the semantic web. Main components of this system are (see Fig.1):

- The Validating RDF Parser (VRP): the RDF Parser supporting semantic validation of both resource descriptions and schemas.
- The RDF Schema Specific Data Base (RSSDB): the RDF Store using schema knowledge to automatically generate an Object-Relational (SQL3) representation of RDF metadata and load resource descriptions.

[1] http://www.w3.org/RDF/
[2] http://www.w3.org/
[3] http://www.ics.forth.gr/

– The RDF Query Language Interpreter (RQL): the Declarative Language for uniformly querying RDF schemas and resource descriptions 7.
– The RDF Update Language Interpreter (RUL): declarative update language for RDF graphs which is based on the paradigms of query language RQL 4.

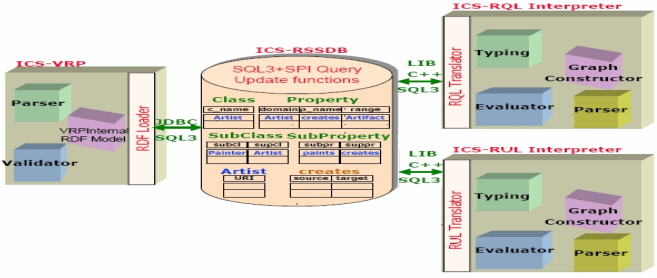


Fig.1 RDF Suite Architecture 6

Main advantages of RDF Suite toward other RDF tools (e.g. Jena, Seasame, RDF Store and KAON) are in the method of effective representation of the triplets in database and in design of query and update language for RDF.

RDF Suite will be used as the knowledge repository for integrated project KP-Lab, where team of the Technical University of Kosice is responsible for design a development of knowledge technologies middleware 1.

Acknowledgement

The work presented in this paper was supported by European Commission DG INFSO under the IST program, contract No. 27490 within the KP-Lab project.

References

1. F. Babič, J. Paralič, P. Smatana, P. Smrž: Knowledge Practices Laboratory, ICTE 2006
2. The ICS-FORTH RDFSuite: High-level Scalable Tools for the Semantic Web, http://139.91.183.30:9090/RDF/
3. G. Karvounarakis, A. Magkanaraki, S. Alexaki, V. Christophides, D. Plexousakis, M. Scholl, K. Tolle: Querying the Semantic Web with RQL. Computer Networks and ISDN Systems Journal, Vol. 42(5), August 2003, pp. 617-640. Elsevier Science
4. M. Magiridou, Stavros Sahtouris, Vassilis Christophides, Manolis Koubarakis, RUL: A Declarative Update Language for RDF, 2005, Fourth International Semantic Web Conference (ISWC'05), Galway, Ireland, November

Using Ajax for RDF/OWL processing*

Emil Gatial and Zoltán Balogh

Institute of Informatics, Slovak Academy of Sciences, Bratislava, Slovakia
emil.gatial@savba.sk

Abstract. The key concept of web browser's RDF/OWL processing rose from merging the ideas of Web 2.0 and semantic Web framework. While main keynote of Web 2.0 is "Network as platform" that stands for delivering applications entirely through the web browser, the key idea of semantic Web is to provide documents with computer meaning on WWW. The results of such fusion will provide seamless semantic document communication over the Internet. The technologies such AJAX and client side XML document processing enable web developers to write semantic-aware web browser applications and access Internet resources in asynchronous manner. The main role of this article is to provide a brief overview on such emerging frameworks and point out the advantages of such approach on a short example of the client side RDF/OWL browser application.

Introduction

The contemporary Web development user interface focuses primary on the server side technologies and overlooks possibilities of technologies integrated within the last generation of web browsers. In many cases the appropriate combination of used server/client technologies can show the best way for the web developers. The idea to develop ontology browser application by using technologies like AJAX, JavaScript, XML and XSLT rose as a small part of the IST K-Wf Grid project. Following text will try to explain the key concept of client side ontology processing.

First of all, the data exchange format must be specified. Even though, the name of AJAX (stands for Asynchronous Javascript And XML) [1] methodology may evoke that the XML format takes place for data exchange, but in general, any kind of data can be transferred using a communication method. Ajax's most appealing feature, however, is its asynchronous nature, which means that the responses can be handled in any time and therefore the client application is dynamically changed over the time. Ontology browser application uses OWL format [2] to transfer data from server to client. The client application should be able to process commands as well (automatic update events, modification events). An *XMLHttpRequest* object should be initialized [3] at the start point of client application. Another issue covers parsing the OWL data, extracting useful information and creating the content of web page. This issue is real

* This work is supported by projects K-Wf Grid EU RTD IST FP6-511385, NAZOU SPVV 1025/2004, RAPORT APVT-51-024604, VEGA No. 2/6103/6, VEGA 2/7098/27

challenge, because implement full specification of OWL format in detail is an overwhelming mission. Even though, for browsing purposes it is sufficient to handle only smaller bunch of OWL tags. The OWL tag handling can be made using methods of the *XMLDocument* object [5]. The following paragraph briefly explains the OWL tag processing, which is followed by short program code describing the main loop.

- Ontology importing: In this stage the *owl:import* tags must by processed in order to find other parts of ontology. Importing process should start by construction of the ontology import tree, where each of ontology URI is stored in the ordered list as like the breadth search (top-down) and then the ontology is processed in reverse order to guarantee the completeness of ontology class and individual elements.
- Processing ontology classes and properties: This step handle extracting the *owl:Class* tags and construction of dynamic tree by processing the *rdfs:subClassOf* tags and store the tree structure in DOM. The properties are extracted by processing *owl:ObjectProperty* and *owl:DataProperty* tags and stored in existing DOM class object.
- Processing ontology instances: Instances are identified by ontology prefix, which has to be found within the set of loaded ontologies. Such extracted information are stored in an attribute of DOM class object.

```
var rootRDF = xmlDocument.getElementsByTagName("RDF");
for(var id in rootRDF.childNodes) {
   xmlElement = rootRDF[0].childNodes[i];
   if(xmlElement.tagName == "owl:Class") {
      createClassElement(xmlElement);
   }
   else if(xmlElements.tagName == "owl:DataProperty") {
      …
   }
   else if(nsMap[xmlElement.prefix]) {
      createInstanceElement(xmlElelemt);
   }
   else …  // hande ObjectProprty, Restrictions, etc.
}
```

Conclusion

The shortened description of client side ontology processing using JavaScript and AJAX methodology tries to show best practice in development of web applications that take advantage of ontological description and web browser built-in technologies.

