3rd edition

dsie '08

7 | 8 feb

FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

PROCEEDINGS OF DSIE '08

Doctoral Symposium on Informatics Engineering

FEUP edições

EDITORS
António Augusto de Sousa
Eugénio da Costa Oliveira

Colecção Colectâneas . ***18***

3rd edition

dsie '08

7 | 8 feb

FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

PROCEEDINGS OF DSIE '08

Doctoral Symposium on Informatics Engineering

FEUP edições

EDITORS
António Augusto de Sousa
Eugénio da Costa Oliveira

Editores/Editors : António Augusto de Sousa e Eugénio da Costa Oliveira
Adaptação Gráfica/Design Adaptation . uTICM (FEUP)
Impressão e acabamentos/Printing: MULTIPONTO - Rafael Valente & Mota, S. A.
1ª Edição/1st Edition . 2008
Depósito Legal nº 276421/08
ISBN: 978-972-752-099-2

A qualidade de reprodução das imagens esteve dependente dos ficheiros fornecidos pelos autores./Image quality is related to the files sent by the authors.

ORACLE

phc Software

sage
Software para uma gestão eficaz.

Autodesk

Grupo AutoSueco

INOVA
INOVA+ | INOVAMAIS, S.A.

agilus I+D

FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

U.PORTO

DSIE 08 SECRETARIAT
Faculdade de Engenharia da Universidade do Porto
Rua Dr. Roberto Frias, s/n
4200-465 Porto - Portugal
Tel: +351 225 081 502
Fax: +351 225 081 443
Website: *www.fe.up.pt/dsie08*
Programme Committee: *dsie08-pc@fe.up.pt*
Organizing Committee: *dsie08-oc@fe.up.pt*

FOREWORD

DSIE'08 - Doctoral Symposium in Informatics Engineering 2008, follows the two previous editions of CoMIC (Scientific Research Methodologies Conference) and aims to be a forum for the discussion and application of good practices of scientific research, namely in what Computer Science and Informatics Engineering is concerned. DSIE'08 is organized in the context of the Doctoral Program on Informatics Engineering (ProDEI) at the Faculty of Engineering of the University of Porto.

DSIE'08 implicitly displays what the PhD students have learnt during their first semester course on Scientific Research Methodologies (MIC), including appropriate methods to deal with their own research, as well as student's capabilities to produce well written scientific texts. This symposium is also a good opportunity for students to be jointly involved in all the aspects of a scientific meeting organization and participation, although lightly supervised by the course's responsible professors.

Being mainly devoted to the students of the Doctoral Program in Informatics Engineering current edition, DSIE'08 also accepted submissions from former PhD students of the same program and from PhD students of different, though related, programs.

Current contributions already reveal that many of the students have interesting ideas (although in an embryonic state) about what specific research topic they wish to tackle. Also, papers can be found where the state of the art description still is the major paper's contribution. Once again, participants still are in the beginning of their own scientific work, so this can be perfectly understood and accommodated in the symposium program.

This DSIE'08 Proceedings volume includes fifteen papers that have been revised and selected according to guidelines informed by the aforementioned principles and are assembled according to the different research topics that will be presented in six different technical sessions: Computer Graphics (3 papers), Information Acquisition and Retrieval (2 papers), Artificial Intelligence (2 papers), Information Systems (3 papers), Programming Paradigms and Techniques (3 papers) and E-Learning (2 papers).

On the top of those sessions, distributed among a two-day duration time, DSIE'08 also encompasses two invited talks and a technical session in charge of some of the sponsors.

We, the professors responsible for the MIC course, would like to acknowledge all those who were deeply involved in the success of this event that, we hope, will contribute for a better understanding of both the themes that have been addressed during the course, the best scientific research methods and the good practices for writing scientific papers.

Eugénio Oliveira and A. Augusto Sousa

(in charge of the MIC course - Scientific Research Methodologies)

CONFERENCE COMMITTEES

Programme Committee Co-Chairs
Carla Pereira (ProDEI 2007)
João Vila Verde (ProDEI 2007)

Senior Programme Committee
A. Augusto Sousa
A. Lucas Soares
Ademar Aguiar
Ana Paiva
Ana Paula Rocha
António Castro
António Coelho
António Pereira
Eugénio Oliveira
Francisco Vasques
Gabriel David
Henrique L. Cardoso
J. Canas Ferreira
J. Correia Lopes
J. Falcão e Cunha
J. Pascoal Faria
Jaime Villate
Jorge Alves da Silva
Jorge Barbosa
Luis Paulo Reis
Mario de Sousa
M. Velhote Correia
Miguel Monteiro
Nuno Escudeiro
Nuno Flores
Pedro Souto
Raul Vidal
Rosaldo Rossetti
Rui Camacho
Sérgio Nunes

Programme Committee
Alexandre Azevedo (ProDEI 2007)
Carla Lopes (ProDEI 2007)
Fernando Teodósio(ProDEI 2007)
Filipe Correia (ProDEI 2007)
João Barbosa (ProDEI 2007)
Jorge Esparteiro Garcia (ProDEI 2007)
Jorge Mota (ProDEI 2007)
José Carlos Miranda (ProDEI 2007)
José Joaquim Moreira (ProDEI 2007)
Paulo Alves (ProDEI 2007)
Pedro Valente(ProDEI 2007)
Sérgio Barbosa (ProDEI 2007)
Válter Rocha (ProDEI 2007)

Organising Co-Chairs
Filipe Correia (ProDEI 2007)
Jorge Esparteiro Garcia (ProDEI 2007)

Organising Committee
Alexandre Azevedo (ProDEI 2007)
Carla Lopes (ProDEI 2007)
Fernando Barbosa (ProDEI 2007)
Fernando Teodósio (ProDEI 2007)
Jorge Mota (ProDEI 2007)
José Miranda (ProDEI 2007)
José Moreira (ProDEI 2007)
Paulo Alves (ProDEI 2007)
Pedro Valente (ProDEI 2007)
Válter Rocha (ProDEI 2007)

CONTENTS

1. Technical Programme

Invited Speaker - Luís Sarmento

. Sonhos, dúvidas existenciais e manhas de um aluno de doutoramento

Session 1 - Computer graphics

. X3D and Java Fusion in a Medieval Fantasy Game 15
. Métodos de Iluminação Global Baseados em Fotões 27

Session 2 Information aquisition and retrieval

. Sensing the World: Challenges on WSNs 39
. Evaluation and Comparison of Automatic Methods to Identify Health Queries 52

Session 3 - Artificial inteligence

. Calibration Agent for Ecological Simulation: A Metaheuristic approach 64
. A System for Automatic Construction of Exam Timetable Using Genetic Algorithms 76

Invited Speaker - Luís Amaral

. Necessidade de novos indicadores da construção da sociedade da informação

Session 4 - Information systems

. Collaborative Ontology Specification 88
. Trends on Adaptive Object Models Research 100
. An Approach to Improve Speed and Objectivity in Audits 112

Session 5 - Programming techniques and paradigms

. Aspect-Oriented Web Development in PHP 123
. Comparing Three Aspect Mining Techniques 134
. A New Grouping Method for XSL 1.0 146

Session 6 - E-learning

. Using Learning Styles and Neural Networks as an Approach to eLearning Content and Layout Adaptation 158
. Competency Management Articulation for Human Resources 170

2. Authors in Alphabetical Order 181

3. Papers in Alphabetical Order 183

1. TECHNICAL PROGRAMME

X3D and Java Fusion in a Medieval Fantasy Game

José Carlos Miranda[1,2], Nuno Martins[2]

[1] Faculdade de Engenharia da Universidade do Porto, Portugal
[2] ESTG, Instituto Politécnico da Guarda, Portugal
jcmira@ipg.pt, martins@sal.ipg.pt

Abstract. The presentation of 3D contents on the Internet is very interesting and presents many potentialities, especially if there is a great amount of interactivity within the applications. The X3D is a good format to use in the creation and visualisation of 3D contents, but the interactivity, which is possible to develop without using an external programming language, is limited. A prototype of medieval fantasy game has been developed, trying to demonstrate the integration capacities of the X3D format and the Java programming language. The integration of these two technologies was possible through the programming interface SAI (Scene Access Interface) that showed a good outcome, allowing a considerable increase of the power of interactivity necessary for this type of applications.

Keywords: Virtual Worlds, 3D modelling, X3D, Java, Xj3D, RPG game, Human-Animation.

1 Introduction

The 3D contents visualization on Internet has great potentialities in different areas like architecture, medicine, education, entertainment, amongst others. One of the most important standards that emerged in 1995 and that allowed the three-dimensional world visualization and interaction on the Internet was VRML – Virtual Reality Modeling Language [1], based on the Open Inventor technology of Silicon Graphics. The last version of this format is from 1997. Since then this language naturally evolved to another one, called X3D – eXtensible 3D. X3D is the new standard for 3D content distribution on Internet and its specification was approved by ISO in December 2004, and since then many revisions have been taken place [2].

It is important to refer to the great need for interactivity within this type of presentation. X3D is appropriate for the creation of 3D content visualization, but the interactivity that is possible to create without using another external programming language is limited. Due to the huge quantity and diversity of objects that exist within X3D, there are many potentialities to explore when using an external programming language. An adequate programming environment for this manipulation is Java [3], [4].

To assess the potentialities of X3D and the external programming language Java integration, an application has been developed for the specific area of entertainment.

A Role Playing Game (RPG) was developed once this application requires a big amount of interactivity between the player and the virtual world. The appropriate programming interface to establish the connection between these two technologies is called Scene Access Interface (SAI) which allows the exchange of events between the X3D scene and the Java application, producing much more powerful interactions.

To overcome the lack of information on this recent study-area, a Website has been built, where some examples of this integration can be found. This site [5] is recommended to users that already have some knowledge of X3D or VRML, since the main objective is to explain the integration between X3D and the Java language.

In section 2 some subjects related with the virtual world of the developed prototype are presented. The programming interface that allows the communication between the X3D and the Java language and the considered philosophy related to the game programming are described in section 3. Section 4 represents the results and section 5 includes the conclusions and a few orientative lines towards future work.

2 Virtual World Creation

In this section there are references to some theories about the creation of the virtual world. The first task to carry out is to put the plans on paper, in a textual format or by drawings. These drawings are going to be a kind of storyboard and the starting point for the creation of the virtual world.

2.1 Planning and Interaction Strategies

Some aspects that the creator should think of, are the tasks that the player will be able to carry out in that world. A virtual world where the player simply can walk will not be necessarily interesting. The basic idea underlying the creation of tasks that players will carry out, is that once they are finished the player will receive some kind of reward. This was taken into account in the developed game creating missions that the player should carry out and that lead to some rewards.

According to Roehl [6] it is also important that each scenario of the virtual world has different characters that the user interacts with, because the human being is social by nature. These new characters should have some kind of "artificial intelligence" creating a more realistic world. In the world built for the game there are some places that the user will be able to explore, as well as characters with which he can interact and communicate (Fig. 1).

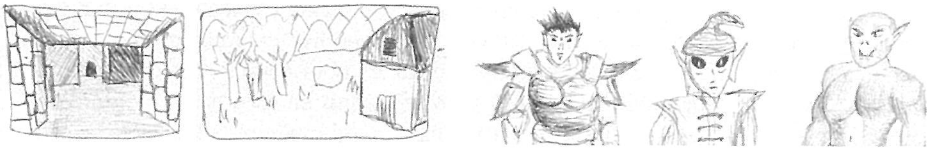


Fig. 1. Drawings of different scenes and some characteres of the game

Human beings like to discover and learn things. The user will feel a huge satisfaction upon discovering the "tricks" necessary to reach an objective like opening a door, finding several places that looked inaccessible, solving puzzles, especially if the user is rewarded after each discovery. A way to maintain a user "prisoner" to a virtual world is by rewarding him in different phases of the game, according to the task that he is doing. According to Shneiderman [7] a way to captivate the users of a virtual world is through the attribution of scores. In the implemented game, these scores were replaced by the characteristics of the character, whose value will increase during the game while the player is completing the missions.

Another characteristic of the real world that equally is important in a virtual world is the danger sensation. Many activities in the real world become more interesting due to the fact of the existence of danger, for example to ride in the russian mountains, doing bungee jumping, or climbing mountains. Fig. 2 shows some examples of weapons that had been imagined for the battles of the present game. In these battles there is the possibility that the player loses life points, or even dies. This sensation of danger is doubtlessly very important in a virtual world specially to turn it realistic and less monotonous, plus the fact that the human being likes to run risks and defeat adversaries.

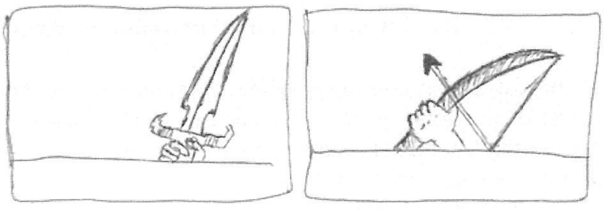


Fig. 2. Drawings of several weapons

2.2 Scenes and 3D Objects

After planning, the next task was the creation of the scenes and the 3D objects. A modelling tool should be used which controls the geometry at the vertices, edges and polygons levels (Fig. 3) not simply using the basic geometrical primitives: cube, cone, sphere, cylinder, etc.

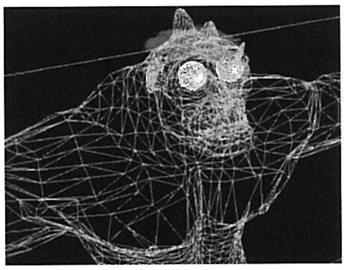


Fig. 3. Polygons of a X3D character

For the creation of the objects that compose the virtual world two modelling tools were used: the Cosmo Worlds and the Vizx3D.

The Cosmo Worlds can be defined as a world builder [8], having been used in the creation of some objects and respective scene positioning, because it includes precision tools that allow a huge amount of control. This is a VRML world builder software, that implies a subsequent conversion to the X3D format. On the other side, the Vizx3D [9] is a specific X3D world builder and, in our opinion, perhaps, the strongest software for the X3D model creation at the moment this project was developed, for it includes enough powerful modelling tools and it was used for the construction of more complex geometries. It was also within this software that the materials and the textures were attributed to the objects, as well as the creation of all animations. It was also in Vizx3D that the addition of environments and some simple interactivity were created.

Previously modelled objects in 3D Studio MAX were also used and later converted to the X3D format, using conversion tools, like the AccuTrans 3D [10].

In the creation of a virtual world for the internet, the optimization process of a 3D scene is very important, because it will allow for bigger loading speeds of the X3D files to the browser, as well as improve it's navigation performance by increasing the rendering speed of the 3D scene.

In the construction of the virtual world several optimization techniques were used, like:

1. Using the less ammount of polygons as possible, eliminating those that will always be invisible, despite of the point of view. According to Hartman [11] the planned worlds to be seen on the Internet should not have more than, approximately, 1000 visible triangles at every single moment.
2. Using materials and textures to create a higher complexity of surfaces effects, without using too many polygons. According to Ballreich [12], a texture should have the smallest possible size and should be squared.
3. Reusing textures and sounds whenever it is possible, with reduced sizes.
4. Limiting the use of lights to maintain the rendering speed at reasonable levels.
5. Structuring prudently the hierarchy of the objects that compose the scene, permitting the browsers to perform the visibility calculation in a more efficient way.
6. Applying characteristics of computer graphics that help to increase the efficiency, for example, different levels of detail (LOD), Billboards, Inlines and Collisions Detection.

These techniques are described, in detail, by Miranda [13].

3 X3D and Java Fusion

While X3D is an adequate language for the creation of three-dimensional objects, the Java language can be used to add complex behaviours to those objects. The integration of these two technologies is possible through SAI - Scene Access Interface [14]. This programming interface has some functions that permit the manipulation of the X3D scene through the utilization of the Java programming language.

When this work was developed, the only browser that allowed the viewing of X3D scenes manipulated through Java is Xj3D [15], a stand-alone browser entirely developed by the Web3D Consortium and that has been used as the only tests platform that integrates X3D with Java [16], [17].

In the following section the integration of X3D with Java will be explained, using the SAI programming interface.

3.1 SAI – Scene Access Interface

SAI is the programming interface (API) used to establish the communication between the X3D and Java. Through this connection it was possible to exchange events between the X3D nodes and the Java application.

The graphic interface components of the developed game in this project are the X3D browser and the Java application. While the browser allows the user to navigate in a three-dimensional environment, the 2D interface created in Java offers some tools that enable the interaction and manipulation of the objects of that 3D environment. In this way, pressing a button will send an event for the X3D scene, changing the characteristics of an object.

SAI allows two kinds of access to the 3D scene, one named by internal access and another one by external access. Both cases can be used together, using the advantages of each one of these approaches.

In the internal access, there is a script node inside the X3D file that will make the connection with the Java class. So, it is the X3D scene that, after being loaded in the X3D browser, will call the Java class where the code exists that will perform the intended alterations in the 3D scene (Fig. 4).

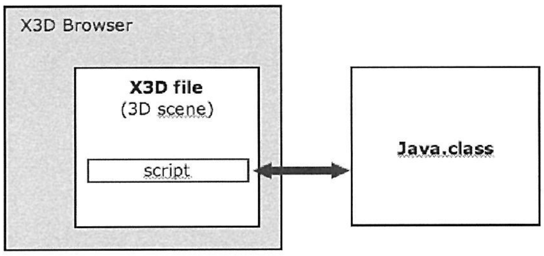


Fig. 4. Internal access

Within the external access, the Java class will create a 3D window and, after that, will load the X3D scene to the previously created 3D window (Fig. 5). In this type of access it is the Java application that will load the X3D file. This becomes very useful when an application with two windows is needed, one composed by the 3D scene and another one by the controlling interface, like the developed prototype in this work.

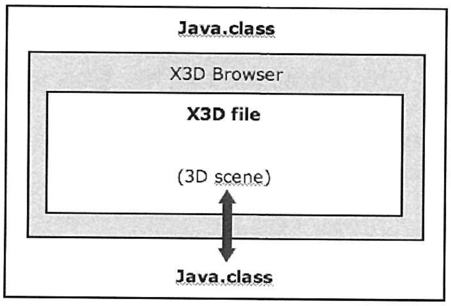


Fig. 5. External access.

In the following section the considered philosophy related to the game programming and some used methodologies will be explained.

3.2 Game Programming

Whenever the interaction and manipulation of objects on the X3D scene through the Java programming language are necessary, it will be necessary to access to those objects properties. After this access to those properties it is possible to proceed to its manipulation. To better clarify this integration process, one case of the implemented game – the battles of the player with the existing monsters – will be explained.

To simulate the monster's field of view, a proximity sensor was placed on the area where the monster will be found (Fig. 6).

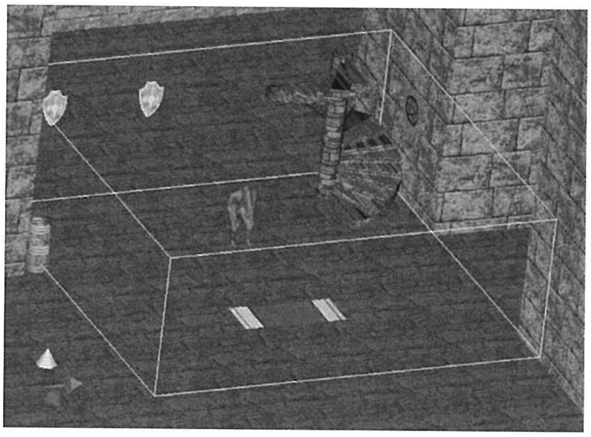


Fig. 6. Proximity sensor in the area where is found the monster

Whenever the player enters the proximity sensor area, an event is triggered. The Java application is automatically notified of that event occurrence, setting the necessary actions, which in that case, would be the beginning of the monster attack.