References

1. AJAX: Getting started, http://developer.mozilla.org/en/AJAX:Getting_Started
2. OWL Web Ontology Language Reference, http://www.w3.org/TR/owl-ref
3. XMLDocument object, http://www.xulplanet.com/references/objref/XMLDocument.html

Tvorba nezávislého rozhrania pre ontologickú organizačnú pamäť

René Pázman

Softec s.r.o.
rene.pazman@softec.sk
http://www.softec.sk/, http://rene.pazman.googlepages.com/

Abstrakt V rámci projektu na vývoj nástrojov na spracovanie informácií a znalostí z heterogénnych zdrojov používame organizačnú pamäť ako centrálne úložisko dát. V príspevku opisujeme náš prístup pri tvorbe rozhrania na ontologickú časť organizačnej pamäte a použitie návrhových vzorov pri jeho implementácii. Zameriavame sa na vytvorenie vrstvy pre taký prístup k ontológii, ktorý je nezávislý od implementácie ontologického úložiska.

Kľúčové slová RDF, ontológia, ontologické úložisko, Sesame, návrhové vzory

1 Úvod

V projekte NAZOU ([4]) riešime vývoj nástrojov pre získavanie, organizovanie a udržovanie znalostí v prostredí heterogénnych informačných zdrojov. Ide o štátnu úloha výskumu a vývoja, ktorá je približne v dvoch tretinách plánovaného trvania.

Cieľom projektu je navrhnúť metódy na získavanie, organizovanie, udržiavanie a poskytovanie informácií z internetu a tieto metódy implementovať formou nástrojov. Nástroje sú v projekte testované a vyhodnocované v pilotných aplikáciách.

Spracovávané informácie sa týkajú ohraničenej informačnej domény. Pre otestovanie nástrojov v rámci pilotných aplikácií sme zvolili doménu pracovných ponúk (zbieranie pracovných ponúk na internete a poskytovanie informácií o nich potenciálnym uchádzačom o prácu).

Informácie získané z vybraných zdrojov sú uložené v systéme. Pre tento účel systém (pilotná aplikácia) obsahuje úložisko dát a informácií. Informácie pre používateľov sú poskytované z tohto úložiska. Cieľová forma reprezentácie informácií sú štruktúrované dáta — ontológia.

2 Organizačná pamäť v projekte

Pôvodný návrh architektúry pilotnej aplikácie predpokladal tesnú funkčnú spoluprácu nástrojov realizovanú pomocou priamych funkčných prepojení. Pre návrh

takejto architektúry sa zvolila forma viacerých pohľadov na aplikáciu. Boli navrhnuté jej podsystémy, prípady použitia a komponenty. Informačná doména sa zovšeobecnenila na doménu ponúk s archetypmi aktérov a dát — ponuka, producent a konzument ([5]). Návrh predpokladal doménovo špecifický prístup k údajom, teda niektoré funkčné rozhrania boli závislé od zvolenej (aj keď zovšeobecnenej) domény.

Takýto návrh architektúry sa ukázal v našom projekte priamo nepoužiteľný. Dôvodov bolo viacero, najmä geografická distribuovanosť vývoja, nehomogénnosť znalostí členov tímu a ich časovej alokácie v projekte, neexistencia detailného návrhu cieľovej funkčnosti aplikácie a potreba opakovaného použitia nástrojov v iných doménach a v inej vzájomnej zostave.

Aktuálna pilotná aplikácia je vytvorená na voľnejšom základe. Ide o architektúru štýlu tabuľa (blackboard), kde nástroje sú samostatné jednotky, pracujúce nad organizačnou pamäťou a komunikujúce najmä cez organizačnú pamäť. Takéto uvoľnenie väzieb medzi nástrojmi umožnilo o.i. aj lepšiu distribuovateľnosť prác v projekte a ťažisko prác sa mohlo presunúť z riešenia integračných konfliktov na návrh a overovanie metód spracovania informácií. Organizačná pamäť sa týmto stala dôležitým centrálnym prvkom.

Organizačná pamäť sa v projekte NAZOU skladá z troch vrstiev: interakčnej vrstvy na vzdialený prístup k pamäti, manipulačnej vrstvy s programátorskými rozhraniami pre nástroje a fyzickej vrstvy, v ktorej sú jednotlivé úložiská. Fyzická vrstva je realizovaná úložiskami troch typov — súborovým, relačným a ontologickým.

Pre realizáciu ontologického úložiska bol zvolený systém Sesame ([1]). V ďalšom texte opisujeme náš prístup pri návrhu rozhrania na ontologické úložisko.

3 Ontologické rozhranie

Pri návrhu rozhrania na ontologické úložisko sa brali do úvahy viaceré požadované vlastnosti:

1. *Nezávislosť od informačnej domény.* Aj keď v pilotnej aplikácii používame doménu pracovných ponúk, jadro nástrojov je implementované nezávisle od informačnej domény — z toho dôvodu, aby sa nástroj dal použiť aj v inej doméne. Takto sa navrhla aj architektúra nástroja, keď sa upustilo od pôvodnej architektúry systému.
2. *Nezávislosť od úložiska.* Nástroje nemajú byť pevne viazané na zvolené úložisko Sesame, ale mali by byť bez väčších zmien použiteľné aj s inými úložiskami (napr. Jena).
3. *Jednoduchosť.* Rozhranie by malo byť jednoduché a zrozumiteľné pre autorov nástrojov.
4. *Zachovanie výkonnosti.* Rozhranie nesmie zhoršiť výkonnosť úložiska, teda jeho použitie nesmie znížiť odozvu úložiska oproti použitiu natívneho rozhrania úložiska.

5. *Univerzálnosť.* Rozhranie má poskytovať všetky potrebné možnosti úložiska, najmä na vyhľadávanie pomocou špecializovaných dopytovacích jazykov, ale má tiež umožňovať prácu s indivíduami, RDF grafmi a pod.

Rozhranie na ontológiu tvoria tri vrstvy. V spodnej vrstve je natívne rozhranie úložiska, v našom prípade Sesame API, ktoré je tvorené Java rozhraniami (interfaces) a triedami a je súčasťou úložiska Sesame.

Vrchná vrstva je definíciou nezávislého rozhrania, ktoré tvorí fasádu ontologického úložiska pre jeho používateľov (nástroje). Táto vrstva je tvorená výhradne Java rozhraniami. Nástroje pre svoju prácu poznajú iba túto vrstvu.

Stredná vrstva spáje tieto dve vrstvy — adaptuje natívne rozhranie úložiska (spodnú vrstvu) na nezávislé rozhranie pre nástroje (vrchnú vrstvu). Je tvorené Java triedami, ktoré realizujú Java rozhrania z vrchnej vrstvy a používajú Java rozhrania a triedy zo spodnej vrstvy.

V prípade potreby použitia iného ontologického úložiska (ako napr. Jena) sa spodná vrstva vymení za natívne rozhranie úložiska (Jena API) a vytvorí sa stredná vrstva pre nové úložisko — tú je treba implementovať nanovo. Vrchná vrstva sa nezmení, čo umožní zmenu úložiska bez zmeny implementácie nástrojov.

Vzhľadom na túto situáciu sme pre implementáciu rozhrania použili najmä návrhový vzor *Adapter* ([3]), známy aj ako *Wrapper*, obaľovač. Pre realizáciu dopytovania pomocou špecializovaných dotazovacích jazykov (ako napr. SeRQL a RDQL) sa použil tento návrhový vzor 4-krát. Jedno použitie vzoru umožnilo obaliť spojenie na ontologické úložisko a ďalšie tri použitia zabezpečujú prístup k výsledkom dopytov (pre výslednú tabuľku, riadok výslednej tabuľky a bunku takéhoto riadku).

Okrem tohto základného vzoru sa použili aj ďalšie vzory:

- *Abstract Factory* ([3]). Vzor je použitý pri vytváraní spojenia na ontologické úložisko. Použitie vzoru umožňuje dynamickú zmenu implementácie rozhrania ontologického úložiska (umožňuje prechod medzi Sesame a Jena). Výber implementácie rozhrania je riadený nastavením v konfiguračnom súbore.
- *Singleton* ([3]). Vzor sa používa pri vytváraní spojenia na ontologické úložisko pri testovaní.
- *Dependency Injection* ([2]). Tento vzor sa využíva v projekte pre takú vzájomnú integráciu nástrojov, ktorá umožňuje voľné a konfigurovateľné väzby medzi nimi. Vzor je možné použiť aj pre vytvárenie spojenia na ontologické úložisko.

4 Záver

Navrhnuté a implementované rozhranie spĺňa stanovené požiadavky — je nezávislé od úložiska a informačnej domény, jednoduché a prehľadné a efektívne (napr. netransformuje zbytočne výsledky vyhľadávania). Poskytuje potrebné funkcie na dopytovanie ontológie a na prácu s výrokmi (statements), RDF grafmi a inštanciami (indivíduami).

Rozhranie je v projekte realizované pre Sesame, ale dá sa implementovať aj pre iné úložisko, napr. Jena. Rozhranie v súčasnosti poskytuje základné funkcie pre prácu s ontológiou. Plánujeme ho rozširovať o ďalšie funkcie, ktoré Sesame poskytuje.

Uvažujeme tiež s jeho rozširovaním o funkcie na udržiavanie dát a propagovanie zmien v údajoch medzi nástrojmi. Tieto funkcie by mali byť doménovo nezávislé.

Okrem toho zvažujeme vytvoriť nad ním doménovo špecifickú nadstavbu, ktorá by umožňovala jednotnú prácu s doménovými objektmi. V tomto prípade by však doménová závislosť mala ostať na úrovni pojmov znovupoužiteľných v iných príbuzných doménach, ako sú napr. pojmy ponuka, poskytovateľ ponuky, záujemca o ponuku, všeobecné preferencie záujemcu o ponuku a pod.

Referencie

1. Broekstra, J., Kampman, A., van Harmelen, F.: Sesame: An Architecture for Storing and Querying RDF Data and Schema Information. In: Fensel, D., Hendler, J., Lieberman, H., Wahlster, W. (Eds): Semantics for the WWW, MIT Press. Available at http://www.cs.vu.nl/ frankh/postscript/MIT01.pdf. (2001)
2. Fowler, M.: Inversion of Control Containers and the Dependency Injection pattern. Available at http://www.martinfowler.com/articles/injection.html. (2004)
3. Gamma, E., Helm, R., Johnson, R., Vlissides, J.: Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley, Professional Computing Series. (1995)
4. Návrat, P., Bieliková, M., Rozinajová, V.: Methods and Tools for Acquiring and Presenting Information and Knowledge in the Web. In: Rachev, B., Smrikarov, A. (Eds.): CompSysTech 2005, Varna, Bulgaria. pp. IIIB.7.1–IIIB.7.6. (2005)
5. Vranić, V., Marko, V.: Developing a Product-Line Based Architecture in a Domain Under Research. In: Návrat, P., Bartoš, P., Bieliková, M., Hluchý, L., Vojtáš, P. (Eds.): Tools for Acquisition, Organisation and Presenting of Information and Knowledge, Proceedings in Informatics and Information Technologies, Research Project Workshop, Bystrá dolina, Nízke Tatry, Slovakia. pp. 211–222. (2006)

RIDAR – Relevant Internet Data Resource Identification*

Zoltán Balogh

Institute of Informatics, Slovak Academy of Sciences, Dubravská cesta 9,
845 07 Bratislava, Slovak Republic
balogh@savba.sk

Abstract. Information acquiring systems often require identifying primary internet resources. RIDAR allows exploiting existing search engines to retrieve links to relevant Internet resources based on users-supplied search terms or more complicated search expressions. Details about identified resources (URL, title, etc.) are stored into databases.