Movements of both legs and arms of the monster, as well as the movements of rotation and translation in the direction of the player make the produced animation.

To control the monster translation movements towards the player, the basic principle of animation was used. A position interpolator and a clock were applied. The position interpolator has uniquely two keyFrames: an inicial keyFrame, made by the initial coordinates of the monster and the final keyFrame, that will always be updated with the actual coordinates of the player while he is moving (Fig. 7).

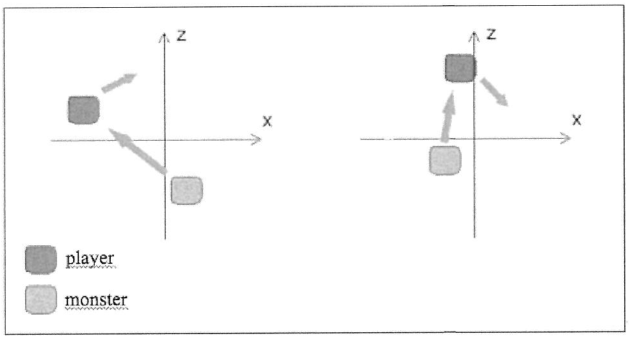


Fig. 7. Monster translation movements

To control the rotation movements of the monster in order to be face to face with the player all the time, the basic principles of analytic geometry were used. The schematic representation is showed in Fig. 8.

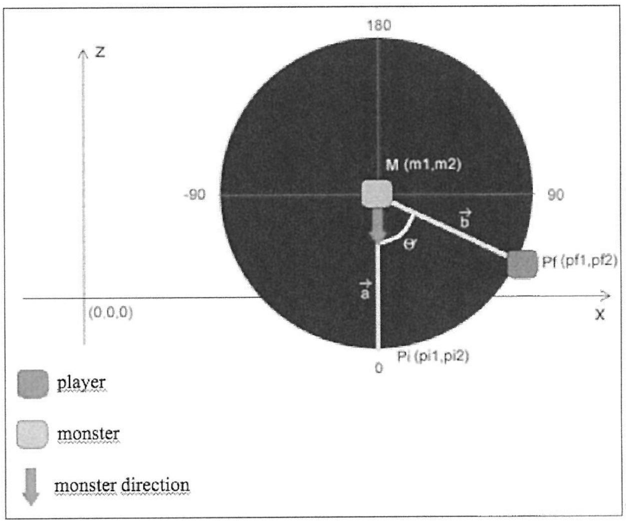


Fig. 8. Schematic representation of monster rotation movements

Initially, the monster is turned into a specific direction, represented by vector $\vec{a}$. The moment the player enters the field of view of the monster is represented by position Pf. The monster is in position M and will have to have a rotation θ to face the player. It will be necessary to calculate the angle between vectors $\vec{a}$ and $\vec{b}$ in order to know the rotation angle of the monster. The presented formulae in equation 1 was used to determine the inner product of vectors $\vec{a}$ and $\vec{b}$.

$$\theta = \arccos \frac{\vec{a} \times \vec{b}}{\left\|\vec{a}\right\| \times \left\|\vec{b}\right\|} \qquad \textbf{(1)}$$

$\left\|\vec{a}\right\|$ is the norm of vector $\vec{a}$, $\left\|\vec{b}\right\|$ is the norm of vector $\vec{b}$ and θ is the angle between the two vectors.

The calculation of the angle θ will have to be continuously performed and applied to the coordinate system of the monster, ensuring that the monster will always be facing the player.

Next it becomes necessary to control when the player is near the monster, the moment when the monster will start his attack, causing damage. To control this situation, a proximity sensor is placed near the body of the monster. This proximity sensor wraps the monster and is placed inside the group that is part of the animation of that monster, so that when the monster is moving, he transports the proximity sensor with himself (Fig. 9).

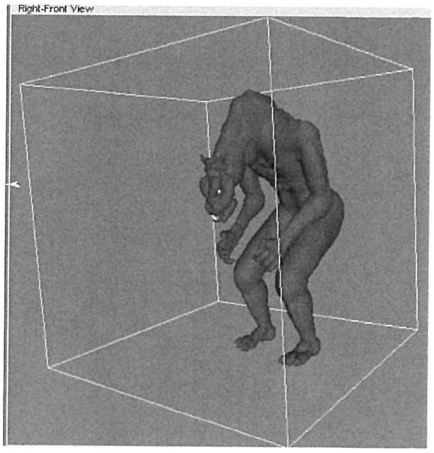


Fig. 9. Proximity sensor of the monster

From the moment that the player enters the defined region by this second proximity sensor, another event will be triggered indicating the program that the attacks can begin. This way, the attack animations of the both monster and the player will start. The damage caused by these attacks is controlled by specific formulae that are based in the physical characteristics of the player, the characteristics of the equipped weapon and the monster characteristics.

A battle finishes when the points of life of the player or of the monster get to zero, causing dead. This is the way through which the battles and both the monster -and player damage are controlled.

After examining this example it is possible to better understand the philosophy underlying this game programming. The created world is made up by a large number of sensors, for example, proximity sensors, touch sensors, visibility sensors, and others. This way, each sensor will notify the player's interactions to the Java application that will trigger an action in the 3D world specific object. Several programming code examples of this game can be found in the Website [5] developed in the field of this project.

4. Results

To assess the potencial of X3D technology and the Java programming language integration, a medieval fantasy RPG prototype has been developed. The graphic interface is composed by two components: the X3D browser and the Java application (Fig. 10). While the browser allows the user to navigate in a three-dimensional environment, the interface created by Java offers several tools that enable the interaction and manipulation of the specific 3D environment objects.

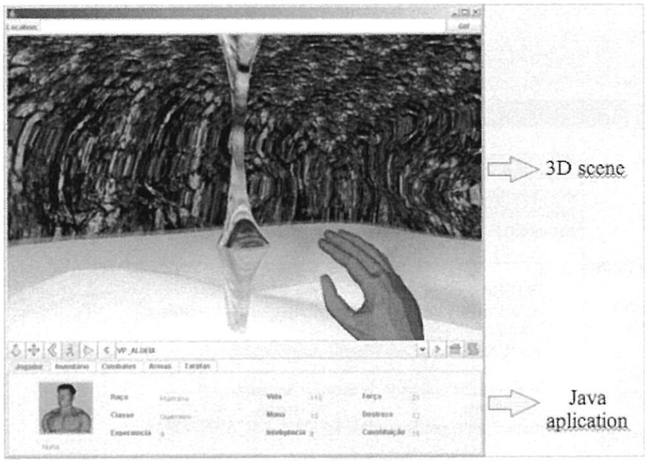


Fig. 10. Graphic interface with two components: 3D scene and Java application

In the beginning of the game the user must select and configure the characteristics of his character, like the race, colour, strength, dexterity, intelligence, etc. In this initial phase there are many user interaction possibilities within the three-dimensional scene. Through the Java developed interface it was possible not only to manipulate some characteristics of the character, but also to add new elements in the 3D scene. After this, the player will be able to explore a virtual world in a medieval fantasy environment. The created world is made of different environments, such as mountains, caverns, villages, castles and forests. The player should carry out some tasks and explore the world the way that he wants. There is not a linear way to play the game, and the player has a big range of freedom in its exploitation. He will be able to decide with which character he speaks, which monsters attack, when he should flee, etc.

To play sounds in a X3D scene it was necessary the installation of OpenAL [18], a library used in applications and games that require audio 3D.

In Fig. 11 several images of the prototype developed in this project are presented.

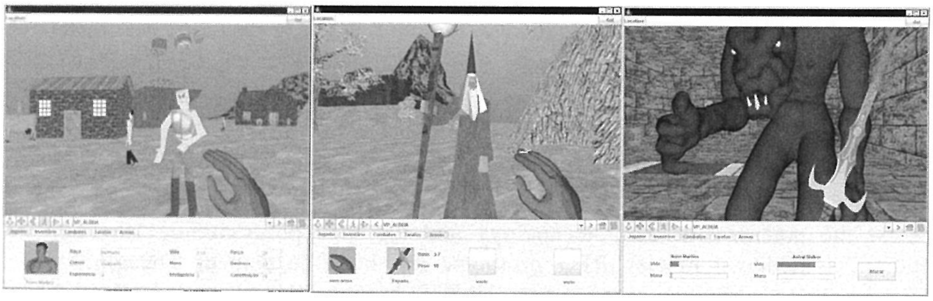
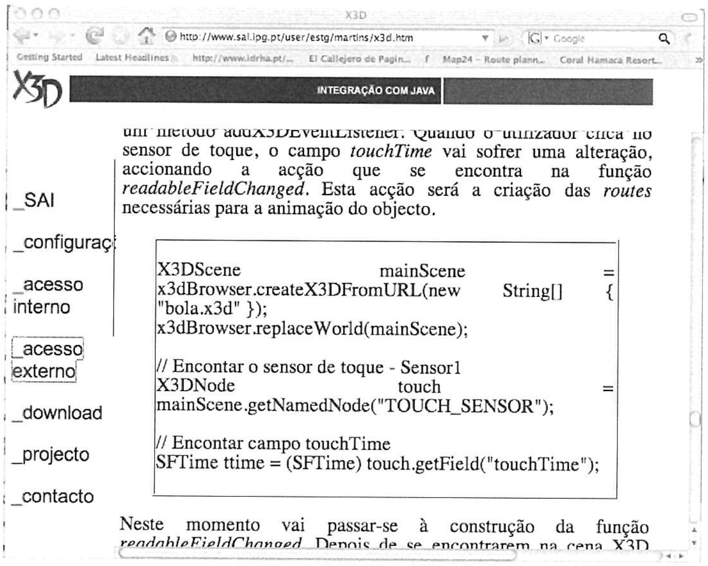


Fig. 11. Images of the developed game

A Website where several programming code examples of the X3D and Java integration can be found, has also been developed (Fig. 12).



um método addX3DEventListener. Quando o utilizador clica no sensor de toque, o campo *touchTime* vai sofrer uma alteração, accionando a acção que se encontra na função *readableFieldChanged*. Esta acção será a criação das *routes* necessárias para a animação do objecto.

```
X3DScene mainScene =
x3dBrowser.createX3DFromURL(new String[] {
"bola.x3d" });
x3dBrowser.replaceWorld(mainScene);

// Encontar o sensor de toque - Sensor1
X3DNode touch =
mainScene.getNamedNode("TOUCH_SENSOR");

// Encontar campo touchTime
SFTime ttime = (SFTime) touch.getField("touchTime");
```

Neste momento vai passar-se à construção da função *readableFieldChanged*. Depois de se encontrarem na cena X3D

Fig. 12. Web tutorial with programming code examples.

5. Conclusion

In this work a decision was taken to use open source 3D technology – X3D. The choice of the X3D instead of the VRML seem logic, considering that X3D came after VRML, containing, amongst other advantages, a much stronger scene codification. The objects of the 3D scene are, actually, in XML code, instead of text, as it used to happen with VRML.

The first objective of this project was not the creation of a game, but the creation of an application that would demonstrate the capacities and potencialities of the conection between the X3D world and a Java application. The implementation of the game demonstrates many characteristics of this integration. The capacity of using an external programming language to add specific behaviours to 3D scene objects raised the power of interactivity turning the X3D much more powerfull.

Although the prototype developed was directed to the specific area of entertainment, the integration of these two technologies can be used in different fields of application like architecture, medicine, teaching, publicity, and others.

The used browser – Xj3D – works properly with the fusion of the X3D with Java and supports most of the characteristics of X3D. But, it still represents some instability due to the fact of still beeing in the phase of development. It is expected that a new version of this browser will be soon released, with many of its problems solved.

It is still important to mention the need of extra time for the correct configuration of the computer, in order to allow the comunication between X3D and Java.

The main dificulty in the development of this work was the lack of information about this theme. The existing information is not very clear, specially when it concerns the integration of X3D and Java. In order to overcome this lack of information in this recent area and, in some way, contribute to the development and dissemination of this new field, a website with several practical examples has been created. This website is online and can be consulted on the oficial site of Xj3D, in the tutorials section [19].

5.1 Future Work

The development of a game is not an easy task once it requires a teamwork action, each element with a specific function. The future work will be directed to the creation of new game-areas and the implementation of new missions and objectives for the player to reach. Adding inteligence to the characters, making a more realistic world, corresponds to another future aim.

During this work it was impossible to use SAI in on-line application, due to the fact that the Xj3D is a stand-alone browser. Thus another objective will be to make the created game available on the Internet.

Acknowledgments. Special thanks goes to Alan Hudson, Xj3D project manager and code programmer, for his unconditional support and valuable comments.

References

1. The Virtual Reality Modeling Language, Dezembro 2007 http://www.web3d.org/x3d/specifications/vrml/VRML1.0/index.html
2. X3D and Related Specifications, Dezembro 2007 http://www.web3d.org/x3d/specifications/
3. Harney, J., Blais, C., Hudson, A., Brutzman, D.: Visualizing Information Using SVG and X3D: X3D Graphics, Java and the Semantic Web. In: Geroimenko, V., Chen, C. (eds.) 2005. LNCS, vol.5555, pp. 10-10. Springer, Heidelberg (2005)
4. Rudolph, M., Zhang, Y. J.: A universal java interface to native 3D rendering platforms using multiple scenegraph representations. 7th International Conference on Computer-Aided Industrial Design & Conceptual Design. pp. 413-417. IEEE Xplore (2006).
5. X3D – Integração com Java, Dezembro 2007 http://www.sal.ipg.pt/user/estg/martins/x3d.htm
6. Roehl, B.: Playing God – Creating Virtual Worlds with Rend386.Wait Group Press, EUA (1994)
7. Shneiderman, B, Plaisant, C..: Designing the user interface – Strategies for Effective Human-Computer Interaction. Addison Wesley, EUA (1998)
8. Cosmo Worlds 2.0 gives 3-D Web authoring a boost, Janeiro 2008 http://www.infoworld.com/cgi-bin/displayArchive.pl?/98/20/i12-20.81.htm
9. Media Machines - Developer Resources and Forums, Janeiro 2008 http://www.mediamachines.com/developer.php
10. AccuTrans 3D by MicroMouse Productions, Janeiro 2008 http://www.micromouse.ca/
11. Hartman, J., Vogt, W.: Cosmo Worlds 2.0 User's Guide. Silicon Graphics, Inc, EUA (1998)
12. Ballreich, C.: Late Night VRML 2.0 with Java. Texture Map Creation. Ziff-Davis Press, Macmillan Computer Publishing (1997).
13. Miranda, J.C., Sousa, A.A.: Urbanismo e Ambientes Virtuais – Divulgação e Discussão na Comunidade. Virtual - Revista Electrónica de Visualização, Sistemas Interactivos e Reconhecimento de Padrões - ISSN: 0873-1837. (2000)
14. Extensible 3D (X3D) - Part 2: Scene Access Interface (SAI) ISO/IEC 19775-2:2004, Janeiro 2008 http://www.web3d.org/x3d/specifications/ISO-IEC-19775-X3DAbstractSpecification/
15. Xj3D - Java based X3D Toolkit and X3D Browser, Janeiro 2008 http://www.xj3d.org/
16. Hudson, A.D., Couch, J., Matsuba, S.N.: The Xj3D browser: community-based 3D software development. ACM SIGGRAPH 2002 conference abstracts and applications, pp. 327-327. ACM Press, Texas (2002)
17. 3dtest - 3D Temps reel et interactive, Janeiro 2008 http://www.3d-test.com/interviews/xj3d_1.htm
18. openAL - Cross-Platform 3D Audio, Janeiro 2008 http://www.openal.org/
19. Xj3D Tutorials - Portuguese SAI Tutorial, Janeiro 2008 http://www.xj3d.org/tutorials/index.html

Métodos de Iluminação Global Baseados em Fotões

Alexandre António de Oliveira Azevedo

Faculdade de Engenharia da Universidade do Porto
Rua Roberto Frias, Porto
Pro07001@fe.up.pt

Abstract. Este paper apresenta um estado de arte acerca dos métodos de iluminação global baseados em fotões. Inicialmente é feita uma breve introdução às características da iluminação global, podendo estas ser formuladas como a solução da equação de *rendering* ou da equação potencial. Posteriormente, são analisados diversos métodos de iluminação global baseados em fotões de um modo unificado que também permite fazer comparações.

Keywords. Computer Graphics, Global Illumination, Photon Tracing, Light Tracing, Photon Maps, Monte Carlo Ray Tracing, Rendering.

1. Introdução

Este documento apresenta uma descrição relativa aos métodos de iluminação global baseados em fotões. Métodos esses que constituem um importante contributo para a evolução da computação gráfica, nomeadamente, a produção de imagens tridimensionais visualmente realistas.

Os algoritmos de iluminação global têm como objectivo a simulação de todas as reflexões de luz numa cena virtual, considerando a reflexão e refracção difusas e a reflexão e refracção especulares e a representação exacta da intensidade da luz em cada ponto da mesma. A equação de *rendering* constitui a base matemática de todos os algoritmos de iluminação global e pode ser utilizada para calcular a radiação emanada em qualquer superfície num modelo. Devido à sua simplicidade, generalidade e independência dimensional, o método *Monte Carlo* ou o método *random walk* é um dos mais importantes métodos na resolução do difícil problema da iluminação global. Isto acontece, especialmente quando o método Monte Carlo é um método de ultimo recurso quando todos os outros métodos analíticos ou numéricos falham [1].

O transporte de luz ou as múltiplas interacções de luz entre as superfícies que produzem a iluminação global, podem modelar-se ou formular-se pela equação de *rendering* ou pela equação potencial.

Em 1986, com base na física da propagação da luz, Kajiya [2] apresentou a equação de *rendering* ou integral que podia ser vista como o problema central da iluminação global. Kajiya descreveu as aproximações que eram feitas por vários algoritmos que haviam surgido na computação gráfica, como os algoritmos de iluminação directa, o de radiosidade e o de *ray tracyng*. A equação de *rendering* engloba esta grande variedade de algoritmos de *rendering* e proporciona um contexto unificado para os ver como aproximações mais ou menos exactas na solução de uma equação única.

A equação de *rendering* descreve o transporte de radiância através de um meio não-participativo num ambiente tridimensional (3D).

A equação de *rendering* apresenta-se da seguinte forma:

$$I(x, x') = g(x, x')\left[\epsilon(x, x') + \int_S \rho(x, x', x'')I(x', x'')dx''\right]. \qquad (1)$$

onde:

$I(x,x')$ está relacionado com a intensidade da luz passando do ponto x' para o ponto x

$g(x,x')$ é um termo "geometrico"

$e(x,x')$ está relacionado com a intensidade da luz emitida do ponto x' para o ponto x

$p(x,x',x'')$ está relacionado com a intensidade da luz passando do ponto x'' para o ponto x por um retalho de superfície em x'

A equação define que a intensidade de transporte de luz de um ponto de uma superfície para outro simplesmente é a soma da luz emitida e o total de intensidade de luz que se espalha para **x** de todos os outros pontos de superfície.

No mesmo artigo, apresentou também uma solução que resolvia de forma completa essa equação integral com base no método de Monte Carlo. O seu algoritmo generalizava as ideias do algoritmo de traçado estocástico de raios de R. Cook e Carpenter [3], sendo capaz de levar em consideração as inter-reflexões entre superfícies.