Keywords: Information Resource Identification, Internet Search Services Integration

Introduction

This tool exploits the potential of existing search engines to identify relevant information resources on the Internet based on users-supplied serch terms or more complicated search expressions. Tool can integrate any search engine which exposes a web service API. Currently, RIDAR supports and had integrated the following search engines: Google and Yahoo!

RIDAR provides generic interfaces which allow integrating search engines as well as targets for storing search results (databases).

Implementations

In order to access API of any search engine, one must register to obtain an application ID (or license key). License key must be used each time the API is accessed. License key for Google and application ID for Yahoo was obtain just for the purpose of the NAZOU project. Limitation of such registration is that the number of queries is limited for each license key.

RIDAR also allows storing retrieved results into any target such as database or generic file. Currently MySQL target is implemented in RIDAR.

References

1. Google Web APIs (beta). http://www.google.com/apis/
2. Yahoo! Search Web Services. http://developer.yahoo.net/search/index.htmlAlberto

* This work is supported by projects K-Wf Grid EU RTD IST FP6-511385, NAZOU SPVV 1025/2004, RAPORT APVT-51-024604, VEGA No. 2/6103/6, VEGA 2/7098/27

Knowledge modelling

Towards ontology language handling imperfection

Alan Eckhardt, Peter Vojtáš

Dpt. Software Eng., Fac. Math. Phys., Charles University Prague
Inst. Computer Sci., Czech Acad. Sci, Prague

Abstract. This abstract summarizes research efforts to develop a description logic sufficiently expressive and still effective to enable ontology languages to cope with imperfection (e.g. uncertainty, user preference, vagueness, imprecision, ...). We offer a model based on connection between three description logic systems, classical $\mathcal{EL}$ description logic, f-$\mathcal{EL}^{@}$ description logic with fuzzy aggregation and concepts (crisp roles) and a variant of Bayesian description logic B-$\mathcal{EL}$. We report on some preliminary experiments.

In this abstract we consider the phenomena of imperfection (covering uncertainty, vagueness, user preference, imprecision, ...) in web modeling languages and description logic.

In [6] we have considered querying web resources containing several vague concepts of user's preferences. Typical example is a user looking for a hotel for vacation which is cheap and close to a beach (another application domain was considered in [1]). These particular preferences need to be combined to get an overall ordering of results. In [6] we have proposed f-$\mathcal{EL}^{@}$ description logic allowing existential restrictions, crisp roles, fuzzy concepts and fuzzy aggregation functions. Our system is less expressive than fuzzy description logic of U. Straccia [4] but more effective. Fuzzy roles require to use reification when representing them in RDF and OWL languages.

As an inductive counterpart, in [5] we have investigated the problem of finding dependencies (annotation function, fuzzy aggregation) between global classification of resources (e.g. hotels) and multiple monotone classifications of particular attributes (e.g. price and distance). We have used Bayesian learning and embeddings between different models: GAPgeneralized annotated programs of M. Kifer and V.S. Subrahmanian [3], LP$_{mon}$-classical logic programs with monotonicity, BLP-Bayesian logic program of K. Kersting and L. De Raedt [2] and Bayesian networks BN (or BN$_p$, a special monotonized version), see Fig.1. Specific induc-

[1] Email: peter.vojtas@mff.cuni.cz , supported by Czech IT project 1ET100300517, 1ET100300419 and MSM–0021620838

tive methods (so far without formal model) were tested in [1].

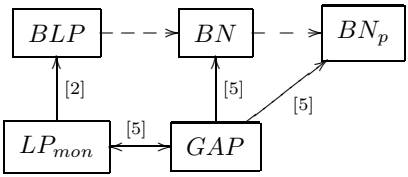


*Fig.*1

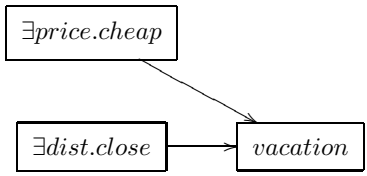


*Fig.*2

Assume, we have a finite ordinal preference scale T, e.g. $T = \{$best, good, acceptable, bad, worst$\}$.

Concept descriptions in B-$\mathcal{EL}$ are either $\top$, atomic concepts, $\exists r.C$ or a complex concept can be defined by a T-Box axiom of the form $C|(C_1, \ldots, C_n)$.

A B-$\mathcal{EL}$ interpretation is a pair $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \bullet^{\mathcal{I}} \rangle$, with nonempty domain $\Delta^{\mathcal{I}}$ and interpretation of language elements $\bullet^{\mathcal{I}}$: $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$, for a instance name (e.g. $hilton \in \Delta^{\mathcal{I}}$), $A^{\mathcal{I}}$ is a probability distribution over $\Delta^{\mathcal{I}} \times T$, for A a concept (understood as a probability of $A(a) = t$, e.g. $cheap(1000) = acceptable$), $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, for r role name (roles correspond to web data and remain crisp - imperfection comes with user concepts, e.g. $price(hilton, 1000)$). For each complex concept (T-Box axiom) of the form $C|(C_1, \ldots, C_n)$, there is a conditional probability density $p^{\mathcal{I}}$ specifying for every $a \in \Delta^{\mathcal{I}}$ and every $t \in T$ and $t_1, \ldots, t_n \in T$ the value $p^{\mathcal{I}}(C(a) = t | C_1(a) = t_1, \ldots, C_n(a) = t_n)$, e.g. of being $vacation(hilton) = good$ based on acceptable price and best distance, see Fig2. $(\exists r.C)^{\mathcal{I}}$ corresponds to BLP combination function max of [2].

Prelimenary experiments were done in [1] (we assume data are certain, our interpretation of these data introduces imperfection based e.g. on user profile and context). Classical, fuzzy and Bayesian $\mathcal{EL}$ description logic (both induction and deduction) can be merged as in [5], see Fig1.