Embora capaz de calcular a troca de energia entre superfícies com características de reflexão arbitrarias, a principal desvantagem do algoritmo era exigir que fosse gerado um número muito grande de amostras para se produzir uma imagem com pouco ruído.

Em 1993 S.N.Pattanaik e S.P.Mudur [4] apresentaram a equação potencial que juntamente com a equação de rendering forma um sistema de junção de equações e proporciona uma framework matemática para todas as aproximações conhecidas de computação de iluminação baseadas em óptica geométrica.

S. N. Pattanaik e S. P. Mudur afirmam que a iluminação de qualquer ponto de uma superfície num ambiente complexo 3D é devida à emissão de luz daquele ponto (se existir) e/ou devido à reflexão daquele ponto de luz recebida de todas as direcções hemisfericas em redor daquele ponto [4].

Em objectos sólidos opacos, por causa das propriedades ópticas de superfícies, principalmente reflexão, a luz emitida de qualquer superfície em qualquer direcção pode iluminar muitas outras superfícies de um ambiente. Este fenómeno pode ser capturado pela noção de um potencial associada a todo o ponto e direcção no ambiente.

A equação potencial proporciona uma expressão para a capacidade potencial, W, de qualquer, (x,Θ_x) para a iluminação de S, uma serie de pontos e direcções ao redor desses pontos.

Emissão de um ponto x ao longo de Θ_x pode directamente e/ou indirectamente iluminar S. A radiação emitida a partir de (x,Θ_x) pode directamente ser tida em conta para a iluminação da serie S se (x,Θ_x) pertence à serie. Assim para representar o componente directo S. N. Pattanaik e S.P.Mudur utilizaram uma função g definida sobre todos os pontos de superfície e todas as direcções em redor desses pontos em que $g(x,\Theta_x)$ é 1 se $(x,\Theta_x) \in S$, e 0 caso contrario [4].

A quantidade de luz emitida de (x,Θ_x) responsável pelo componente de saída em S devido a uma ou mais reflexões pode ser exprimida da seguinte maneira. A emissão de qualquer (x,Θ_x) atingirá o ponto de superfície y mais próximo e depois possivelmente será reflectido. A fracção do fluxo incidente sendo reflectido em qualquer das direcções hemisféricas Θ_y à volta de y é:

$$f_r(y, \Theta_y, \Theta_x)cos\theta_y d\omega_y \tag{2}$$

Depois esta fracção vezes o potencial do ponto y ao longo de Θ_y integrado no hemisfério de saída à volta de y, ou seja, o componente indirecto é representado da seguinte maneira:

$$\int_{\Omega_y} f_r(y, \Theta_y, \Theta_x)W(y, \Theta_y)cos\theta_y d\omega_y \tag{3}$$

Assim, a expressão completa da função potencial é representada da seguinte maneira:

$$W(x, \Theta_x) = g(x, \Theta_x) + \int_{\Omega_y} f_r(y, \Theta_y, \Theta_x)W(y, \Theta_y)cos\theta_y d\omega_y \tag{4}$$

2. Métodos baseados em Fotões

Existem diversos métodos de iluminação global baseados em fotões. Estes metodos, de uma maneira geral produzem imagens com elevado grau de realismo, no entanto, requerem um esforço computacional elevado.

2.1 Photon tracing

Photon tracing é o inverso do método *visibility ray-tracing*[1] e utiliza suposições semelhantes e simplificadas [5].

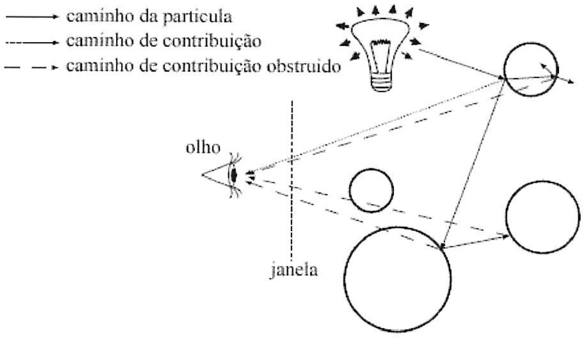


Fig. 1. Método photon tracing

Também deixa de localizar ao bater uma superfície que não tem reflexão coerente ou refracção. Em *photon tracing* os raios são emitidos das fontes de luz, e a cada embate a superfície é examinada para saber se tem reflexão/refracção ideal, e reflexão/refracção incoerente. Nas direcções de reflexão ideal ou refracção, o traçado é continuado começando raios filhos novos (Figura 1). O efeito das interacções incoerentes é armazenado num mapa ou é projectado para o olho traçando-o para a posição da câmara.

2.2 Light Tracing

O método *light tracing* [6], [7] baseia-se na simulação do modelo de fotões de luz.

[1] Visibility Ray Tracing é um método de iluminação global que apenas modela reflexões ideais e transmissões (também designadas como *componentes coerentes*) que seguem leis de geometria óptica como a lei de reflexão e a lei de Snellius-Descartes de refracção mas não tem em conta a reflexão especular ou refracção múltipla e difusa ou incoerente.

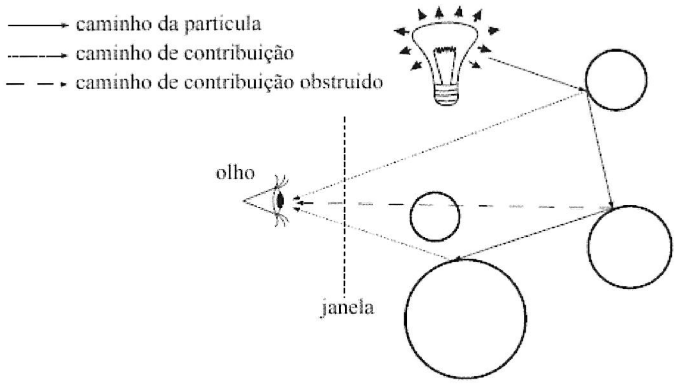


Fig. 2. Método light tracing

Estes fotões são propagados à medida que são gerados nas fontes de luz. Neste método os fotões executam um passeio fortuito pela cena que começa nas fontes de luz. Sempre que embatem numa superfície, um raio é traçado desde o ponto de intersecção ao olho (figura 2) e a contribuição é acrescentada ao pixel seleccionado (se existente). O algoritmo do *light tracing* é um algoritmo probabilistico *"shooting"*. O *sampling* baseia-se nas funções bidireccionais de distribuição de reflexão (BRDF[2]) nas superfícies reflectidas, e numa função de densidade de probabilidade adaptativa (PDF) nas fontes de luz.

O método *light tracing* é a implementação directa da quadratura Monte-Carlo da formulação multi-dimensional da equação potencial. Quando a próxima direção é determinada, *BRDF based importance sampling* pode ser aplicado e combinado com Roleta-russa. Escolhe uma direcção fortuita de acordo com a densidade t_i que é aproximadamente proporcional a w_i (importance sampling). O passeio é continuado com uma probabilidade a_i igual à aproximação do albedo (roleta russa). O valor medido de um único passo do caminho é

$$P = \frac{L^e \cos\theta}{N \cdot p^e} \cdot \frac{w_1}{t_1 \cdot a_1} \cdot \frac{w_2}{t_2 \cdot a_2} \cdot \ldots \cdot w(eye) \cdot g. \tag{5}$$

se este ponto é visível ao pixel e caso contrário zero. Aqui L^e é a emissão do ponto de partida, Θ é o angulo entre a superfície normal da fonte de luz e a primeira direcção, p^e é a probabilidade de seleccionar este ponto de fonte de luz e direcção inicial,

[2] A Bidirection Reflectance Distribution Function, BRDF foi introduzida por Nicodemus et al. Como uma ferramenta para descrever a reflexão de luz numa superfície. A função BRDF é uma aproximação simplificada da função BSSRDF. A BRDF assume que a luz que atinge uma superfície num local dessa superfície, é reflectida nesse mesmo local e descreve o modelo de iluminação local.

$w(eye)$ é o coseno avaliado de BRDF a um dado ponto desde a última direcção para o olho, e g é o parâmetro de câmara dependente da superfície. Se o *sampling* BRDF ideal for utilizado, ou seja, se w_i for proporcional a t_i e ambos w_i/t_i e a_i forem iguais ao *albedo*, e a sampling da fonte de luz ideal for utilizada, ou seja, se p^e for proporcional a $L^e \cos\Theta$, assim, cos de $L^e \cos\Theta/N \cdot p^e = \Phi/N$, a estimativa seguinte pode ser obtida:

$$P = \frac{\Phi}{N} \cdot w(eye) \cdot g. \qquad (6)$$

Esta estimativa tem variação alta se a câmara for frequentemente escondida dado que se o ponto não é visível da câmara, a contribuição é zero.

Este algoritmo também aplica BRDF sampling para todos menos os últimos passos. A direcção do último raio de visibilidade pode estar longe da direcção preferida pelo BRDF. Isto degrada o desempenho de sampling de importância se a superfície visível é muito brilhante. Assim, espelhos visíveis ou refractares (vidro) coloca dificuldades aos algoritmos *shooting*.

As vantagens do *light tracing* [7] são as seguintes:

- Todos os tipos de transporte de luz são tratados de uma maneira uniforme.
- Não é necessária nenhuma malha.
- Todos os percursos da luz são gerados correctamente.
- As partículas são disparadas em direcções com alto potencial de capacidade.

As desvantagens são:

- O algoritmo é dependente da vista.
- A qualidade das imagens depende fortemente do número de partículas que são disparadas.
- Detalhes de alta frequência tais como reflexões, refracções ou causticos[3] são difíceis de ser renderizados correctamente.

2.3 Photon-mapping

O algoritmo *photon mapping* proposto por Jensen é muito utilizado em computação de iluminação global e bastante popular no meio académico e na industria. Este método funciona bem mesmo em ambientes complexos e todas as direcções de luz podem ser facilmente simuladas (figura 3).

[3] Um cáustico é um padrão de luz que é focalizado numa superfície depois de ter tido o caminho original de raios de luz alterado por uma superfície intermediária..

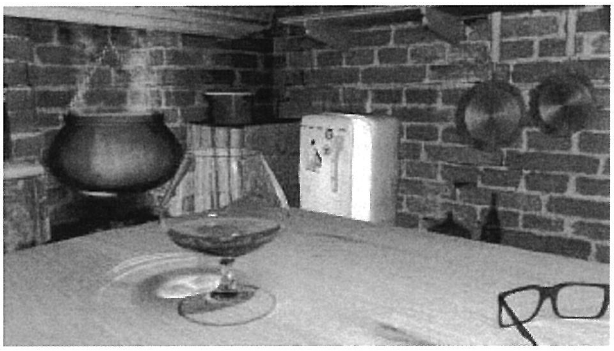


Fig. 3. Exemplo de um ambiente complexo com base em photon-mapping

O método *photon-mapping* [8], [9] constitui uma aproximação estocástica à solução da equação de *rendering*, que permite calcular cáusticos, interreflexões difusas, meios participativos (ex.: fumo, nevoeiro), etc.

Por exemplo, como os raios de luz atravessam uma taça de vinho (Figura 4) em cima de uma mesa e o líquido que contém, eles são refractados e focalizados na mesa. O vinho também muda o padrão e cor da luz.

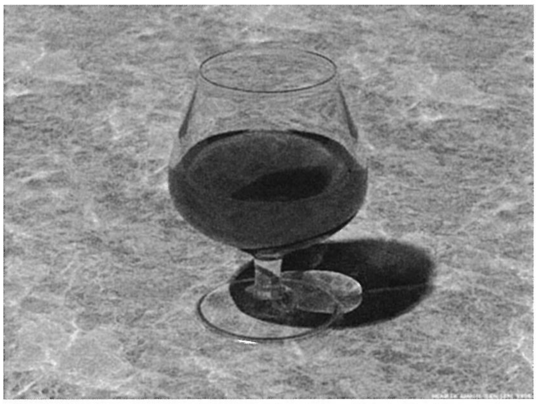


Fig. 4. Exemplo de cáusticos

Este fenómeno também é frequente em superfícies metálicas curvas.

Particularmente, este método é mais eficiente que todas as outras técnicas existentes no *rendering* de alta qualidade de efeitos cáusticos.

Um *photon map* é uma colecção de fotões que atingem o término dos caminhos gerados na fase do algoritmo de *shooting*. O *photon map* é organizado numa *KD-tree* para efectuar uma recuperação eficiente. O embate dos fotões é armazenado com o poder de cada fotão em comprimentos de onda diferentes, posição, direcção de chegada e com a superfície normal.

O *Photon mapping* [10], [11] convencional pode ser visto como um processo constituído por duas etapas:

- Construção dos *photon maps:* Nesta etapa, os *photon maps* são construídos com base na emissão de um grande número de fotões (pacotes de energia) desde as fontes de luz até cada superfície da cena.. Cada fotão é localizado na cena através da utilização de um método semelhante ao *path tracing*. Sempre que um fotão atinge uma superfície é armazenado dentro do *photon map* e um método roleta russa é utilizado para determinar se o fotão é absorvido ou reflectido. A direcção nova de um fotão reflectido é calculada utilizando o BRDF da superfície.

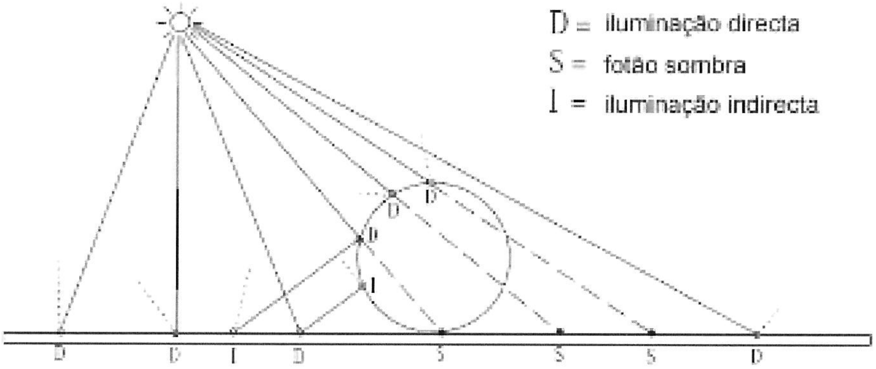


Fig. 5. Os fotões no photon map global são classificados para optimizar o rendering de sombras

São também utilizados fotões de sombra, traçando raios com origem na fonte luminosa ao longo de toda a cena. No primeiro ponto de intersecção um fotão normal é armazenado e nos pontos de intersecção seguintes são armazenados fotões de sombra. Estes fotões de sombra são utilizados durante a fase de *rendering* para reduzir o número de raios de sombra (Figura 5).

Existem dois *photon maps* separados, um *caustics photon map* para armazenar fotões relativos a cáusticos e um *global photon map q*ue é utilizado como uma aproximação do fluxo de luz na cena e é criado pela emissão de fotões para todos os objectos. Esta separação, contribui para melhorar a velocidade, reduzir a necessidade de memória e aumentar a exactidão o método.

Os fotões são armazenados numa *KD-tree* equilibrada. Esta estrutura de dados permite um *rendering* mais eficiente e reduz a necessidade de memória para o embate de cada fotão, permitindo representar cada fotão em apenas 20 bytes.

- *Photon rendering*: Etapa em que é feito o *render* da imagem final, sendo a luminosidade de cada pixel calculada com base na media de um conjunto de amostras. Cada amostra consiste em traçar um raio desde o olho até cada pixel na cena. A luminosidade devolvida por cada raio é calculada na primeira superfície interceptada pelo raio e é igual à luminosidade de superfície, $L_s(x, \Psi_r)$, deixando o ponto de intersecção x, na direcção Ψ_r do raio. $L_s(x, \Psi_r)$ é calculada através da seguinte equação de rendering:

$$L_s(\mathbf{x}, \Psi_r) = L_e(\mathbf{x}, \Psi_r) + \int_{\Omega} f_r(\mathbf{x}, \Psi_i; \Psi_r) L_i(\mathbf{x}, \Psi_i) cos\theta_i \, d\omega_i \tag{7}$$

Onde L_e é a luminosidade emitida pela superfície, L_i é a luminosidade de entrada na direcção $\Psi_{i,}$ f_r é o BRDF e Ω é a esfera das direcções de entrada. L_e é considerada directamente a partir da definição de superfície, não necessitando de outros cálculos. O valor do integral $L_{r,}$ depende dos valores de luminosidade do resto da cena e pode ser resolvido directamente utilizando tecnicas de *Monte Carlo* como o *path tracing*.

A equação de retribuição (7) pode ser dividido num conjunto de vários componentes. Sendo L_r expresso como:

$$\begin{aligned} L_r \;\; = \;\; & \int_{\Omega} f_r L_{i,l} \cos\theta_i \, d\omega_i + \\ & \int_{\Omega} f_{r,s}(L_{i,c} + L_{i,d}) \cos\theta_i \, d\omega_i + \\ & \int_{\Omega} f_{r,d} L_{i,c} \cos\theta_i \, d\omega_i + \\ & \int_{\Omega} f_{r,d} L_{i,d} \cos\theta_i \, d\omega_i \end{aligned} \tag{8}$$

$$f_r = f_{r,s} + f_{r,d} \quad \text{and} \quad L_i = L_{i,l} + L_{i,c} + L_{i,d}$$

Nesta equação (8), a luminosidade de entrada foi dividida em contribuições das fontes luminosas, $L_{i,l}$, contribuições das fontes luminosas por reflexão de especular (cáusticos), $L_{i,c}$ e iluminação indirecta suave, $L_{i,d}$ (luz que foi difusamente reflectida pelo menos uma vez. O BRDF encontra-se separado numa parte de difusa, $f_{r,d}$, e numa parte especular $f_{r,s}$.

2.4 Ray Maps

Havran, Bittner, Herzog e Seidel apresentaram uma estrutura de dados mais eficiente para representar o transporte de luz, chamada *ray maps* [12]. *Ray maps* consiste basicamente numa eficiente estrutura de indexação baseada em *KD-tree*: O espaço é subdividido por *KD-voxels* que guardam referencias de todos os segmentos de raio que interceptam um *voxel*. De modo semelhante ao *photon mapping* convencional, os raios vizinhos são determinados pela expansão de uma esfera à volta do ponto onde a radiação tem de ser calculada. Deste modo, todos os raios referenciados por *voxels* que interceptam a esfera são ordenados relativamente à distância deles para aquele ponto. Para calcular esta distância várias métricas foram propostas, as quais significativamente influenciam a estimativa de radiação.

Embora os *ray maps* sejam claramente superiores para a *ray cache* Havran et al. reportaram tempos de *rendering* até cinco vezes superiores ao convencional *photon mapping*. Apesar de algumas estratégias reduzirem a utilização de memória e tempo de computação os *ray maps* ainda podem ser muito custosos:

- *Ray maps* têm a necessidade de armazenar referências a raios em todos os lugares ao longo de um raio, potencialmente até mesmo em espaço vazio. Isto aumenta as exigências de memória dado que são armazenadas muitas referências que nunca serão utilizadas para uma estimativa de radiação.
- Se o *lookup radius* R é muito maior que a extensão de um *KD-Voxel* muitas referências para um único raio podem ser encontradas num *lookup* vizinho mais próximo. Esta situação muito comum conduz um custo indirecto significaste.

Além disso a resolução de *ray maps* é completamente determinada pela densidade de raio (limitada pela definição de algum utilizador de maximo de *KD-tree* de profundidade definida). Assim não há nenhuma possibilidade para adaptar localmente a sua resolução de acordo com a geometria de uma cena.

2.5 Efficient Ray Based Global Illumination Using Photon Maps

Arno Zinke e Andreas Weber introduziram uma técnica baseada em raios para aproximar a iluminação global para cenas complexas [13]. Estes autores consideram os seus desenvolvimentos simples, no entanto, constituem uma alternativa mais eficiente do que anteriores métodos tais como o *ray mapping,* quando a exactidão perfeita no *rendering* de cenas complexas de imagens tridimensionais não é necessária. Esta técnica combina a simplicidade e a eficiência do *photon mapping* com as técnicas baseadas em raios. Referências a raios deveriam ser adaptativamente distribuídos com respeito para com a geometria na cena evitando referencias desnecessárias em espaço vazio e permitindo uma definição pelo utilizador de um intercâmbio entre precisão e eficiência.

Em contraste com *photon mapping* convencional, onde a estimação de radiação só tem em conta fotões em superfícies, a estimação de densidade baseada em raios necessita de procurar os n segmentos de raio mais próximos com respeito a um determinada posição no espaço. Para eficazmente encontrar os raios próximos, um único segmento de raio é representado por vários fotões num *photon map* (Figura 6).

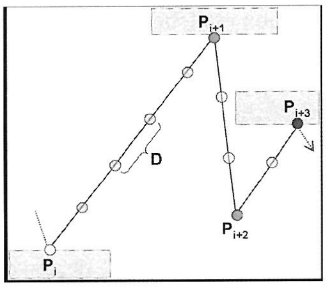


Fig. 6. Segmentos de raio são representados por vários fotões no photon map

Se um destes fotões é encontrado na procura da vizinhança mais próxima, o caminho de luz está a interceptar a esfera (Figura 7).

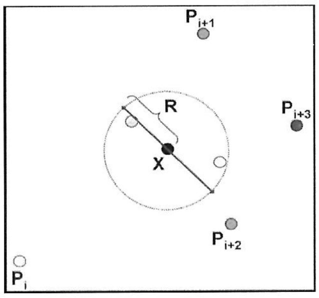


Fig. 7. Identificação de segmentos de raio vizinhos em redor de um ponto X

A principal vantagem de armazenar fotões referentes a um raio (em relação ao armazenamento de referências para raios em *ray maps*) é a de que as densidades dos fotões podem ser adaptadas a uma cena, considerando que para *ray mapping* a resolução dos *ray maps* é determinada pela densidade de raio local. Por exemplo nenhuma referência precisa de ser armazenado em espaço vazio. É de notar que as referências de fotões utilizam muito menos memória que os fotões convencionais (nesta implementação 8 bytes por fotão). Porém, para aumentar a eficiência, um fotão é apenas armazenado no *map* se for um contributo potencial numa estimativa de radiação. Isto é alcançado pela utilização de uma *grid* binária para identificar regiões vazias. Se um *voxel* desta grid é interceptado por um objecto este é marcado como ocupado ou vazio caso contrário. A resolução espacial de um *voxel* é escolhida para ser igual ao *lookup radius* R máximo. As referências de fotões são apenas armazenadas no *map*, se o *voxel* que inclui o fotão está ocupado ou tem pelo menos um *voxel* vazio na vizinhança.

3. Conclusão

Os métodos de iluminação global baseados em fotões têm tido uma evolução significativa ao longo do tempo em termos de produção de imagens tridimensionais de cenas complexas cada vez mais realistas. No entanto ainda se verifica a existência de algumas limitações em termos de de capacidade de processamento e de memória que estes métodos necessitam para ser possível correrem inteiramente em GPUs. Desenvolvimentos como os de Arno Zinke e Andreas Weber que procuram optimizar e aumentar a eficiência ou eficácia do *rendering* de imagens realistas em detrimento da perda de alguma exactidão constituem um dos caminhos para desenvolvimentos futuros nesta área. Por outro lado, é previsível que à medida que os GPUs de nova geração forem surgindo, essa necessidade de optimização se torne cada vez menos crítica e proporcione uma maior abertura para o desenvolvimento de algoritmos cada vez mais aperfeiçoados e de representação cada vez mais realista de imagens tridimensionais complexas.

Agradecimentos

Ao Prof. Dr. António Augusto de Sousa e ao Prof. Dr. Fernando Nunes Ferreira pela discussão elucidativa para o desenvolvimento deste trabalho.

Referencias

1. Bekaert, P., *Hierarchical and Stochastic Algorithms for Radiosity*. 99, Katholieke Universiteit Leuven.
2. Kajiya, J.T. *The Rendering Equation*. in *ACM Computer Graphics (SIGGRAPH '86 Proceedings)*. 86.
3. R. Cook, T.P. and L. Carpenter, *Distributed ray tracing*, in *Computer Graphics (SIGGRAPH '84 Proceedings)*. 84. p. 137-145.
4. Pattanaik, S.N. and S.P. Mudur, *Eficient Potential Equation Solutions for Global Illumination Computation*, in *Pergammon Press*. 93. p. 1-19.
5. László, S.-K., *Monte-Carlo Methods in Global Illumination*. 99/00, Vienna: Institute of Computer Graphics - Vienna University of Technology.
6. Dutre, P., E. Lafortune, and Y.D. Willems, *Monte Carlo light tracing with direct computation of pixel intensities*, in *Compugraphics '93*. 93, Alvor. p. 128-137.
7. Dutré, P. and Y.D. Willems, *Importance-driven Monte Carlo Light Tracing*, in *5th Eurographics Workshop on Rendering, Darmstadt, Germany* 94, Department of Computer Science - Katholieke Universiteit Leuven.
8. Jensen, H.W., *Realistic Image Synthesis Using Photon Mapping*. 01, Natick, Massachusetts: A K Peters.
9. Jensen, H.W. and N.J. Christensen, *Photon Maps in Bidirectional Monte-Carlo Ray-Tracing of Complex Objects*. Computers & Graphics, 95. 19(2): p. 215-224.
10. Jensen, H.W., *Global Illumination using Photon Maps*. 96, Department of Graphical Communication - The Technical University of Denmark.
11. Jensen, H.W. and P. Christensen, *High Quality Rendering using Ray Tracing and Photon Mapping*, S.D. University of California, Editor. 07, Pixar Animation Studios.
12. Havran, V., et al., *Ray maps for global illumination*, in *Eurographics Symposium on Rendering*. 05. p. 43-54.
13. Zinke, A. and A. Weber, *Efficient Ray Based Global Illumination Using Photon Maps*, I.f. Informatik, Editor. 07.

Sensing the World: Challenges on WSNs

Válter Rocha

Instituto Agilus de Inovação em Tecnologia de Informação, S.A.
Rua Dr. Afonso Cordeiro 877, sala 202
4450-007 Matosinhos
Portugal
valter.rocha@iaiti.pt
FEUP - Faculdade de Engenharia da Universidade do Porto
valter.rocha@fe.up.pt

Abstract. Wireless sensor network (WSNs) is a demanding multidisciplinary field that has been target of research in the last decade. The increasing demand for security and automated monitoring of things and places makes WSNs a promising technology although there are still many open research contributions needed to make this the standard of environmental sensing.

The goal of this article is to identify the research challenges on WSNs by dividing them into functional groups, building on previous work. We followed a structured approach based on a simplified yet complete vision of a design space for WSNs. Moreover, this work aims to identify research gaps and investigation fields yet unexplored or hardly explored by researchers in order to plot paths for future research. Several challenges and research areas were identified, like Models for Sensor Networks, Benchmarking Methodologies, Distributed Processing, Interface WSNs and Network Reprogramming.

Some of these challenges will become even clearer as increasingly WSNs become Wireless Sensor and Actuator Networks (WSANs).

Key Words: Wireless Sensor Networks, WSN: Pervasive Networking, Mesh Networking.

1 Introduction

Embedded devices have been part of our lives for the last 50 years but just recently the advances in semi-conductor technology, in low power wireless radios and MEMS (Micro-Electro-Mechanical Systems) technologies have enabled the production of cost effective, low-power, and small scale devices. These inexpensive, low-power communication devices can be deployed throughout a physical space, providing sensing, processing and communicating capabilities.

WSNs are the breakthrough approach based on networks of devices that can be densely deployed in human aggressive and inaccessible environments, to sense and instrument the environment and monitor with high accuracy physical phenomena. Each one of these devices is called a sensor node. Each node should not be larger than a few square millimeters and its target cost is less than US$1.00, including radio, microcontroller, power supply and sensor (capable of sensing temperature, light, vibration, sound, etc.).

To realize the opportunities created by this concept, information technology must address a new collection of challenges. The individual nodes in a WSN are inherently resource constrained: They have limited processing speed, storage capacity, and communication bandwidth. These devices have substantial processing capability in the aggregate, but not individually, so we must be able to combine their many capabilities within the network itself. Due to the multidisciplinary nature of WSNs, these challenges are somehow disperse. Consequently each study on the topic presents a view on existing challenges and difficulties present in a certain field.

In order to present these challenges in a systematic way, a structured approach was followed. This approach was based on the work presented in [1] where the need for standardization in order to coordinate efforts and encourage developments in this area is discussed.

This paper is organized as follows. We start by describing the Design Space in Section 2, continue by presenting some possible applications on WSNs in Section 3. On sections 4 and 5 we discuss the architecture of a single node and an entire network. Section 6 presents technology gaps and their respective research paths. Finally we discuss the conclusions and future work in section 7.

2 Design Space

Although WSNs share common problems and solutions with embedded systems and ad hoc networks in what concerns hardware and network issues, there are new challenges that rise from the fact that these devices are resource constrained.

Several researchers approach the design of WSNs on an application oriented basis while others, like for example Römer and Mattern [1], structure such challenges in classes and name this structure "The Design Space of Wireless Sensor Networks". Based on this work, we propose our simplified yet complete vision of the design space by refactoring the one proposed in [1]. Some proposed design space dimensions like deployment, coverage and network size or like network topology and connectivity relate to the same problem influencing each other, therefore should not be dissociated. Others like Application requirements and Environment interaction that relate to the application specific needs should be introduced to further improve the design concepts on WSNs.

2.1. Application requirements and Environment interaction

Application specific requirements define almost every other topic in WSNs. The large amount of heterogeneous sensor nodes can be used freely to create many different solutions. There is no such thing as a general solution still research should keep generalization in mind.

As WSNs are environmental event-driven, their activity graph can vary a lot during time. While most of the time activity is very low, when some environmental change occurs events come in bursts and can generate traffic problems.

2.2. Network Dynamics

The deployment and mobility of nodes affect the network topology as there is uncertainty on the nodes location and density. Moreover, some nodes can be deployed or arrive to a specific network, attached to an object or by their own means, becoming active members and making deployment a continuous process. This forces the network protocols to be flexible and dynamic in order to react to the different demands of applications. Also the degree and speed of movement can influence in the time the nodes are available and therefore their usability in the network.

2.3. Cost, Size, Resources, Energy and Lifetime

The application design requirements determine the size, cost and energy specifications. While some applications may need thousands of simple nodes, highly resource constrained, others could require just some specialized and complex nodes or a mix of the two solutions. Similarly the price of each node can vary from be as low as US$1.00 or reach US$1000.

Energy resources, and consequently the WSN lifetime, are directly dependant on cost and size factors as the size of the batteries or harvesting devices is directly proportional to the amount of energy they accumulate or generate. Also computing power and storage depend on the price, although currently this is not such a relevant issue because used Microcontroller Units and flash storage chips are at a very low price.

The amount of resources available at the nodes is also a determinant factor in the degree of complexity of each sensor node's embedded software and possibly with the entire network's routing and processing capabilities.

2.4. Heterogeneity and Complexity

Sensor networks can be deployed in practically any scenario and may require many types of distinct sensor nodes to accomplish the expected results. Differences in nodes range from computing power to size, sensor specialization or mobility.

The more heterogeneous the network is the more powerful and generic it becomes. But on the other hand, more robust and complex routing and communications protocols it requires.

Heterogeneous networks may use nodes with high processing capabilities or high energy resources to distribute processing efforts and power usage amongst the network nodes.

Typical configurations use a more powerful specialized node (sink node) to transmit data generated in the network to a computer source via internet, GPRS/UMTS or satellite, and possibly providing position by means of a GPS device.

2.5. Infrastructure and Communication Modality

As infrastructures are very expensive and sometimes impossible to deploy, Ad Hoc networks are becoming the standard in WSNs. While in the first case, communications are made via the base station in the second nodes communicate with each other directly without the need for an infrastructure. Nodes usually serve as routers, forwarding messages hop-by-hop on behalf of other nodes.

In both cases several communication modalities can be used: radio, diffuse light, laser, inductive and capacitive coupling or sound. The most common is radio transmission because it doesn't need line of sight which is a requirement in many applications.

2.6. Network Topology and connectivity

The application requirements play an important role in the Network Topology by defining either if the sensor coverage (area of interest covered by sensor nodes) should be dense or sparse, if the network should be more or less dynamic and the deployment area wide or narrow. A dense coverage is used if more accuracy and redundancy is required. The redundant sensor nodes could be used to make the network more robust replacing "dead" sensors or managing power savings between sensors by entering sleep mode. If the application requires mobility or a random deployment of some sensor nodes, the protocols have to be more dynamic to cope with the changes experienced in the network.

The Network Topology depends on its Diameter (max number of hops between any two nodes) and size (number of nodes) determined by the specific requirements.

The simplest topology is the single-hop network where every sensor can communicate directly to all the other sensors. A most elaborated topology is needed when nodes can't reach at least one node. To be able to communicate to a non reachable node, other nodes are used as routers. This is called multihop networking. In some applications on top of the arbitrary graph generated by the multihop network there is a simpler overlay structure that forms an organized tree or set of connected stars.

Data routing, distributed processing, latency and robustness are affected by the topology of the network.

The topology of the network and communications hardware ranges can influence the connectivity of the network. If every node can reach any node, the network is connected. If the nodes can are arranged in partitions and a mobile node provides communication occasionally, the communication is called sporadic. Communications protocols are influenced by connectivity.

2.7. Quality of Service (QoS)

QoS refers to the capability of a network to provide better service to selected network traffic over various technologies. A sensor network should provide the necessary quality of service to cope with the application needs. For example, if the application requires real-time access to data the network should be designed to support such feature. Minimal QOS requires that the network should always be operative and immune to: failures (robustness), deliberate attacks (tamper-resistance), Communications eavesdropping, and detection (unobtrusiveness or stealth). Moreover the network should provide the means to control and efficiently use network resources.

3 Applications for sensor networks

Such an enabling technology provides means for many new types of applications to become implementable. As described in [2], applications can be classified in three classes: i) monitoring space (environmental and habitat monitoring); ii) monitoring things (precision agriculture, indoor climate control, surveillance, treaty verification, and intelligent alarms); and iii) monitoring space and things (the interactions of things with each other and the encompassing space, the most complex interactions, including monitoring wildlife habitats, disaster management, emergency response, ubiquitous computing environments, asset tracking, healthcare, and manufacturing process flow).

Research projects that involve WSNs are on their way for military applications, environmental observation, intruder surveillance, bridge and tunnels structure monitoring, civil protection applications and many others. Examples on research activities are the system for bird observation on Great Duck Island [3], ZEBRANET [4], Glacier Monitoring [5], Cattle Herding [6], Bathymetry [7], Ocean Water Monitoring [8], Grape Monitoring [9], Cold Chain Management [10], Rescue of Avalanche Victims [11], Vital Sign Monitoring [12], Power Monitoring [13], Parts Assembly [14], Tracking Military Vehicles [15], Self-Healing Mine Field [16], Sniper Localization [17], Early Warning Fire Detection [18].

WSNs need to be designed to accomplish low cost, low power solutions. Both hardware platforms and network protocols have to be well suited to the intrinsic constraints of this new technology and, in the near future, when the cost and size problems are overcome, many other applications will become feasible.

4 Node Architecture

4.1. Hardware design

A typical hardware configuration for a wireless sensor node is composed of a micro-controller, data storage, radio / optical / ultra-sound transceiver, power supply and sensors.

The Micro-Controller Unit (MCU) is usually an Atmel or Texas Instruments, with analog-to-digital converters (ADCs), and SPI, I2C, as I/O inputs to interface with sensors and actuators. Some other important issues like chip computing power, data storage, size and cost, power consumption in awake and sleep modes and in transitions between the modes are decisive factors for choosing the MCU.

The size and low power constraints limit the amount of the on-chip storage. Typically MCUs reach 10 Kbytes of RAM for data and less than 100 Kbytes of ROM for program storage. However, small size Flash memory is getting cheaper and provides large amount of data storage for a low cost. Still the power consumption of such devices reaches 100mW or even more when in high speed write procedure so this has to be taken into account when designing the node.

Infineon or Chipcon radio devices are usually used in WSNs. The digital modulation techniques more common are amplitude-shift-keying (ASK) and frequency-shift-keying (FSK) although some researchers [19] are testing other modulation methods like On-Off-Keying (OOK).

Some nodes don't need to wake up much often because of the nature of measurements they have to make, but as they have to be in touch with the network they need to wake up to receive incoming messages. An important step towards low power communications would be the wake-up on radio transmissions. This way, nodes could save power by sleeping and when a message arrives, they wake up and respond.

The need of monitoring the world is the reason of the existence of WSNs and transducers play a great role on electrically representing physical phenomena. There are several types of sensors nowadays. While piezoelectric sensors are usually very expensive and big, MEMS transducers provide good accuracy for low cost and small size boosting the miniaturization of sensor nodes.

Every aspect of the WSNs' design has to take into account the power limitations and needs as this is certainly one of the most important issues in the WSNs' concept. As the deployment of sensor nodes mostly occurs in inaccessible places, power sources have to be designed to their network node's expected lifetime.

MCUs are getting smaller and more power efficient per clock frequency operating now at about 1mW @ 10MHz and 1uW @ sleep. Solar panels generate 10mW/cm^2 indoors and their internal peripherals like ADCs are now produced for low power consumptions.

4.2. Embedded Software

In a highly complexity technology such as WSNs, Operating Systems (OS) play an important role in solving many important design issues regarding hardware abstraction and resource management. OSs designed for WSNs are quite different from the ones used in traditional embedded systems mainly because of the power, processing and storage constraints that require power efficiency, reactivity, mobility, fault-tolerance, and concurrency.

There are two main types of OSs used in WSNs: Multithread-Driven is the traditional embedded systems approach adapted to WSN issues, where each task is given a CPU slot and all tasks concur to access resources; Event-Driven responds to events such as an incoming data packet or a sensor reading, by calling an event handler and allocating CPU and resources to solve the event in hands to completion. Event-Driven OS are best suited to address the requirements of WSNs because of the event intrinsic nature of WSNs.

Some examples of multithread OSs are RETOS [20] and MANTIS-OS [21]. The mostly used Event-Driven OS are TinyOS [22], SOS [23], Contiki [24], and Yatos [25]. Although Contiki runs an Event-Driven kernel, it introduces a new concept named protothreads that provides a linear, thread-like programming style on top of the kernel.

Another important concept when choosing an OS to run on WSNs is the support for safe and efficient re-programming of nodes. This is already addressed in some OSs like TinyOs, SOS and a virtual Machine for TinyOS named Maté [26] that allow a complete reprogramming of sensor nodes. Furthermore SOS and Maté allow efficient reprogramming of parts of the code by uploading the corresponding binary file.

Other issues like battery management, peripheral support and communications stack should be embedded in OS features.

5 Network Architecture

The concept of WSNs implies many different aspects that should be optimized for better power saving, performance and efficiency. Each application can demand different network topologies and different deployment densities. This impacts the communications and routing protocols design.