References

1. A. Eckhardt. Metody pro nalezení nejlepší odpovědi s různými uživatelskými preferencemi (in Czech), MSc thesis Charles University, 2006
2. K. Kersting, L. De Raedt. Bayesian Logic Programs, Technical Report 151, University of Freiburg, 52 pages
3. M. Kifer, V. S. Subrahmanian. "Theory of generalized annotated logic programming and its applications", J. Logic Programming, 12 (1992) pp 335–367
4. U. Straccia. A fuzzy description logic for the Semantic web, In FLSW, Elsevier (2006) 73–90
5. P. Vojtáš, M. Vomlelová. "On models of comparison of multiple monotone classifications", In Proc. IPMU'2006, Éditions EDK, Paris, pp. 1236-1243
6. P. Vojtáš. A description logic with combination of user preference concepts, crisp roles and existential restrictions, In EJC 2006, pp. 176-183, 2006

Pulse Coupled Neural Network Models for Dimension Reduction of Classification Space*

Radoslav Forgáč, Igor Mokriš

Institute of Informatics, Slovak Academy of Science, Dúbravská cesta 9, 945 07 Bratislava, workplace Demänová 393, 031 01 Liptovský Mikuláš, Slovakia
forgac@aoslm.sk, mokris@aoslm.sk

Abstract. The paper is aimed at overall survey of Pulse Coupled Neural Networks (PCNNs) for dimension reduction of classification space. The standard PCNN model and its modifications are shown.

Keywords: Pulse Coupled Neural Network, feature generation, dimension reduction.

1 Introduction

Direct image processing in multidimensional image space is very difficult or impossible practically. The image processing would be very complicated and its complexity depends on dimension space of processed images. For that reason image processing is not solved in the input space of processed images with high dimension, in generally. Dimension reduction in our case (Fig. 1) is transformation of input image space to feature space with lower dimension by using feature generation and feature selection

$$f: R^D \rightarrow R^d \tag{1}$$

where $d < D$, what brings the easier image processing and analysis and at the same time the classification process will be easier, too.

It is necessary to fulfilled several criteria in the image recognition process by using features approaches [3]:

- precision of image representation by features,
- relevance and minimization of the number of features, which describe the image,
- invariance of features against geometric transformations and distortions,

* This work was supported by Slovak Science and Technology Assistance Agency under the contract No. APVT-51-024604 and Slovak Science Agency VEGA No. 2/7098/27.

- minimization of computational demands of feature generation and selection algorithms.

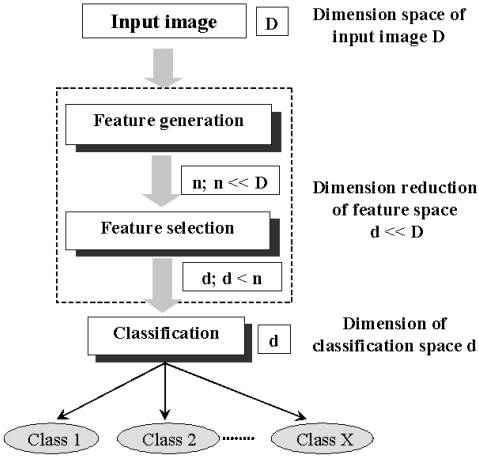


Fig. 1 Image recognition process.

There are available several methods for dimension reduction that create a new formal representation of images [3] especially Pulse Coupled Neural Networks.

2 Standard PCNN

The basic model of PCNN was proposed by Eckhorn to explain the experimentally observed synchronous activity in the mammalian visual cortex [1]. Eckhorn defined the new model with coupling term, synaptic connections as leaky integrators and pulse generator called a neuromime. This model was more described by Johnson [4]. The PCNN has a fixed structure and do not need the learning typical for standard neural networks. Generated features of PCNN are invariant to geometrical transformations.

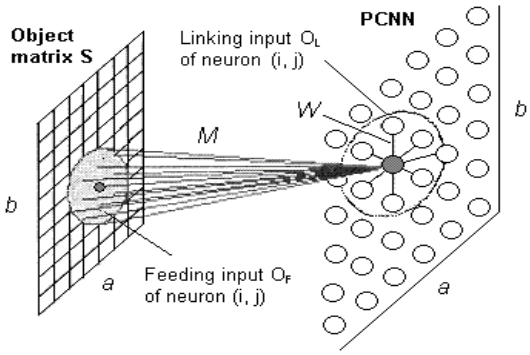


Fig. 2 PCNN structure.

The standard PCNN model is described as iteration by the following equations:

$$F_{ij}(n) = S_{ij} + F_{ij}(n-1) \cdot e^{-\alpha_F} + V_F \cdot (M * Y(n-1))_{ij} \tag{2}$$

$$L_{ij}(n) = L_{ij}(n-1) \cdot e^{-\alpha_L} + V_L \cdot (W * Y(n-1))_{ij} \tag{3}$$

$$U_{ij}(n) = F_{ij}(n) \cdot (1 + \beta \cdot L_{ij}(n)) \tag{4}$$

$$\Theta_{ij}(n) = \Theta_{ij}(n-1) \cdot e^{-\alpha_\Theta} + V_\Theta \cdot Y_{ij}(n-1) \tag{5}$$

$$Y_{ij}(n) = \begin{cases} 1 \text{ if } U_{ij}(n) > \Theta_{ij}(n) \\ 0 \text{ otherwise} \end{cases} \tag{6}$$

where F_{ij} is the feeding input, L_{ij} is the linking input, n is an iteration step, S_{ij} is an intensity of pixel i,j in the input matrix, W and M are the weight matrices, * is the convolution operator, Θ_{ij} is threshold potential, Y is the output of the neuron, V_L and V_F are coefficients of potentials, α_L and α_F are decayed constants.

The main disadvantage of PCNN is difficult mathematical model with high number of parameters and finding their optimal values for feature generation, next high number of iteration steps in the feature generation process and problem of feature selection with the highest information value. These problems were very often solved through an experiment. The much better solution of this problem is to modify the PCNN model [2, 5, 7].

3 Modified models of standard PCNN

From the set of modified models of standard PCNN was applied especially PCNN with modified feeding input (M-PCNN) [6], Intersecting Cortical Model (ICM) [5] and Optimized M-PCNN (OM-PCNN) [2].

The M-PCNN does not include the exponential and convolutional item for feeding input in equation (2). The advantage of this version is the elimination of time for

features generation and parameter reduction. The efficiency of recognition remains comparable with standard PCNN. The M-PCNN was applied to object detection, noise filtering and image segmentation.

The ICM is the PCNN modification focused on object detection especially. The ICM model is much easier in relation with standard PCNN algorithm and may be described by terms

$$K_{ij}(n) = (W * Y(n-1))_{ij} = \sum w_{ijkl}.Y_{kl}(n-1) \tag{7}$$

$$U_{ij}(n) = S_{ij} + t_u \cdot U_{ij}(n-1) + K_{ij}(n) \tag{8}$$

$$Y_{ij}(n) = \begin{cases} 1 \text{ if } U_{ij}(n) > T_{ij}(n-1) \\ 0 \text{ otherwise} \end{cases} \tag{9}$$

$$T_{ij}(n) = t_t \cdot T_{ij}(n-1) + V_t \cdot Y_{ij}(n) \tag{10}$$

where n is iteration step, K_{ij} the element of convolution matrix K, * is convolution operator, W is the matrix of weight coefficients w_{ijkl}, which determines interconnection of neuron (i, j) and its neighbour neurons (k, l). S_{ij} is the intensity of given neuron (i, j). U_{ij} is the activation quantity of neuron, Y_{ij} is the output quantity of neuron, T_{ij} is threshold potential of neuron, parametres t_u a t_t are decay coefficients and parameter V_T is coefficient of threshold potential.

The main goal of OM-PCNN was to reduce the number of generated features to reach high image recognition performance. The optimization was based on the M-PCNN, where the following disadvantages were eliminated:

1. high number of parameters and problems with their optimization,
2. optimal number of iteration steps determination,
3. the most significant feature loses its information value.

The comparison analysis for OM-PCNN and M-PCNN confirmed higher successfulness of image recognition by O-PCNN with lower number of generated features [2].

4 Summary

The PCNN research was focused on the PCNN optimization with aim to reduce the number of generated features to reach high image recognition performance. The analysis and comparison of the various PCNN models shows that the OM-PCNN is the most suitable for the dimension reduction of classification space and invariant image recognition.

References

1. Eckhorn, R. et al.: Feature Linking via Synchronization among Distributed Assemblies: Simulations of Results from Cat Visual Cortex, Neural Computation, Vol. 2, 1990, pp. 293-307.
2. Forgáč, R.: Dimension Reduction of Image Classification Space by Pulse Coupled Neural Networks. [PhD thesis]. FEI TU Košice, 2005, (in Slovak).
3. Forgáč, R., Mokriš, I.: Artificial Neural Networks on Dimension Reduction of Feature Space and Classification. [Scientific monograph]. UMB Banská Bystrica (2002), ISBN 80-8055-743-8 (in Slovak).
4. Johnson, L. J.: Pulse Coupled Neural Nets: Translation, Rotation, Scale, Distortion and Intensity Signal Invariance for Images, Applied Optics, Vol. 33, No. 26, 1994, pp. 6239-6253.
5. Kinser, J. M.: A Simplified Pulse-Coupled Neural Network. In: S.K. Rogers, D.W. Ruck (Eds.), Applications and Science of Artificial Neural Networks II, Proceedings of SPIE, Vol. 2760, No. 3, 1996, pp. 563-567.
6. Ranganath, H. S. - Kuntimad, G.: Image Segmentation Using Pulse Coupled Neural Networks. In: Proceedings of IEEE World Congress on Computational Intelligence, Orlando, 1994, pp. 1285-1290.

Modeling of knowledge creation processes based on Activity theory

František Babič, Jozef Wagner

Centre for Information Technology, Faculty of Electrical Engineering and Informatics
Technical University in Košice, 042 01 Košice, Slovakia
{Frantisek.Babic, Jozef.Wagner}@tuke.sk

Abstract. The general challenges for modeling of the knowledge creation processes are the following: all participants should get an up-to-date understanding of the process, they should be able to make their individual and interconnected changes of the process-related information and they should have the possibility to reflect on the course of process and practices of working together. Activity theory (AT) is a powerful and clarifying descriptive theory focusing on understanding of human activity and work practices. Human activity can be represented as a hierarchy of concepts with three levels: activities (first level of concepts) realized through chains of actions (second level), which are carried out through operations (third level). Knowledge creation processes can be modeled using AT by means of the process ontology.

Keywords: Knowledge creation processes, activity theory, ontology

1 Knowledge processes

Knowledge Process (KP) is defined as a set of activities conducted during learning or work, e.g. activities conducted for a specific purpose, or a set of ordered steps across time intended to reach a goal or to produce a specific outcome.

Knowledge creation is one of the core aspects of learning and of the knowledge development in general (this includes also knowledge adoption, distribution, review and revision) within an organization. From the methodological point of view the knowledge creation processes have been studied in different contexts [1]:

- Carl Breiter's knowledge building approach
- Nonaka and Takeuchi's model of organizational knowledge creation
- Yrjö Engeström's theory of expansive learning based on Activity Theory.

2 Activity theory

Activity theory (AT) is a powerful and clarifying descriptive theory focusing on understanding of human activity and work practices. It incorporates notions of intentionality, history, mediation, collaboration and development [2].

AT originated in the former Soviet Union in the 1920's and 1930's as a part of the cultural-historical school of psychology founded by Vygotsky[4]. The theory provides a powerful framework for describing and analyzing collaborative processes.

Activity systems are never static but evolve when contradictions emerge between the elements within the activity system.

The fundamental unit of AT is the human activity that can be described as a hierarchy of concepts with three levels [3]: activities (first level concepts) are realized through chains of actions (second level concepts), which are carried out through operations (third level concepts). Human activity is always directed toward a material or ideal object satisfying a need and the subject s reflection of, and expectation to, this object characterizes the motive of the activity.

3 Knowledge processes tool

One of the main tools in the KP-Lab[1] project is the so-called KP-Lab Shared Space (ShSp) that will support the knowledge creation and innovative practices. Part of the ShSp is the Knowledge processes tool (KPT) that provides a set of functions and interfaces necessary for creation, management, and annotation of knowledge processes composed from various elements.