5.1. Physical, MAC, and Link layers

The design of the physical layer could improve significantly the energy efficiency of the network. Some factors like overhead, redundancy, etc. should be looked at when designing the physical layer. Some work on this topic can be found in [27] and [28].

Media Access Control (MAC) is one of the most active research areas in WSNs. The main research topic is related to keeping the nodes in power saving mode as much time as possible. Consequently, most of the work is closely related to TDMA.

Some work has been done regarding packet size, energy efficiency, Forward Error Correction (FEC) and transmission power variation on the energy spent per useful bit. In [29], the authors mention that there is work relating to "taking into account the degree of redundancy that an aggregated message carries on the link layer, which is much more specific to the situation in wireless sensor networks". However there is already some work done in this field.

Although most of the concepts related to addressing techniques are similar to Ad Hoc Networks, there are also specific problems. For example, geographic addressing nodes could be very helpful for routing algorithms. A new interesting concept is content-based addresses that seem very intuitive in WSNs because of their data-oriented nature.

5.2. Synchronization and Localization

In some applications the data acquired in all nodes makes sense as a whole and therefore needs to be synchronized. This is not as trivial as it could appear because there are delays in transmissions and there is no broadcasting clock to synchronize nodes. This is a much interesting research area for such applications. A good work in this area is [30].

The localization of sensor nodes using just the relative positions of the sensors is a very important and researched area in which many approaches have been made such as exploiting received signal strength indicators, time of arrival, time difference of arrival, or angle of arrival. Distributed algorithms are playing a great role in increasing precision.

5.3. Topology and Network Layer

When the number of nodes in the range of a particular node is big, a traditional flooding-based routing could quickly reach enormous amounts of repetition of messages. Two very common approaches to address this issue are transmission power control and clustering. Some other approaches stated in [31] include:

- Small Minimum Energy Communication Network: Creates a sub-graph of the sensor network that contains the minimum energy path.
- Gossiping: Sends data to one randomly selected neighbor. Avoids implosion problem but message propagation takes longer time.
- SPIN: Whenever a node has available data, it broadcasts a description of the data and sends it only to the sensor nodes that express interest.

- SAR: Creates multiple trees where the root of each tree is one hop neighbor from the sink. A sensor node selects a tree for data to be routed back to the sink according to the energy resources and additive QoS metric.
- LEACH: Forms a two level cluster hierarchy, where cluster members send data to the cluster head which in turn sends it to the base station. Energy dissipation is evenly spread by dissolving clusters at regular intervals and randomly choosing the cluster heads.
- Directed Diffusion: A sink sends out an interest which propagates in the network and sets up gradients for data to flow from source to sink.

The network layer is the most active research area after MAC and topology control. Some Ad Hoc solutions could serve as an inspiration to developers in this field, nevertheless there are specific issues regarding only WSNs (scalability, energy efficiency and data-centricness). The traditional routing algorithms unicast, multicast, anycast, and convergecast are still used in WSNs. New approaches designed specifically for WSNs, like geographic and data-centric routing, are also explained in this section.

Unicast routing protocols is a very well covered area in Ad Hoc Networks. The main contribution of this algorithm is the way it deals with energy as a scarce resource. It takes in consideration battery power in nodes to calculate the best path. A good example of unicast protocol is LEACH. However other protocols that disregard power may perform more efficiently in some circumstances.

Multicast is similar to unicast. Energy-efficiency problem is once again covered by this protocol. One emerging approach is stochastically constrained multicast, where just a certain percentage of nodes answers when a request is made. The objective is to rotate sleeping patterns of nodes. This can harmonize application requirements with lower layer behavior.

Anycast relates to sending messages to an object name that is multiple instantiated in the network. Typically the closest one is used. This is commonly useful for service discovery which is still a much unexplored area in WSNs.

Convergecast consists on collecting data from several nodes in a central point (Sink). This is an important concept in WSNs as it is very close to in-networking processing and aggregation concepts.

Geographic routing is defined as directing a packet not to a node but to a target area instead. Any node present in the area is a candidate destination node and can receive and process a message. This has an obvious importance for WSNs because of the dynamics of the networks and the environmental-driven monitoring.

Data-centric routing is perhaps the core abstraction of WSNs. It combines the applications need to access data by using a natural framework for in-network processing. When a sensor node has new readings publishes these values. Interested nodes can subscribe to such events. One of the most popular and more cited publications in this area is "directed diffusion" [32], even though some of its performance and functional characteristics are not entirely understood or explained. Another recent approach is similar to pear-to-pear data storage in the internet, but this is still not thoroughly investigated.

5.4. Transport

The transport protocol, just recently attracted the attention of researchers. This layer addresses mainly the issues of QoS versus the amount of energy needed to provide such service a good work can be found on [34].

6 Technology Gaps and Research Opportunities

As this is a novel technological area there are still many roads ahead and many gaps that can be filled. At the writing of this survey some important issues still lack an efficient approach and are therefore open for contributions from researchers with interest in this field. Some of these topics that have to be addressed are here mentioned as a starting point to researchers interested in this technology.

6.1. Models for Sensor Networks

Some efforts have bcen made to systematize common problems and main challenges. A good approach was made in [1] to clarify the problems per topics, there are many possible approaches, as it was referred, and the model needs to be refined. There are questions that still need to be answered:

How can we classify networks based on applications and how does this influence on the network model?

How can we design the network to be energy efficient? What type of routing is best suitable for WSNs? Does the network model depend on each application?

Is there a general approach for models? Is there the need to design different algorithms to different network models? Is it possible to make distributed pre-processing?

6.2. Benchmarking WSNs

As referred before, each research group presents different views of models for WSNs relating to different parts of the design space. As there is no accepted standard model the lack of a clear and uniform view makes it very hard to evaluate protocols and compare them to each other.

After researchers reach one or more model representation suitable for some or all application classes, appropriate metrics for evaluating strengths and weaknesses of protocols need to be defined.

The next step is to develop an automatic procedure of classifying protocols using metrics defined, comparing performance to the ideal protocol and to the trivial one.

Such a platform would benefit the complete community providing means to benchmark research results and therefore helping specialists improve their one work.

6.3. Error Detection and Correction algorithms

WSNs are about gathering and distributing data. Communications are often in noisy and dynamic environments. There is still open space for the development of error detection and correction algorithms to shield communication from errors.

6.4. Security

Security issues are always important when collecting and distributing data. Research in this field seams to derive from the ad hoc networks security solutions, still and as referred before, WSNs have specific problems that could introduce important security problems. There is clearly a technology gap that is open for researchers to fill.

6.5. Distributed processing and storage algorithms

WSNs are by definition distributed. The acquisition of data is done everywhere and deployed anywhere in the network. The resource constraints force the need to reduce the amount of communications and distribute data storage. Data acquired in a node could be pre-processed (filtered, compressed, aggregated, etc.) in that node to minimize transmissions, and processed as it flows through its path to the destination node. Some inspiration in peer-to-peer networks could be useful because of the similarities faced regarding distributed data storage problems. The need for distributed algorithms becomes even clearer as increasingly WSNs are becoming Wireless Sensor and Actuator Networks (WSANs). In cases where actuators have to react in conjunction to fulfill an objective there is the need to make the decision in a distributed way. The application of results from research in Social Networks could also be explored in what concerns decision making.

6.6. Routing Protocols

The purpose of WSNs is to harvest and distribute data. This has to be done in an efficient way especially as the deployment of such networks is usually made in difficult access places and the resources limit the network's lifetime. Good routing protocols are already in place although most of them assume static topologies and therefore some improvements could be useful.

A good approach would be a protocol that could predict the positioning of nodes by making a graph on the relative location of nodes to each other.

A good survey dedicated exclusively on routing protocols can be found in [33].

6.7. Transport protocol

At the time of this document some good transport protocols designed for WSNs are available, however there are still open issues regarding energy efficiency on transport protocols. One possible approach could be designing a protocol that reacts on energy, network topology and node location factors. Moreover the transport protocol should be able to provide transport of data between heterogeneous networks by allowing intermediate proxies. This would be useful both for WSNs and for traditional wireless networks.

6.8. Interface WSNs

Considerable amount of work has been done regarding WSN related issues yet interactions with the outside world and with other WSNs are still far from being completely explored.

One very interesting approach is made by the University of California at Berkley in the TinyDB [35] project. This approach faces the interface to WSNs as if they were a database. The main issue is that the SQL queries are not optimized for energy savings.

A possible approach is to design energy-efficient query language to take full advantage of the WSNs as referred in [36], [37], and [38]. Other possible approach is to integrate node location and network topology inputs in the database to allow appropriate routing of queries and information.

6.9. Network Reprogramming

Although operating systems already contemplate some aspects of network programming there is still space for research in this area. Some important issues on power-efficiency could still be improved and an enabling new concept of peer reprogramming could be explored. The idea behind this concept is very simple: Due to the heterogeneity of the networks and to the resource constraints of the program storage capabilities of the sensor nodes, although they might share the same hardware, some are specialized in specific tasks. In some cases it is interesting for a node to teach another node in range to do its work, when for example its battery is ending.

7 Conclusions

With the advances in processing power, sensor accuracy, miniaturization and production costs, WSNs will be used in a wider range of applications in the near future. Already deployed WSNs serve as a proof of concept, providing the means to monitor environmental data formerly impossible to collect due to the inaccessibility or characteristics of the environments where such physical phenomena take place.

Research on this field is very promising and has been attracting the scientific community for the last years. Some of the more active research areas are models for sensor networks, distributed processing, routing protocols, transport protocol and network reprogramming. Recent developments prove that there are many applications feasible with the available technology, assuring that in a near future WSNs will become commodities in what concerns sense and instrument the physical world. Furthermore WSNs are a factor of progress, enhancing productivity in sectors such as agriculture, transport and construction.

References

1. Kay Römer, Friedemann Mattern, "The Design Space of Wireless Sensor Networks", IEEE Wireless Communications, Vol. 11, No. 6, pp. 54-61, December 2004.
2. David Culler, Deborah Estrin, Mani Srivastava, "Overview of Sensor Networks ", Special Issue in Sensor Networks, IEEE Computer 37(8), Aug 2004: pp. 41-49.
3. Mainwaring et al., "Wireless Sensor Networks for Habitat Monitoring", WSNA, Atlanta, GA, USA, Sept. 2002.
4. P. Juang et al., "Energy-Efficient Computing for Wildlife Tracking: Design Tradeoffs and Early Experiences with ZebraNet", Proc. ASPLOS X, San Jose, CA, Oct. 2002.
5. K. Martinez et al., "GLACSWEB: A Sensor Web for Glaciers", Adjunct Proc. EWSN 2004, Berlin, Germany, Jan. 2004.
6. Z. Butler et al., "Networked Cows: Virtual Fences for Controlling Cows", WAMES 2004, Boston, MA, USA, June 2004.

7. W. Marshall et al., “Self-Organizing Sensor Networks”, UbiNet 2003, London, U.K., Sept. 2003.
8. “ARGO — Global Ocean Sensor Network”, http://www.argo.ucsd.edu, Jan. 2008
9. R. Beckwith, D. Teibel, and P. Bowen. “Pervasive Computing and Proactive Agriculture”, Adjunct Proc. PERVASIVE 2004, Vienna, Austria, Apr. 2004.
10. R. Riem-Vis, “Cold Chain Management Using an Ultra Low Power Wireless Sensor Network”, WAMES 2004, Boston, USA, June 2004.
11. F. Michahelles et al., “Applying Wearable Sensors to Avalanche Rescue”, Computers and Graphics, vol. 27, no. 6, 2003, pp. 839–47.
12. H. Baldus, K. Klabunde, and G. Muesch. “Reliable Setup of Medical Body-Sensor Networks”, Proc. EWSN 2004, Berlin, Germany, Jan. 2004.
13. Kappler and G. Riegel, “A Real-World, Simple Wireless Sensor Network for Monitoring Electrical Energy Consumption”, Proc. EWSN 2004, Berlin, Germany, Jan. 2004.
14. S. Antifakos, F. Michahelles, and B. Schiele, “Proactive Instructions for Furniture Assembly”, Proc. Ubicomp 2002, Gothenburg, Sweden, Sept. 2002.
15. The 29 Palms Experiment: Tracking Vehicles with a UAV-Delivered Sensor Network. http://robotics.eecs.berkeley.edu/~pister/29Palms0103, Jan 2008
16. W. M. Meriall et al., “Collaborative Networking Requirements for Unattended Ground Sensor Systems”, Proc. IEEE Aerospace Conf., Mar. 2003.
17. G. Simon, A. Ledezczi, and M. Maroti. “Sensor Network-Based Countersniper System”, Proc. SenSys, Baltimore, MD, USA, Nov. 2004.
18. G. Gonçalves, A. Sousa, J. Pinto, P. Lebres, J. Borges de Sousa, “Pilot Experiment of an Early Warning Fire Detection System”, 3rd European Workshop on Wireless Sensor Networks, Switzerland, February 2006.
19. Jan Rabaey, et al, “PicoRadio: Communication/Computation PicoNodes for Sensor Networks”, Technical Report, Electronics Research Laboratory, pp. 7-22, U.C. Berkeley, Dec. 2002.
20. Cha, H., Choi, S., Jung, I., Kim, H., Shin, H., Yoo, J., and Yoon, C., “RETOS: resilient, expandable, and threaded operating system for wireless sensor networks”, Proc. IPSN’07, Cambridge, Massachusetts, USA, April 2007, pp. 25 - 27
21. Bhatti, S., Carlson, J., Dai, H., Deng, J., Rose, J., Sheth, A., Shucker, B., Gruenwald, C., Torgerson, A., and Han, R. 2005, “MANTIS OS: an embedded multithreaded operating system for wireless micro sensor platforms” Mob. Netw. Appl. 10, 4 (Aug. 2005)
22. TinyOs, http://www.tinyos.net, Jan 2008
23. SOS, http://nesl.ee.ucla.edu/projects/sos, Jan 2008
24. Contiki, http://www.sics.se/contiki, Jan 2008
25. Vinícius C. de Almeida, Luiz Filipe M. Vieira, Breno A. D. Vitorino, Marcos Augusto M. Vieira, Diógenes C. da Silva Jr., Antônio O. Fernandes, Claudionor Nunes Coelho Jr., "Sistema Operacional YATOS para Redes de Sensores sem Fio", Workshop de Sistemas Operacionais 2004, Salvador/BA, Agosto 2004
26. P. Levis and D. Culler. Mate, “A tiny virtual machine for sensor networks. In International Conference on Architectural Support for Programming Languages and Operating Systems”, San Jose, CA, USA, Oct. 2002
27. Y. Wang, S. H. Cho, C. G. Sodini, and A. P. Chandrakasan, “Energy Efficient Modulation and MAC for Asymmetric RF Microsensor Systems”, *Intl. Symp. On Low Power Electronics and Design (ISLPED ’01)*, August 2001, pp. 96–99
28. Schurgers, O. Aberthorne, and M. B. Srivastava, “Modulation Scaling for Energy Aware Communication Systems”, In *Intl. Symp. on Low Power Electronics and Design (ISLPED ’01)*, August 2001, pp. 96–99
29. Holger Karl, Andreas Willig: "*A Short Survey Of Wireless Sensor Networks*", Technical Report, Telecommunication Networks Group, Technische Universitt Berlin, 2003

30. J. Elson and K. Römer, "Wireless sensor networks: a new regime for time synchronization", *ACM SIGCOMM Computer Communication Review*, 2003, pp. 149–154
31. Pavlos Papageorgiou, "Literature Survey on Wireless Sensor Networks", July 2003
32. C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva. Directed Di fusion for Wireless Sensor Networking. *IEEE Trans. on Networking*, February 2003.
33. Chieh-Yih Wan, A.T. Campbell, and L. Krishnamurthy, "PSFQ: a reliable transport protocol for wireless sensor networks" ACM International Workshop on Wireless Sensor Networks and Applications, 2002
34. Jamal N. Al-Karaki and Ahmed E. Kamal, "Routing techniques in wireless sensor networks: a survey", IEEE Wireless Communications Journal, December 2004
35. Tinydb, http://telegraph.cs.berkeley.edu/tinydb/, Jan 2008
36. R. Avnur and J. M. Hellerstein, "Eddies : Continuously Adaptive Query Processing", *Proc. 2000 ACM SIGMOD Intl. Conf. on Management of Data*, Dallas, TX, May 2000, pp. 261–272
37. H. Gupta, S. Das, and Q. Gu., "Connected Sensor Cover: Self-Organization of Sensor Networks for Efficient Query Execution" *Proc. 4th ACM Intl. Symp. on Mobile Ad Hoc Networking and Computing (MobiHoc)*, Annapolis, MD, 2003
38. N. Sadagopan, B. Krishnamachari, and A.Helmy, "The ACQUIRE mechanism for efficient querying in sensor networks" *Proc. 1st IEEE Intl. Workshop on Sensor Network Protocols and Applications (SNPA)*, Anchorage, AK, May 2003.

Evaluation and comparison of automatic methods to identify health queries

Carla Teixeira Lopes

Faculdade de Engenharia da Universidade do Porto, Portugal
carla.lopes@fe.up.pt

Abstract. The use of the Web to find health information is a common practice nowadays. The improvement of Health Information Retrieval depends on studies that, frequently, require the identification of health-related queries. Being usually done by human assessors, this identification may turn out to be inefficient and even impracticable in some cases. To overcome this problem we propose, analyze and compare automatic methods to identify health-related queries. One type of methods uses health vocabularies and the other analyses the co-ocurrence of query terms with the word "health" in web page results. Our goal is to compare the two different strategies of automatic classification, to compare several variants in each strategy and to verify if its performance is enough to be executed without human intervention. The evaluation was done comparing the automatic classification with the classification made by a team of ten human assessors, in a pool of 20,000 queries. The use of Yahoo! to calculate the co-occurrence rate at a threshold value of 0,5 was the method with best trade-off between sensibility (73%) and specificity (79%).

Key words: Health Information Retrieval, Web Information Retrieval, Health Queries, Automatic Classifiers, Medical Vocabularies.

1 Introduction

The use of the Web to find health information has become a common practice nowadays. According to a Pew Internet & American Life Project 2006 report [5], eight in ten american users go online for health information and the typical health information session starts at a search engine. 74% of all health seekers also said that health search allowed them to make more appropriate health decisions. Jupiter Research [7] reached similar conclusions, founding that 71% of online consumers use search engines to find health-related information.

The large proliferation of health information Web search and the impact it may have on people's life accent the importance of studies in Health Information Retrieval. Usually, in these studies, one of the first steps is the identification of health-related queries in a pool of queries. A health query is a query that intends to retrieve health-related information and is related to a health information need. The most frequent classification method (as happens in [11]) involves human intervention making it a slow process and requiring the availability of one or more human classifiers. In some cases, the huge volume of queries may even make this classification impracticable. For these reasons, automatic methods of health queries identification could be a useful tool.

Eysenbach and Kohler [3] proposed a method to automatically classify search strings as health-related based on the proportion of pages on the Web containing the search string plus the word "health" and the number of pages containing only the search string. Besides this method, no other automatic mechanism with this goal was found reported in the literature. The nearest, but broader, topic is generic automatic query classification (a good state of the art of this area is done in the paper of Beitzel and Lewis [1]). Yet, as our goal is restricted to the health domain, we believe some simpler and more targeted strategies may be developed.