KPT is based on the process ontology (PO) as scaffolding for modeling of the knowledge processes. The development of PO was complex and relatively long process, which resulted in the actually used version (see Fig.1)

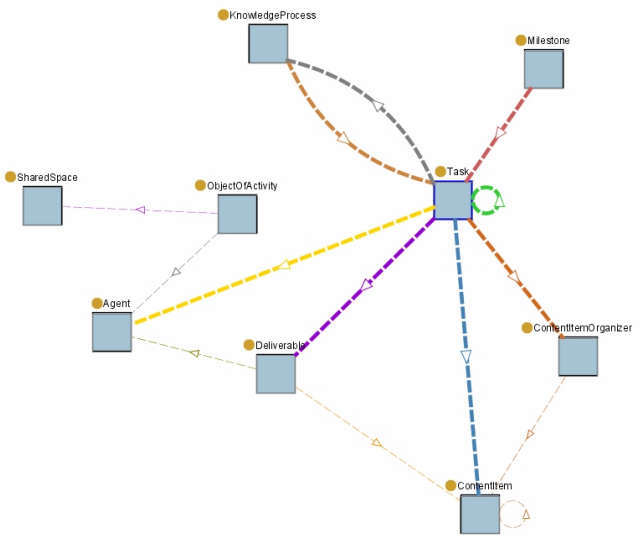


Fig. 1. Final version of the process ontology

[1] www.kp-lab.org

The process ontology is a part of the KP-Lab system model and this is stored within RDFSuite [5] that is used as the knowledge repository.

Structure of the full process is visualized in form of the Gantt chart (see Fig.2), so called Process view that is one of the possible ways how to visualize a Shared Space (see Fig.3).

Fig. 2. Process view (The user interface of KPT)

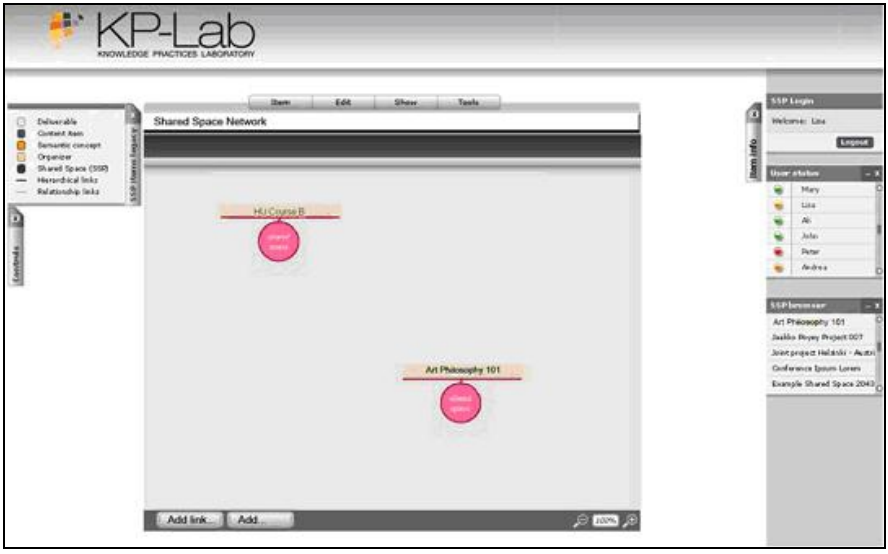


Fig.3. The user interface of the KP-Lab Shared space

Acknowledgments

The KP-Lab Integrated Project is sponsored under the 6th EU Framework Programme for Research and Development. The authors are solely responsible for the content of this article. It does not represent the opinion of the KP-Lab consortium or the European Community, and the European Community is not responsible for any use that might be made of data appearing therein.

References

1. Paavola, S. & Hakkarainen, K:"Trialogical" processes of mediation through conceptual artifacts. A paper at the Scandinavian Summer Cruise at the Baltic Sea, 2004.
2. Nardi, B. A.: 'Activity Theory and Human-Computer Interaction' in Context and Consciousness: Activity Theory and Human Computer Interaction. MIT Press, Cambridge, 1996.
3. Bardram, J. E.: Plans as situated action: An activity theory approach to workflow systems. Proceedings of ECSCW 97**,** Dordrecht, the Netherlands: Kluwer Academic Publishers, 1997.
4. Vygotsky, L.S.: Mind in society: The development of higher psychological processes. Harvard University Press, Cambridge, 1978.
5. Alexaki, V., Christophides, G., Karvounarakis, D., Plexousakis, K., Tolle: The ICS-FORTH RDFSuite: Managing Voluminous RDF Description Bases, In Proc. of the 2nd International Workshop on the Semantic Web (SemWeb'01), in conjunction with Tenth International World Wide Web Conference (WWW10). Hongkong, (2001) 1-13

Constructing multi-theories expert system for UML models validation

/ extended abstract /
Miroslav Líška

Slovak University of Technology
Faculty of Informatics and Information Technologies
Ilkovičova 3, 842 16 Bratislava, Slovakia

mliska@formal-analysis.com

Abstract. The main goal of this article is to present technique for constructing expert system for UML models validation and also to present prototype named Formal Studio. Expert system understand problematic of business process domain and also understand UML modeling. Aim of this solution is to use it in software development process as validation tool and also as intelligent knowledge base based on expert system. It could be great advantage for any software company when it could be able storing its evolving experiences from many different projects into organization knowledge for reuse. Later extension of this software could be targeted into intelligent agent, which could schedule, explain or consult the best cost-efficient software solution for customers with theirs special requirements.

Keywords: expert system, artificial intelligence, software engineering, business process theory, UML theory, formal analysis, formal validation

1 Introduction

Language UML (Unified Modeling Language) brought new generation into specification, realization and documentation into software development process. The main visible facility which supports development process is separating specification from realization. It brought new possibilities for innovations in software disciplines as analysis, design, implementation and others software disciplines. Analysts can do more work above domain specification and designers can utilize more advantages that offer technology in realization. But only simple usability of UML is not immediately warranty of successful project.