Our goal with this research is to propose new automatic methods to detect health queries and to compare them with three variants of the one described by Eysenbach and Kohler [3]. Based on the knowledge that most health queries contain terms that can be mapped to health/medical vocabularies [8, 9], we have decided to use this type of vocabularies to detect the presence of health terms in queries through several different strategies.

The vocabulary chosen is the Consumer Health Vocabulary (CHV) developed as an open source and collaborative initiative to complement the Unified Medical Language System (UMLS). This vocabulary links everyday phrases about health and technical terms used by professionals, aiming to bridge the communication gap between consumers and professionals. It is available for download on the CHV website [12] as several files. We opted for CHV instead of UMLS because the first focuses on concepts employed by consumers in health communications. As we want to analyze queries submitted to generic search engines, it's probable that most queries are submitted by non-health experts.

In short, we want to evaluate the performance of the several methods, to compare the method proposed by Eysenbach and Kohler [3] with methods that use health vocabularies and to compare the different variants of each type of methods.

The next Section of this paper describes the 14 automatic methods we propose and want to compare. This section also describes the processes of implementation and evaluation of the described methods. In Section 3 are presented the results gathered after the execution of the several methods. In Section 4 are discussed the previously presented results. Finally, conclusions are presented together with lines of future work in Section 5.

2 Methods

2.1 Automatic methods to detect health-related queries

We propose 14 automatic methods to detect health-related queries that can be grouped in two distinct categories. A first category (CHV methods), with 11 different methods, uses the CHV. The second category (co-ocurrence methods) contains 3 methods based on the idea that health-related terms should co-occur with the word "health" more often than non-health terms, as proposed by Eysenbach and Kohler [3].

While CHV methods produce a discrete class label indicating only the predicted class (health or non-health) of the query, co-occurence methods produce a continuous output to which different thresholds may be applied to predict a query's class.

CHV methods The CHV can be downloaded in 4 different flat files: concepts terms, ngrams, stop concepts and incorrect mappings. The first file contains concepts and associated terms. Each concept may have many terms and each term is listed in a separate row. The ngrams file lists terms not mapped to the UMLS but associated to medical concepts. The stop concepts file lists concepts excluded from the CHV. The last file lists incorrect combinations of concepts and terms.

This category's methods differ on the subset of the terms used to classify the queries. The presence of one term in a query is sufficient to classify it as a health query. The necessity of several methods emerged from the large size of the initial concepts terms flat file (158,908 terms) and also from its contents (against initials expectations, it included several terms not specifically health-related — p.e.: rail, driver — and even stop-words). The first step involved the removal of stop-words and the replacement of characters that could be misunderstood in regular expressions (used later to parse the files). Then, 11 variants with different lists of terms (all after stop-words removal) were defined: CHV1 (all terms), CHV2 (terms associated with the 200 most frequent concepts), CHV3 (terms associated with the 400 most frequent concepts), CHV4 (terms associated with the 600 most frequent concepts), CHV5 (terms associated with the 800 most frequent concepts), CHV6 (terms associated with the 1,000 most frequent concepts), CHV7 (UMLS preferred terms — with the field `UMLS_preferred_name` set to "yes"), CHV8 (CHV preferred terms — with the field `CHV_preferred_name` set to "yes"), CHV9 (UMLS or CHV preferred terms), CHV10 (6,000 more frequent terms obtained directly from the website — 5,898 terms after stop-words removal) and CHV11 (10,000 more frequent terms obtained directly from the website — 9,872 terms after stop-words removal).

The criteria behind these 11 variants were defined empirically in an iterative process fed by the data analysis of the variants defined at that moment. Different results could have led to different criteria (e.g. use more terms if the previous results were showing performance improvements).

Co-ocurrence methods As mentioned previously, these methods are based on the idea that health-related terms should co-occur together with the word "health" more often than non-health terms. For each query (Q) in the pool, two queries were submitted to a search engine: one (Q1) with the terms of the query Q and another (Q2) with the terms of Q plus the word "health". The co-occurence rate (cooc) of Q is calculated by the proportion of the total number of results of Q2 and the total number of results of Q1:

$$cooc(Q) = \frac{\#results(terms_Q \cap health)}{\#results(terms_Q)}$$

where $terms_Q$ is the set of terms that compose the query Q. If $\#results(terms_Q) = 0$, $cooc(Q) = 0$.

This proportion is an indicator of the relatedness of the query Q to the health domain because it represents the frequency of occurrence of Q's search terms and the word "health" in web pages. For example, the query 'diabetes symptoms' has a co-occurence rate of $\frac{478000}{929000} = 0,51$ and the query 'Pavarotti' has a co-occurence rate of $\frac{359000}{6440000} = 0,06$.

In the work of Eysenbach and Kohler [3], where this method was proposed, Google was the used search engine. Here, we have used Google and Yahoo! to determine the number of results and we have also proposed a variant of these methods that combines both search engines' number of results. We have, therefore, implemented 3 methods with different co-ocurrence rates:

$$G_{cooc_Q} = \frac{\#google(terms_Q \cap health)}{\#google(terms_Q)} \; Y_{cooc_Q} = \frac{\#yahoo!(terms_Q \cap health)}{\#yahoo!(terms_Q)}$$

$$Y + G_{cooc_Q} = \frac{\#google(terms_Q \cap health) + \#yahoo!(terms_Q \cap health)}{\#google(terms_Q) + \#yahoo!(terms_Q)}$$

The differences detected in the number of results of both search engines (also stated in [2]) took us to combine the number of results returned by the two search engines in the third method.

After the calculation of the co-occurence rate, this value was compared with several thresholds (0; 0,05; 0,1; 0,15; 0,2; ...; 0,95; 1). In each comparison, if the co-occurence rate was larger than or equal to the threshold, the query was considered to be a health-related query at that threshold.

2.2 Implementation

To evaluate the methods described previously we've used a collection of 20,000 web queries, randomly sampled from AOL Search in the Fall of 2004. This collection was used by Beitzel and Lewis in a research project [1] where queries were classified into 20 topical categories by a team of approximately ten human assessors. One of the topical categories is health, where 1,197 queries are included.

In CHV methods two other datasets were also used: one text file with a list of stop-words provided by the University of Glasgow [10] and a tab separated value (tsv) file with the CHV Concepts & Terms Flat File available at CHV website [12]. The first dataset file has one stop-word per line and the second dataset file has one line per each term and associated information.

Several Perl scripts were developed to implement the methods described previously. In each CHV method we've used two Perl scripts: one (`generateTermsList.pl`) that generates a subset of health terms and another one (similar in all CHV methods) that classifies queries (see Figure 1). The `generateTermsList.pl` also removes stop-words and replaces special characters that may be misunderstood by regular expressions. The `classifyQueries.pl` simply checks if any of each query's terms is present in the terms list. If present, queries are classified as health-related.

In the co-occurence methods, we've developed scripts (one for each search engine) to automatically get the number of results returned for each query in Google and Yahoo (see Figure 2) through each search engine's API. Each of these scripts was then used by another script (`classifyQueries.pl`) that reads the queries collection file line by line, asks the `numberofResults.pl` for the number of results of two queries (the query read and the query plus the word "health") and writes this information in another file.

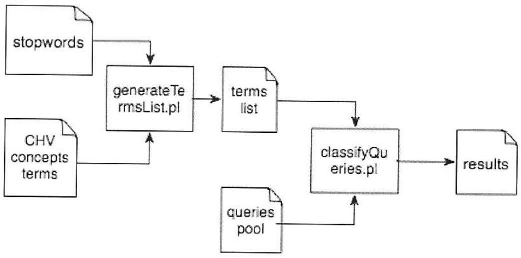


Fig. 1. CHV methods global architecture — dataset files and Perl scripts

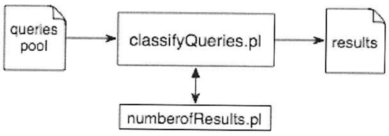


Fig. 2. Co-occurence methods global architecture — dataset files and Perl scripts

2.3 Evaluation

The evaluation of each method was done through the comparison of the classification made by the team of human assessors and the classification of each method. In the CHV methods the classification is immediately delivered after the execution of the described scripts. In the co-occurence methods, the classification only occurs after the calculation of the cooc rate and its comparison with each threshold. The best thresholds are determined after the analysis of all collected data.

3 Results

For each method, measures like sensitivity, specificity and accuracy were calculated. These can be expressed in terms of probabilities of the following events: HC_H (query is classified as health-related in a human classification), HC_NH (query is classified as non-health-related in a human classification), AC_H (query is classified as health-related in an automatic classification) and AC_NH (query is classified as non-health-related in an automatic classification).

Sensitivity (SEN) is expressed as the conditional probability of having an automatic classification of health-related, given that the query was classified as health-related by a human: $P(AC\_H|HC\_H)$.

Specificity (SPC) is expressed as the conditional probability of having an automatic classification of non-health-related when the query was classified as non-health-related by a human: $P(AC\_NH|HC\_NH)$.

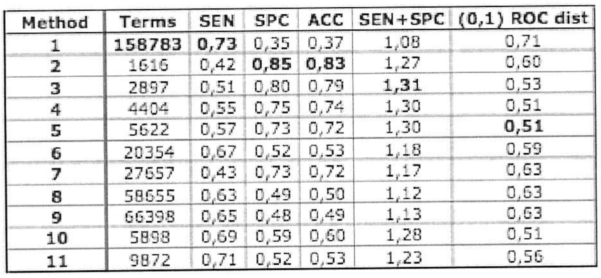
Accuracy (ACC) is the tax of correct classifications (either as health-related or as non-health-related) and is expressed by: $\frac{P(AC\_H \cap HC\_H)+P(AC\_NH \cap HC\_NH)}{P(HC\_H)+P(HC\_NH)}$

Besides the calculation of these measures, two Receiver Operating Characteristics (ROC) graphs for comparing the several discrete classifiers methods and the several continuous classifiers methods were also drawn. A ROC graph is a two-dimensional graph in which sensibility is plotted on the Y axis and the false positive rate (1-specificity) is plotted on the X axis. It is a technique that depicts relative tradeoffs between benefits (true positives) and costs (false positives), being useful for visualizing, organizing and selecting classifiers based on their performance [4].

3.1 CHV methods

Table of Figure 3 presents, for each CHV method, the number of terms used in the classification method (`Terms`), sensibility (`SEN`), specificity (`SPC`), accuracy (`ACC`), sum of sensibility and specificity (`SEN + SPC`) and the distance of each method to the optimal point in ROC space (`(0,1) ROC dist`). Each column's greatest value is highlighted in bold (except the last column where the minimum value is the indicator of a best performance). The inclusion of the `SEN + SPC` value doesn't intend to be an indicator of the best method because sensibility may be preferred over specificity in some cases and vice-versa. It is just a helpful measure to see which method has the greatest overall sum of sensibility and specificity.

Fig. 3. Number of terms, Sensibility, Specificity, Accuracy and other Measures for CHV methods



Method	Terms	SEN	SPC	ACC	SEN+SPC	(0,1) ROC dist
1	**158783**	**0,73**	0,35	0,37	1,08	0,71
2	1616	0,42	**0,85**	**0,83**	1,27	0,60
3	2897	0,51	0,80	0,79	**1,31**	0,53
4	4404	0,55	0,75	0,74	1,30	0,51
5	5622	0,57	0,73	0,72	1,30	**0,51**
6	20354	0,67	0,52	0,53	1,18	0,59
7	27657	0,43	0,73	0,72	1,17	0,63
8	58655	0,63	0,49	0,50	1,12	0,63
9	66398	0,65	0,48	0,49	1,13	0,63
10	5898	0,69	0,59	0,60	1,28	0,51
11	9872	0,71	0,52	0,53	1,23	0,56

To aid the comparison of the several methods a ROC graph was drawn (Figure 4) with each method represented by a different point in the ROC space.

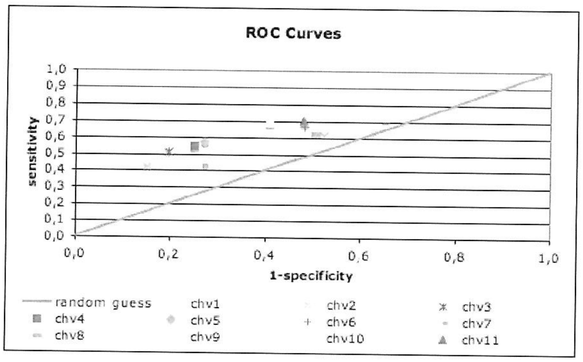


Fig. 4. CHV methods ROC graph

3.2 Co-occurence methods

As mentioned in Section 2, co-occurence methods are continuous classifiers because they produce a continuous output (co-occurence rate) that may be considered an estimate of queries health-relatedness probability. Each method has its own co-occurence rate with the distribution presented in the histograms of the Figures 5, 6 and 7. In these histograms, only co-occurence rates between 0 and 1 are represented. In the three methods were detected queries with co-occurence rates greater than 1: Google has 3,174, Yahoo! has 693 and GoogleYahoo! has 1,417 queries. Google has a co-occurence average of 0.45, Yahoo! of 0.32 and GoogleYahoo! of 0.39. The standard deviation is also greater in Google (0.305), followed by Google Yahoo! (0.243) and Yahoo! (0.228).

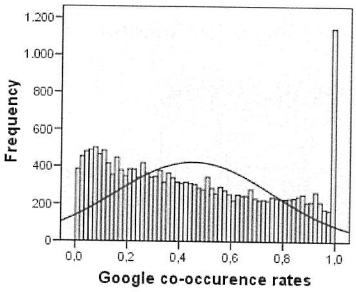


Fig. 5. Google co-occurence rate histogram

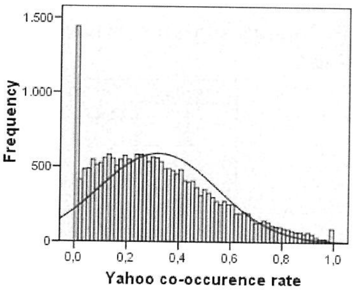


Fig. 6. Yahoo! co-occurence rate histogram

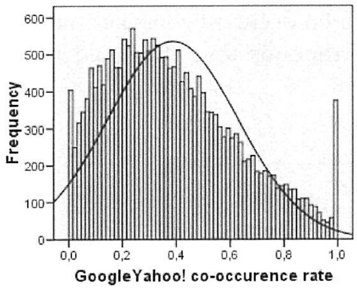


Fig. 7. GoogleYahoo! co-occurence rate histogram

To predict each query health-relatedness, this continuous output was then compared with different thresholds (ranging from 0 to 1). Sensibility, specificity, accuracy, sum of sensibility and specificity and the distance of each method to the optimal point in ROC space, for the several thresholds in each method, are presented in Table of Figure 8. Each column's greatest value is highlighted in bold (except the last column where the minimum value is the indicator of a best performance). Just as in the CHV methods, the sum of sensibility and specificity does not intend to be a single evaluation measure of the optimal threshold.

Fig. 8. Sensibility, Specificity, Accuracy and other Measures for Co-occurence methods

Threshold	SEN			SPC			ACC			SEN+SPC			(0,1) ROC dist		
	Yahoo!	Google	Y+G	Yahoo!	Google	Y+G	Yahoo!	Google	Y+G	Yahoo!	Google	Y+G	Yahoo!	Google	Y+G
1	0,07	0,21	0,12	**0,97**	**0,82**	**0,93**	**0,92**	**0,78**	**0,88**	1,04	1,02	1,05	0,93	0,82	0,88
0,95	0,08	0,28	0,15	0,97	0,80	0,92	0,92	0,77	0,88	1,05	1,08	1,07	0,92	0,74	0,85
0,9	0,13	0,37	0,21	0,96	0,77	0,91	0,92	0,75	0,87	1,09	1,14	1,13	0,87	0,67	0,79
0,85	0,19	0,43	0,29	0,96	0,74	0,90	0,91	0,72	0,87	1,15	1,17	1,19	0,81	0,63	0,72
0,8	0,27	0,49	0,36	0,95	0,71	0,88	0,91	0,70	0,85	1,22	1,20	1,24	0,73	0,59	0,65
0,75	0,36	0,54	0,43	0,93	0,68	0,86	0,90	0,67	0,84	1,29	1,22	1,29	0,65	0,56	0,59
0,7	0,44	0,58	0,51	0,92	0,65	0,84	0,89	0,65	0,82	1,36	1,24	1,35	0,56	0,54	0,52
0,65	0,53	0,63	0,58	0,90	0,62	0,81	0,88	0,62	0,80	1,42	1,25	1,39	0,48	0,53	0,46
0,6	0,60	0,68	0,65	0,87	0,59	0,77	0,85	0,59	0,77	1,47	1,27	1,42	0,42	**0,52**	0,42
0,55	0,67	0,72	0,70	0,83	0,55	0,73	0,82	0,56	0,73	1,50	**1,27**	1,43	0,37	0,53	0,40
0,5	0,73	0,75	0,76	0,79	0,51	0,68	0,79	0,53	0,69	**1,52**	1,27	**1,44**	**0,34**	0,55	**0,40**
0,45	0,77	0,79	0,80	0,74	0,48	0,62	0,74	0,49	0,63	1,50	1,26	1,42	0,35	0,57	0,43
0,4	0,81	0,81	0,84	0,67	0,43	0,56	0,68	0,45	0,57	1,48	1,24	1,40	0,38	0,60	0,47
0,35	0,85	0,85	0,88	0,60	0,39	0,48	0,62	0,41	0,51	1,45	1,24	1,36	0,42	0,63	0,53
0,3	0,88	0,87	0,92	0,52	0,34	0,41	0,54	0,37	0,44	1,40	1,21	1,32	0,49	0,67	0,60
0,25	0,90	0,89	0,93	0,44	0,29	0,33	0,46	0,32	0,36	1,34	1,18	1,26	0,57	0,72	0,67
0,2	0,92	0,91	0,94	0,36	0,24	0,25	0,39	0,27	0,29	1,28	1,15	1,19	0,65	0,77	0,75
0,15	0,93	0,94	0,96	0,27	0,18	0,18	0,31	0,22	0,22	1,20	1,12	1,14	0,73	0,82	0,82
0,1	0,95	0,97	0,98	0,19	0,13	0,11	0,23	0,17	0,15	1,14	1,09	1,08	0,81	0,87	0,89
0,05	0,96	0,99	0,99	0,11	0,06	0,05	0,16	0,11	0,10	1,07	1,05	1,04	0,89	0,94	0,95
0	**1,00**	**1,00**	**1,00**	0,00	0,00	0,00	0,05	0,05	0,05	1,00	1,00	1,00	1,00	1,00	1,00

The ROC curves for each co-occurence method are represented in Figure 9. Each point in the curve corresponds to a threshold value, starting on 1 at the left side of the graph.

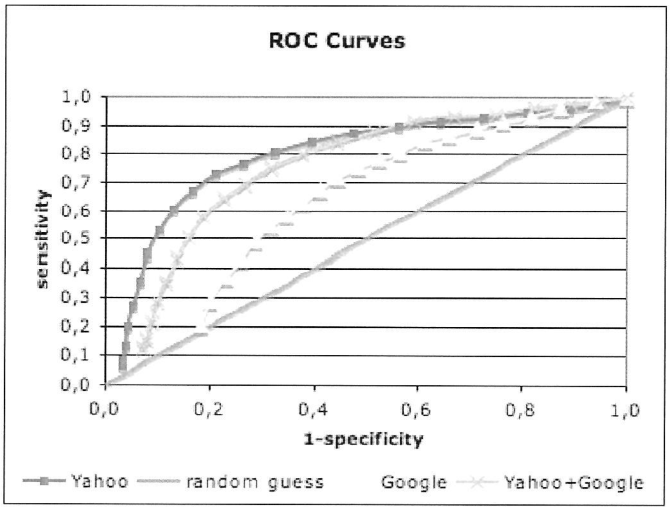


Fig. 9. Co-occurence methods ROC graph

4 Discussion

In Figure 4 it is possible to see that all CHV methods are better than a random guess (represented by a diagonal line) as they are located above it (in ROC graphs, the point (0,1) represents a perfect classification, so better performances are closer to this point). Yet, no method has reached the results initially expected. In fact, the best methods, as can be seen in Figure 4, are CHV2, CHV3, CHV4 and CHV5 (methods that use the list of terms of the 200, 400, 600 and 800 most frequent concepts) and their sensibility doesn't exceed 57%. The specificity and accuracy is greater in CHV2 but sensitivity has a low value (42%) in this method. CHV3 is the method with the larger sum of sensibility (51%) and specificity (80%). CHV5 is the closest to the point optimal point in ROC Space (minimum distance to (0,1)).

We can also see that the relation between the number of health terms and sensibility is not directly proportional. For example, CHV10 has less terms but higher sensibility and specificity than CHV6. This means there are terms more related to the health context than others and that the performance of this type of methods could be improved by a careful selection of terms. Generally, all CHV methods present a low sensibility.

To begin the analysis of co-occurence methods we would like to mention the existence of co-occurence rates greater than 1. Theoretically, these values shouldn't exist because the default operator between terms in both search engines (Google and Yahoo!) is the logic "AND", what means that all terms in a query without operators should appear in results' pages. In theory, adding terms should only result in a maintenance or decrease of the number of results. The number of queries in this situation is larger in Google than in Yahoo! (3,174 against 693). The query "go carts" is one example (with 3,230,000 results in Google) and the query "go carts health" (with 8,470,000 results in Google). This may be explained by the fact that the number of results returned by search engines is usually just a estimate. Google Help Center [6] explains that not providing the exact count allows them to return search results faster. Yet, the high number of these cases is still surprising.

The histograms of Figures 5, 6 and 7 show that the GoogleYahoo! co-ocurrence rate is the closest to the Normal distribution, followed by the Yahoo! co-ocurrence rate. It's also possible to verify the existence of a strange peak at the right side of the Google histogram and at the left side of the Yahoo! histogram. The higher frequency of values near 1 in Google histogram shows that, in a large number of queries' return pages, the term "health" co-occurs with the other terms of the query. The peak in Yahoo! shows that a large number of queries return 0 results.

Analyzing the measures of Table of Figure 8 it's possible to verify that, as expected, sensibility is 1 at a threshold of 0 (co-occurence rates are always bigger than 0 making all queries to be classified as health-related). Naturally, at this same threshold, specificity is 0 (since there aren't queries classified as non-health related). Mainly due to high specificity values at threshold of 1, accuracy is also maximized at this threshold. The sum of sensibility and specificity measure has the best value at a threshold of 0.5 of the Yahoo! method (with 73% of sensibility and 79% of specificity) just as the Yahoo!Google method. The Google method has its best sensibility+specificity value at a 0.55 threshold. The analysis of the distance to the optimal point in the ROC Space keeps the threshold of 0.5 as the best of the Yahoo! method. Using Google, the best threshold value changes to 0.6 in the analysis of this last measure.

In the ROC graph of Figure 9 it's clear the dominance of Yahoo! over Google (always above it). In this graph it is also possible to detect the closer points of each method to the point (0,1).

The idea of joining the estimates of Yahoo! and Google into the third method hasn't produced the expected results (improvements when compared to the two other methods). As can be seen in Figure 9 and Table of Figure 8, the Yahoo!Google method has an intermediate performance, being probably better than Google due to Yahoo! performance.

To test if the differences between Yahoo! (at 0.5) and Google (at 0.55 and 0.6) are significant two McNemar tests were applied: one between Yahoo! and Google (0.55) and other between Yahoo! and Google (0.6). P-value was 0 in both tests what means the differences in proportions between the best of Yahoo! methods and the two better Google methods are significant. This result encourages the use of Yahoo! to the co-occurence methods.

Google results in this sample of 20,000 queries are different from the results of Eysenbach and Kohler [3]. In their work, the threshold of 35% was considered an optimal trade-off between sensitivity (85.2%) and specificity (80.4%). The sample used in their study was composed of 2985 queries. Comparatively, our study had worser sensitivity values (68% or 72%), specificity values (59% or 55%) and different optimal threshold values (0.6 or 0.55). The larger sample used in our study make us believe our results are a better portray of reality.

We would like to emphasize that the methods indicated as optimal may be discarded when compared to others if sensibility is preferable to accuracy or vice-versa. For example, in a situation where we want to reduce to filter the number of queries to be categorized by a human assessor without the risk to eliminate a large number of health-related queries, it is preferable to have good sensibility instead of specificity.

5 Conclusions and Future Work

We evaluated several variants of two type of classifiers: a discrete one, proposed by the author, that uses terms of health vocabularies and a continuous one, proposed by Eysenbach and Kohler [3], that evaluates the query relatedness to health through the co-ocurrence rate of query terms with the word "health" in search engines' results.

While Yahoo! demonstrated a better performance than Google in the co-occurence methods, its results were still worser than Eysenbach and Kohler's results. In their work, at a threshold of 35%, sensibility was 85.2% and specificity was 80.4%, while in our Yahoo! method, at a threshold of 0.5, sensibility was 73% and specificity was 79%. We think our results depict reality more accurately since our sample of queries is much larger (20,000 against 2,985 queries).

None of the methods that used subsets of terms of health vocabularies behaved as well as the Yahoo! method. Yet some of CHV methods behaved better than the Google method (CHV3, CHV4 and CHV5 had better or similar performance than the Google method).

A manual definition of a term list might improve CHV methods. Through the behavior's analysis of the best CHV methods by a human assessor it may be possible to eliminate some of the terms that produce false positives and add some terms that could reduce the number of false negatives. We also aim to define and evaluate this type of methods using the UMLS vocabulary instead of the CHV. Another line of future work in this type of methods involves the definition of a continuous output based on the number of health terms presented in the query (the methods presented in this paper only detect the presence or non-presence of health terms).

We also intend to evaluate co-occurence methods in Portuguese queries, analyzing the co-occurence rate with the "health" Portuguese word. If results in Portuguese are similar to the English results, this method has the advantage of an easier application to other languages (while the vocabularies methods require the definition of foreign languages' lists of terms). It could also be interesting to analyze the co-occurence rate with terms different from "health" or even a set of terms separated by the OR logical operator.

A specific evaluation of each query health-relatedness by a health specialist would also increase the correctness of the several methods' performance evaluation. In fact, some human classifications of health queries used in the dataset are dubious (e.g.: "devils club" and "regedit").

The application of these methods on other datasets would also allow to prove the validity of these results.

References

1. S. Beitzel, E. Jensen, D. Lewis, A. Chowdhury, A. Kolcz, and O. Frieder. Improving Automatic Query Classification via Semi-supervised Learning. In *The Fifth IEEE International Conference on Data Mining*, New Orleans, Louisiana, U.S.A., November 2005.
2. Ionut Alex Chitu. Google Finds Less Search Results. Available from: `http://googlesystem.blogspot.com/2007/12/google-finds-less-search-results.html` [accessed 28 December, 2007].
3. G. Eysenbach and Ch. Kohler. What is the prevalence of health-related searches on the World Wide Web? Qualitative and quantitative analysis of search engine queries on the Internet. In *AMIA 2003 Symposium Proceedings*, pages 225–230, 2003.
4. Tom Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, (27):861–874, 2006.
5. Susannah Fox. Online Health Search 2006. Technical report, Pew Internet & American Life Project, 2006.
6. Google. How does Google calculate the number of results? Available from: `http://www.google.com/support/webmasters/bin/answer.py?hl=en&answer=70920` [accessed 31 December 2007].
7. JupiterResearch. JupiterResearch Finds Strong Consumer Demand and Market Opportunity for Health Search Engines. Available from: `http://www.jupiterresearch.com/bin/item.pl/press:press_release/2006/id=06.07.17-health_search.html` [accessed 2th January 2008].
8. Alexa T. McCray, Russell F. Loane, Allen C. Browne, and Anantha K. Bangalore. Terminology Issues in User Access to Web-based Medical Information. In *AMIA 1998*, 1998.
9. Zeng QT, Crowell J, Plovnick RM, Kim E, Ngo L, and Dibble E. Assisting Consumer Health Information Retrieval with Query Recommendations. *J Am Med Inform Assoc*, 13(1):80–90, Jan-Feb 2006.
10. Mark Sanderson. Stop words list. Available from: `http://www.dcs.gla.ac.uk/idom/ir_resources/linguistic_utils/stop_words` [accessed 30 December 2007].
11. Amanda Spink, Yin Yang, Jim Jansenn, Pirrko Nykanen, Daniel P. Lorence, Seda Ozmutlu, and H. Cenk Ozmutlu. A study of medical and health queries to web search engines. *Health Information and Libraries Journal*, (21):44–51, 2004.
12. Qing T. Zeng. Consumer Health Vocabulary Initiative. Available from: `http://www.consumerhealthvocab.org/` [accessed 27th December 2007].

Calibration Agent for Ecological Simulations: A Metaheuristic Approach

Pedro Valente[1,2], António Pereira[1,2], Luís Paulo Reis[1,2]

[1] LIACC – Laboratório de Inteligência Artificial e Ciência de Computadores
[2] FEUP – Faculdade de Engenharia da Universidade do Porto
Rua Dr. Roberto Frias s/n, 4200-465 Porto, Portugal
pedro.valente@fe.up.pt

Abstract. This paper presents an approach to the calibration of ecological models, using intelligent agents with learning skills and optimization techniques. Model calibration, in complex ecological simulations is typically performed by comparing observed with predicted data and it reveals as a key phase in the modeling process. It is an interactive process, because after each simulation, the agent acquires more information about variables inter-relations and can predict the importance of parameters into variables results. Agents may be seen, in this context, as self-learning tools that simulate the learning process of the modeler about the simulated system. As in common Metaheuristics, this self-learning process, initially involves analyzing the problem and verifying its inter-relationships. The next stage is the learning process to improve this knowledge using optimization algorithms like Hill-Climbing, Simulated Annealing and Genetic Algorithms. The process ends, when convergence criteria are obtained and thus, a suitable calibration is achieved. Simple experiments have been performed to validate the approach.

Keywords: Ecological Modeling, Calibration, Intelligent Agents, Simulation Models, Uncertainty Analysis, Metaheuristics.

1 Introduction

The rapid progress achieved in computers hardware and software development in the last decades, have exponentially increased the usage of mathematical models across almost all fields of science, and simulation is now widely used to test or predict researchers' theories. This is particularly relevant in the fields of physical, chemical, biological, ecological and environmental sciences, health and weather forecast domains.

Models are simplified views of processes and are used to solve scientific or management purposes. Modelers try to translate the actual knowledge about system processes, formulated in mathematical equations and components' relationships, focusing in the processes the researcher is interested in and omitting some or many irrelevant details that aren't important for the problem in consideration [6]. However, these omitted details may have a strong influence on the predicted results produced by the model [16].

Models are intensively used in theoretical and applied Ecology. Simulation models of complex ecological processes are increasingly constructed for use in both the development of ecological theory and the analysis of environmental questions. Such models can never be validated due to the limited observation of system dynamics [13]. They can, however, be assessed to investigate deficiencies in the relationships they define between ecological theory, model structure, and assessment data [18][16].

This study is related with ecological models of aquatic systems, for two main reasons: first, the diversity of components involved (like physical processes include flow and circulation patterns, water temperature, settling of *planktonic* organisms, among others), leaving to a complex ecological models, because of its interactions and dependencies. The second reason, the existence of a real simulator for coastal ecosystems – **EcoDynamo** [8], that permits to explore ecological models, for different point of view or interest (e. g. fishing, tourism, aquaculture, harbor activities, sports, etc.).

In the last century, human population migrates intensively from inland to coastal boundaries and, nowadays, 60% of the world lives within 60km from the sea. So the correct management of the coastal zones is very important for the environment balance, and sustainable development [4].

Computer Simulation is a powerful tool in evaluating complex systems, like Coastal ecosystems. These evaluations are usually in the form of responses to "what if" questions. Practical questions, however, are often of "how to" nature. "What if" questions demand answers on certain performance measures for a given set of values for the decisions variables of the system. "How to" questions, on the other hand, seek optimum values for the decision variables of the system so that a given response or a vector of responses are maximized or minimized [2][19].

Using simulation as an aid for optimization presents several specific challenges. Some of these issues are those involved in optimization of any complex and highly nonlinear function. Others are more specifically related to the special nature of simulation modeling. Simply acknowledged, a simulation optimization problem is an optimization problem where the value of the objective function (objective functions, in case of a multi-criteria problem) and/or some constrains, can only be evaluated by computer simulation, and its validity made by compare it to real data. However, these functions are only implicit functions of decision parameters of the system. In sum, these functions are often stochastic[1] in nature as well.

Considered these characteristics in mind, for example, the objective function(s) and these constraints are stochastic functions of deterministic decision variables. These leave a major problem in estimation of even approximate local derivatives. Furthermore, this work against even using complete enumeration because based on just one observation at each point the best decision point cannot be determined. This is a generic non-linear programming problem.

However, advantages in using simulation optimization, for example, in stochastic systems, like ecological ones, the variance of the response is controllable by various output analysis techniques. Other main strength in using optimization techniques,

[1] Stochastic - A process with an indeterminate or random element as opposed to a deterministic process that has no random element.

reflect the constant change of objective function or constraints from one interaction to another to reflect alternative designs for the systems [19][13].

Those optimization techniques or Metaheuristics can be more intuitive, if previous, by observing simulated model runs, we learn how variables interact, and the sensitivity of tune parameters values. In this case, optimization techniques don't test in all space available, but in sub-space of pre-known validation [20].

The paper is organized as follows: the next section describes in more detail the problem in analysis; section 3, introduces and presents the key features of some known Metaheuristics: hill-climbing, Simulated Annealing and Genetic Algorithms; section 4 presents the architecture of the multi-agent simulation system, and its components; section 5 focus into Calibration Agent and it's methodology applied to an ecological model; the paper concludes with some conclusion and analysis of the project current state and pointers to future work.

2 Problem Statement

One of the problems related in simulated modeling is the lack of fitness in results produced and the real data sets, because a model represents a wider view of reality, in this particularly case, an ecosystem. All models are translated by mathematical formulas, in which main variables are represented. The validation formulas process, are made by specialist, whose sensibility and comprehension on ecological systems, result into better fit to real ones.

The formula validation process made by specialist, basically consist into change parameters values, and compare the result variables with observed data. It is a methodic process of recombination values into optimal results.

When models became complex, with various formulas, parameters and variables, that can be combine or reused, the process of tuning became complex and time consuming. In these cases, use of Parameter Optimization and Simulation Optimization became one of best problem solution.

An optimization problem normally consists on trying to find values, of free parameters of a system, in which objective function is maximized (or in some cases minimized.

Several problems resolutions can be made by searching the best configuration set of parameters, which fulfill the goal (or some goals). The goal is either to minimized or maximized some quantity. This quantity is express by a function *f*, of one or more variables known as the *objective function*.

Variables that can change in the quest for optimality are known as the *decision variables*. If goal is to minimize then *f* is known either as the cost function or the penalty function. In opposition, when the goal is maximization, *f* is referred as the *benefit function* or *utility function*.

These problems, classified into optimizations problems, are important in both theoretical and practical domain. In some case, the values of decision variables can be specified through a number of conditions - the constraints. For example, the range number in a variable can be considered as constraint, in which function *f* must take in consideration.

The *search space* of a problem is defined as the set of all candidate solutions. Each candidate solution is express by an instantiation of decision variables, and therefore by a quantity or objective function result. It is considered *feasible solution* or just a solution, if a candidate satisfies all constraints of the problem. The *search space* is formed by all feasible solutions.

Simulation Optimization procedures are used when our objective function can only be evaluated by using computer simulations. This happens because there is not an analytical expression for our objective function, ruling out the possibility of using differentiation methods or even exact calculation of local gradients. Normally these functions are also stochastic in nature, causing even more difficulties to the task of finding the optimum parameters, as even calculating local gradient estimates becomes complicated.

Running a simulation is always computationally more expensive than evaluating analytical functions thus the performance of optimization algorithms is crucial.

Theoretical problems with calibrating complex models is highlighted by Villa et al. [19] who developed and applied a computer aided search algorithm for exploring model parameter spaces, and compared these explorations against more usual methods of calibration such as eyeballing, hill climbing and Monte Carlo experiments. Villa et al. [19] found that as the number of unknown parameters increases, the number of areas that can be discriminated within the parameter space to fit the same observed data is also increasing. When less is known of a modeled system, systematic calibration of complex models will reveal more potential solutions. Consequently, non systematic calibrations, such as 'eyeballing', have been inconclusive as methods for exploring the total potential parameter space.

3 Optimization and Metaheuristics

There are several situations where one has to deal with problems of growing complexity. These problems arise in diverse areas of knowledge. Often, the problem to be solved can be expressed as an optimization problem where, for each particular instance, the goal is to find a solution which minimizes (or maximizes) a given objective function [5].

Optimization problems are commonly divided into two main categories [5]: those where solutions are encoded as real numbers; and those where solutions are encoded as discrete values. Amongst the later class, a prominent group is that of combinatorial optimization problems, where the objective is to find an optimal combination of solution components from a finite (or possibly countable infinite) set.

For some optimization problems either there is no knowledge how to definitively find a global optimum or the known algorithm has no practical usefulness due to its computational effort. They are known as being of difficult optimization [1][14][7]. For such cases, approximate algorithms play an important role [16]. Although they not (generally) guarantee that a global optimum would be found, they are (usually) able to find sub-optimal solutions within small time budgets.

For discrete problems several heuristics have been developed along the years in order to produce high quality solutions. The majority of them were conceived to solve a particular problem. Thus, some of the considerable effort putted in the development and refinement of such heuristics is more likely to be wasted if a (slightly) different problem has to be solved.

Metaheuristics are algorithmic frameworks that, at some extent, can be applied to a multiplicity of problems without major modifications [5]. They are in fact general high-level heuristics that guide an underlying search strategy in order to intelligently explore the solution space and return high quality solutions.

3.1 Hill-climbing

One of the principles behind Metaheuristic, is the definition of a neighbourhood leads to the definition of locally optimal solutions, or simply local optima. A local optimum is a feasible solution whose objective function is optimal in respect to a given neighbourhood, i.e. none of its neighbours have a better evaluation of the objective function [14].

The most obvious local search strategy is iterative improvement (known as hill-climbing in the case of maximization). Given an initial solution and a neighbourhood relation, the iterative improvement strategy moves to a neighbour if and only if it corresponds to an improvement of the objective function. The search process continues from the found better solution and iterates until no improvement is possible. The algorithmic skeleton of iterative improvement is depicted in figure 1.

```
begin
    $s$ ← GetInitialSolution()
    repeat
        $s$ ← PickImprovedSolution($\mathcal{N}(s)$)
    until $no\ improvement\ is\ possible$
end
```

Fig. 1. Hill-Climbing Algorithm.

3.2 Simulated Annealing

Simulated Annealing (SA), also known as *Monte Carlo* annealing and statistical cooling, is a stochastic local search metaheuristic. It was one of the first algorithms incorporating an explicit mechanism to escape from local optima [7].