2 Identification and problem analysis

When company delivers large and complex software system for its customers, project members usually becomes experts in area of software implementation. They are able to communicate with customer in the proper level of experiences, what is the necessary condition for supporting customer evolution. But this fact also shows how company depends on its experts. It is common that employees changing theirs employers (what is natural fact), but this results into evolution slowdown for software house and also for customer. This problem belongs into more general problem named knowledge instability [1].

Another natural problem in developments process is project complexity. It is important to create proper business process analysis, usecase model, and many realization models. The ideal way is to have optimized transformations between these models, and also have efficient refinement in development process itself, based on previous project experiences. Other problem is to share, evolve and also store this knowledge in organization. Moreover, there is also problem with effective usability of all knowledge in organization, how it can be used in actual project. This can be partially done through model reviewing with other project members or models validating with appropriate UML CASE tool (i.e. against OCL constraints). But these solutions cannot address all problems effectively in optimized, knowledge based software development process. The other possible solution, which has ability to exact addressing for these problems, could be achieved through creating expert system for software development process.

3 Expert system in model driven development

Expert system in software organization has opportunity to solve previous defined problems. It is important to formalize different experiences obtained from different projects, for use in future projects or also for refining actual methodology of development process. This article is targeted into model driven development based on UML, therefore expert system will be also targeted into this domain. Question is how expert system could help organization to do better projects and also how it could it eliminate knowledge instability and also knowledge complexity?

Primary thing is that expert system must understand UML domain, business process domain and other relevant domains. This can be done, when these domains are represented as mathematical theories in expert system. Nowadays, there are many formalization techniques of language UML [2, 3], because it is important to remove many lacks of weak definition of UML semantics. Above these concepts, an expert system could be created with included validation process. This validation process is based on proving, that UML model is mathematical model of these theories (i.e., language realization) [4, 5]. At first, expert system must hold these theories (knowledge) in some appropriate form. Suitable is Skolem form, where formula is transformed into clauses conjunction, where this form is also essential for automatical

proving. Moreover, this formalization can be targeted into different theories based on predicate logic, what brings opportunity to study and evolve different theories separately.

4 The prototype

In present time, I work on such expert system named Formal Studio. Expert system holds theories of business domain and also of UML and is able to validate UML model against these theories. But this software is actually rather prototype than useful tool for development process.

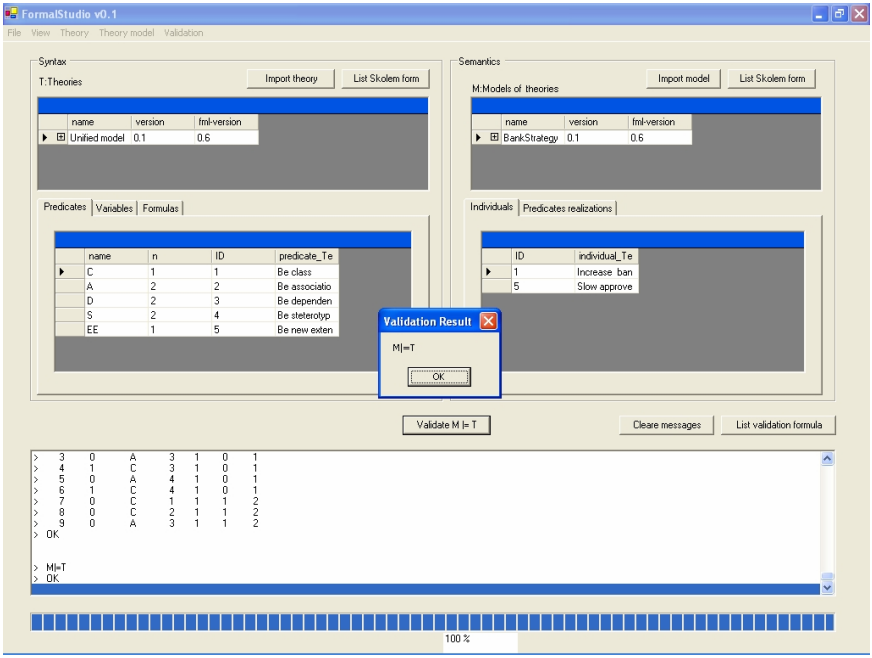


Fig. 3. Example of Formal Studio prototype. On left side is syntax part (theories which consist of variables, predicates, and formulas) and on the left side is semantics (i.e., imported and translated UML model).

On fig. 3 is depicted screenshot after simple UML diagram has been successfully proved against UML theory.

5 Conclusion and future work

Future work will be oriented into improvement of mathematical proving strategy used in created expert system and primarily, for use of this solution into real development process [6, 7]. Expert system in this paper shows only validation functionality without e.g. scheduling, explaining or other parts of standard expert system functionality which must be specified and implemented. Future work will also be oriented into improving architecture of expert system with orientation into agent system. This system e.g. could moreover consult cost-efficient software system solution, or it could be extended with modal logic in validating process.

Acknowledgments. "This work was partially supported by the Scientific Grant Agency of Slovak Republic, grant No. VG1/3102/06."

References

1. Návrat, P. et al.: Artificial Intelligence, Slovak University of Technology , Bratislava, ISBN 80-227-1645-6, (2002)
2. Amálio, N., Stephney, S., Polack, F.: Formal proof from UML model, ICFEM'04, Seattle, USA, 2004. LNCS 3308, pp 418-433. Springer, 2004
3. Evans, A., France, R., Lano, K., Rumpe, B.: The UML as formal modeling notation, Computer Standarts & Interfaces, (1998), 19.
4. Cmorej, P.: An Introduction into logical syntax and semantics, (second edition, Iris, Bratislava 1998)
5. Peregrin, J.: An Introduction into theoretical semantics, Masaryk University of Brno, (1994).
6. Líška, M.: Extensional and intensional semantics of PUML objects, IIT.SRC (2005), 167.
7. Líška, M.: Formal Unified Process, IIT.SRC (2006)

Authors Index

Michal Laclavík, Ivana Budinská, Ladislav Hluchý
Editors

Proceedings

1st Workshop on Intelligent and Knowledge Oriented Technology

1st edition, 100 copies, Published by Institute of Informatics SAS

Printed by Equilibria s.r.o. in Košice

2007

ISBN 978-80-969202-5-9
EAN 9788096920259