Its motivation arises from the physical annealing of solids. Annealing is a thermal treatment applied to some materials (e.g. steel, brass, glass) in order to alter their microstructure, affecting their mechanical properties. The annealing process starts by initially raising the temperature of a substance to high values (melt point). At this state the particles of the substance are arranged randomly. Then, carefully proceeds with a slow cooling process spending long times at temperatures in the vicinity of the freezing point. This process allows the substance to solidify with a crystalline structure, a perfect structure that corresponds to a state of minimum energy – the ground state.

```
begin
    $s \longleftarrow$ GenerateIntiialSolution
    $T \longleftarrow T_0$
    while termination conditions not met do
        $s' \longleftarrow$ PickAtRandom($\mathcal{N}(s)$)
        if $f(s') < f(s)$ then
            $s \longleftarrow s'$
        else
            $s \longleftarrow$ AcceptanceCriterion($s, s', T$)
        UpdateTemperature($T$)
end
```

Fig. 2. Simulated Annealing Algorithm.

SA uses the physical annealing analogy to solve optimization problems [7]. In this analogy, the candidate solutions of the optimization problem have correspondence with the physical states of the matter, where the ground state corresponds to the global optimum (minimum). The objective function corresponds to the energy of the solid at a given state (see figure 2). The temperature initialized to a high value and then decreased during the search process, has correspondence, into some extent, to the iteration count.

The fundamental idea of SA is to make a walk based on a local search strategy but allowing accepting solutions worse than the current one. This provides the algorithm with a good mechanism to escape from local optima [7].

3.3 Genetic Algorithms

Genetic algorithms are adaptive methods, which may be used to solve search and optimization problems [23][1]. They are based on the genetic process of biological organisms. Over many generations, natural populations evolve according to the principles of natural selection, i.e. survival of the fittest, first clearly stated by Charles Darwin in The Origin of Species. By mimicking this process, genetic algorithms are able to evolve solutions to real world problems, if they have been suitably encoded [23].

Before a genetic algorithm can be run, a suitable encoding (or representation) for the problem must be devised. A fitness function is also required, which assigns a figure of merit to each encoded solution. During the run, parents must be selected for reproduction, and recombined to generate offspring (see Figure 3).

```
begin
    $P \longleftarrow$ GenerateInitialPopulation
    Evaluate($P$)
    while termination conditions not met do
        $P' \longleftarrow$ Recombine($P$)
        $P'' \longleftarrow$ Mutate($P'$)
        Evaluate($P''$)
        $P \longleftarrow$ Select($P'' \cup P$)
end
```

Fig. 3. Genetic Algorithm.

Termination conditions of the algorithm can vary. If there is a known optimal value to the fitness function an obvious choice is to stop when that value is reached (or at least within a given precision). However this criterion has usually to be extended due to several factors: the optimal value is unknown, there are no guarantees to reach the optimal value within a given time limit (or there are no guarantees at all). So, termination condition has to include some condition that indubitably stops the algorithm.

4 Simulation System Architecture

The simulation system framework, named **EcoSimNet**, was built to enable physical and biogeochemical simulation of aquatic ecosystems [8][9][10][11][12][3]. The core application, the simulator **EcoDynamo** [8], is an object-oriented program application, built in C++ and is responsible to communicate between model classes and the output devices where the simulation results are saved. The simulated processes include [9]:

- Hydrodynamics of aquatic ecosystems – current speeds, and directions;
- Thermodynamics – energy balances between water and atmosphere and water temperature;
- Biogeochemical – nutrient and biological species dynamics;
- Anthropogenic pressures, such as biomass harvesting.

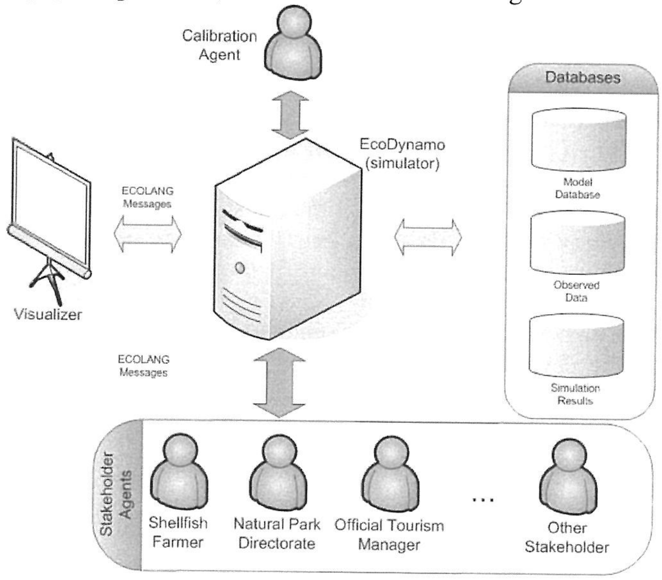


Fig. 4. Agent-based Simulation System Architecture.

Figure 4 defines the EcoSimNet architecture. The simulator has a graphical interface, where users can interact with ecological model properties: definitions as morphology, geometric representation of the model, dimensions, number of cell, classes, variables, parameter initial values and ranges. The user have the power to choose the model, which classes it simulates, the period of time simulated, and the period of time to output results for file or chart. The output files are compatible with major commercial software, for posterior treatments and the charts are generated by MatLab®.

Each model is influenced by its class, objects that represent real entities behavior, like wind, tidal current, dissolved substances, etc. Different classes simulate different variables and processes, with proper parameters and process equations. All data are kept in database files, for posterior comparison with observed data.

This framework has the capacity to extend functionalities by adding external/remote applications (typically the **Agents**)[21]. All applications communicate with simulator by a TCP/IP communication protocol using **ECOLANG** language [12] (the EcoSimNet protocol), defining all semantic messages necessary for management and simulation domain. Allow the interaction between ecological simulation experiments and several agents, representing either users of the system under simulation or applications designed to perform specific modeling tasks.

All Agents can do the same tasks as the users do with the simulator (start/stop the model simulation runs - start, stop, pause, restart and step – select classes, collect variables to output).

The **visualization** application interacts with simulator, representing graphically (2D or 3D) the morphology and stakeholders agents' interaction. The user can see information valid about classes simulated in some unit space (so called boxes).

5 Calibration Agent

Several procedures for automatic calibration and validation are available in the literature, like the Controlled Random Search (CRS) method [16] or the multi criteria model assessment methodology, Pareto Optimal Model Assessment Cycle (POMAC) [14]. However, these procedures do not capture the complexity of human reasoning in calibration process. They try to explore all search spaces, leaving to computer time consuming without best results.

The Calibration Agent (CA) is an Intelligent Agent [15][22] that communicates through the EcoSimNet protocol with the simulation application, assuming control over primary tasks around model understanding (ie. read/change model parameters, run simulation, collect results, etc…). Its purpose is to tune model equation parameters in order to fit the model to observed data, towards model calibration and validation.

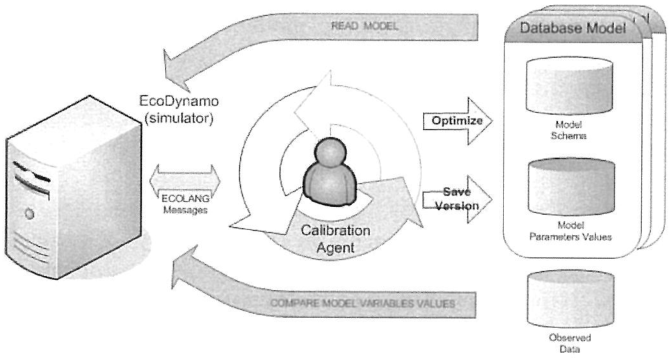


Fig. 5. Calibration Agent System Architecture.

All communications are made with simulator, using **ECOLANG** messages to compute input/outputs for the model loaded by application.

The CA acquires knowledge about the behavior of the system processes in five steps (see figure 5): 1) Simulator loads from Database Model, the schema and initial parameters values for model equation, 2) CA acquires the list of parameters and their values, 3) change the values of the parameters, using some knowledge based techniques, 4) run simulation model and 5) compare the lack of fitness between variables values simulated and observed ones.

The process finishes when the criteria of convergence is obtained or the user, in the graphical interface, stops the optimization. The user can save the parameter's vector of values, into the model database (Model Parameters Values Database), to load it in a new optimization, like the first step of the process.

This process is interactive, and its success depends, almost completely on the process of selecting the right parameter and their correct values. For this reason, the use of intelligent agents makes all the difference, because of its capacity of learning and change its strategies at any time of the computing cycle.

The knowledge about the behavior of all system processes, until these days restrained for the experts, or "modeler" in the traditional calibration process, shall be used to guide the selection of new values for the parameters contained in different mathematical relationships. It is important to understand the flow inside mathematical expressions for better calibration. From simulator, CA only knows the model classes (entities simulated), the variables (result from expressions) inside them and the values of the parameters. In the present system, the CA, learn knowledge model in three phases [9]:

- Capturing relationships among classes and inter-variable;
- Analyzing the intra and interclass sensitivity of different variables to different parameters and among variables;
- Iterative model execution (run simulation), measuring model lack of fit, adequacy and reliability, until a convergence criteria is obtained.

These phases make the methodology more robust with the minimal understanding of model flow, and can be transposed to others models. The complete calibration procedure is show in Figure 6.

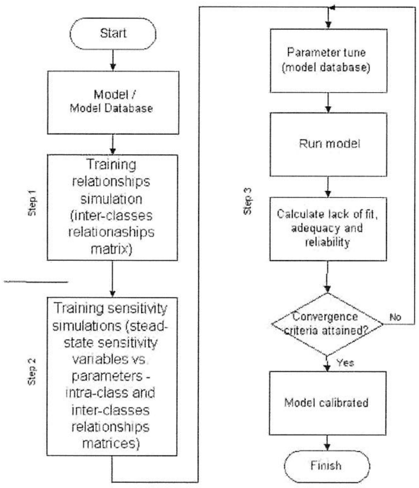

Start
Model / Model Database
Training relationships simulation (inter-classes relationaships matrix)
Step 1
Training sensitivity simulations (stead-state sensitivity variables vs. parameters - intra-class and inter-classes relationships matrices)
Step 2
Parameter tune (model database)
Run model
Calculate lack of fit, adequacy and reliability
Step 3
Convergence criteria attained?
No
Yes
Model calibrated
Finish

Fig. 6. Calibration Agent Procedure diagram [9]

The first step in CA is choosing the model to tune. This task became simplified, because the model loaded, for CA, is choosed by user interface in the simulator, and all extern modules read the same model.

Step1 and 2 from the diagram represent the understanding phase model. In these steps relationships matrix for classes and variables inside the same class and inter-class are constructed.

How can the CA know for each class variable, the interaction with others variables? This process is simplified because there are 2 internal simulator messages that reveal relationships: **Inquiry** and **Update** methods. Each class that interacts in ecosystem could know the values of variables in others entities – using the Inquiry method. If some class influences another uses the Update method for change some variable value in the other class, like in predator/prey model, where one class influences the other and vice-versa [6].

After this acquisition information or knowledge, by run model simulation for some period of time, the agent has in its power the minimal understanding how the entities model flow in the simulation.

After this phase the tuning process begins, applying the Metaheuristics algorithms related above. The optimization algorithm runs in search the adequate parameters values, not randomly but following the matrix of relationships constructed prior. The choice of optimization algorithms is influenced by the number of variables/parameters unknown that the model deals. Each optimization algorithm is tuned and combined with another Metaheuristics technique, to give the best values, in the valid time period.

6 Conclusions and Future Work

Model calibration is performed by comparing observed with predicted data and is a crucial phase in the modelling process. Because it is an iterative and interactive task in which, after each simulation, the expertise (or modeller) analyses the result and changes one or more equation parameter trying to tune the model, this "tunning" procedure requires a good understanding of the effect of different parameters over different variables. This is particularly painful in the simulation of ecological models, where the physical, chemical and biological processes are combined and the values of various parameters, which integrates the functions of the processes, are only estimated and may vary within a range of values commonly accepted by the researchers.

Using a calibration agent for model tuning enables full automate a very complex and tedious problem to solve manually, and without change the simulation code application. Because it is an agent, it can "live" abroad of core simulation and it is easier to upgrade the parameter tune techniques.

With this approach, we considered that some knowledge is gained into step 1 and 2 of the agent procedure diagram, but it is not trivial to compare the influence of parameters in variables between classes.

Simple controlled experiences have been made to test the validity of this approach for model calibration. However in terms of Metaheuristics, and model complexity, we can take result of other optimization techniques, like reinforcement learning.

The result of this work will be applied in the calibration of the Ria Formosa (Algarve) ecological model, in the context of ABSES project.

Acknowledgments

This work is partially supported by the ABSES project – "Agent-Based Simulation of ecological Systems", (FCT/POSC/EIA/57671/2004).

References

1. Beasley, D., Bull, D.R. and Martin, R.R. (1993). An Overview of Genetic Algorithms: Part 1, Fundamentals, University Computing, Vol. 15, No.2, pp. 58-69, Department of Computing Mathematics, University of Cardiff, UK.
2. Beven, K., Binley, A., 1992. The future of distributed models: model calibration and uncertainty prediction. Hydrological Processes 6, 279–298.
3. Cruz , F., Pereira, A., Valente, P., Duarte , P. and Reis, L. P.: Intelligent Farmer Agent for Multi-Agent Ecological Simulations Optimization. In: J. Neves, M. Santos and J. Machado (eds): EPIA 2007, LNAI 4874, pp.593-604, 2007. Springer-Verlag, Berlin Heidelberg. ISBN: 978-3-540-77000-8 (doi:10.1007/978-3-540-77002-2_50).
4. Duarte, P., Meneses, R., Hawkins, A.J.S., Zhu, M., Fang, J., Grant, J.: Mathematical modelling to assess the carrying capacity for multi-species culture within coastal waters. Ecological Modelling 168, 109–143 (2003).
5. Glover, Fred; Kochenberger, Gary - Handbook of Metaheuristics (International Series in Operations Research & Management Science). Springer, 2003.

6. Jørgensen, S. E. and Bendoricchio, G.: Fundamentals of Ecological Modelling. Elsevier Science Ltd, 3rd edition. 2001.
7. Kirkpatrick, S.; Gelatt, C.; Vecchi, M. Optimization by Simulated Annealing, Science, 220 (4598), pp. 671-680, 1983.
8. Pereira, A. and Duarte, P.: EcoDynamo – Ecological Dynamics Model Apllication (Technical Report), University Fernando Pessoa, 2005.
9. Pereira, A., Duarte, P., Reis, L.P.: Agent-based Ecological Model Calibration – On the Edge of a New Approach. In: Ramos, C. and Vale, Z. (eds.), Proceedings of the International Conference on Knowledge Engineering and Decision Support, pp. 107-113, ISEP, Porto, Portugal, July. ISBN: 972-8688-24-5.
10. Pereira, A., Duarte, P., Reis, L.P.: Agent-Based Simulation of Ecological Models. In: Coelho, H., Espinasse, B. (eds.) Proceedings of the 5th Workshop on Agent-Based Simulation, Lisbon, pp. 135–140 (2004)
11. Pereira, A., Duarte, P., Reis, L.P.: An Integrated Ecological Modelling and Decision Support Methodology. In: Zelinka, I., Oplatková, Z., Orsoni, A. (eds.) 21st European Conference on Modelling and Simulation, ECMS, Prague, pp. 497–502 (2007)
12. Pereira, A., Duarte, P., Reis, L.P.: ECOLANG – A Communication Language for Simulations of Complex Ecological Systems. In: Merkuryev, Y., Zobel, R., Kerckhoffs, E. (eds.) Proceedings of the 19th European Conference on Modelling and Simulation, Riga, pp. 493–500 (2005)
13. Reynolds, J. H., Ford, E. D. - Multi-Criteria Assessment of Ecological Process Models. Washington: Currently Department of Statistics, University of Washington,, 1998. NRCSE-TRS No. 010.
14. Roberts, M., Howe, A. and Whitley, L.D.: Modeling Local Search: A First Step Toward Understanding Hill-climbing Search in Oversubscribed Scheduling. IN: American Association for Artificial Intelligence (www.aaai.org), 2005.
15. Russel, S., Norvig, P.: Artificial Intelligence: A modern approach, 2nd edn. Prentice-Hall, Englewood Cliffs (2003).
16. Scholten, H. and van der Tol,: Quantitative Validation of Deterministic Models: When is a Model Acceptable?. Proceedings of Summer Computer Simulation Conference, 404-409, SCS int., San Diego, CA, USA (July 12-22, 1998, Reno, Nevada, USA)
17. The DITTY project description [online]. Available at http://www.dittyproject.org [visited January, 8, 2008]
18. Thomas Back and Hans-Paul Schwefel. *Evolutionary computation:An overview.* In T. Fukuda, T. Furuhashi and D. B. Fogel, editeur, Proceedings of 1996 IEEE International Conference on Evolutionary Computation (ICEC '96), Nagoya, pages 20{29, Piscataway NJ, 1996. IEEE Press.
19. Villa, F., Boumans, R.M.J., Costanza, R., 1998. Calibration and testing of complex process-based simulation models. Proceedings of the Applied Modeling and Simulation (AMS) Conference, Honolulu, 12–14 August 1998.
20. Wang, Q.L., 1997. Using genetic algorithms to optimize model parameters. Environmental Modeling and Software 12 (1), 27–34.
21. Weiss, G.: Multiagent Systems. MIT Press, Cambridge (2000).
22. Wooldridge, M.: An Introduction to Multi-Agent Systems. John Wiley & Sons, Ltd., Chichester (2002).
23. Holland, J. "Genetic algorithms," Scientific American, Jul., pp. 44-50, 1992.

A System of Automatic Construction of Exam Timetable Using Genetic Algorithms

José Joaquim Moreira

proDEI, Faculdade de Engenharia da Universidade do Porto
Rua Dr. Roberto Frias
4200-465 Porto, Portugal
jjmoreira@gmail.com

Abstract. The complexity of the problem of exam timetables is justified by the scheduling size of the examinations and the high number of constraints and criteria for allocation. This paper presents a method of solution to the problem of automatic construction timetables for the exams. Among several mathematical models of representation, the final option was for a model matrix, which is justified by the benefits that this model presents when used in the algorithm of solution. The method of solution is a meta-heuristics that includes a genetic algorithm. The model is directed to the construction of exam timetables in institutions of higher education. The results achieved in real and complex scenarios are satisfactory; the exam timetabling meets the regulations imposed. We conclude that when the algorithm does not determine a solution with no penalty, is because that solution does not exist.

Keywords: scheduling, timetabling problems, exam timetabling, genetic algorithms.

1 Introduction

Every school year, each season of exams, the various departments of an institution of education facing the difficult task of drawing up timetables for examinations.

The difficulty due to be great complexity of the construction of timetables for exams, due the scheduling size of the examinations and the high number of constraints and criteria of allocation, usually circumvented with the use of heuristics little strict, based on solutions from previous years.

The objective of this work is the schedules of examinations. The main purpose is to demonstrate the possibility of building them, automatically, using computers.

The term scheduling applies to a kind of problems that, according Wren [1] distribute objects, subject to certain constraints, in a pattern of time or space, so that the costs of these are minimum. Objects may be people, vehicles, machines, exams, etc.., constraints are the rules that govern the process of scheduling, and some are inviolable while others take the form of principles that must be obeyed.

The problem of production of a factory described by Thompson [2], the problem of traveling salesman approached by Wren [1] and the problem of school timetabling, with a solution proposed by Queirós [3], for example, can be seen in perspective problems of sequential scheduling.

This subject has received special attention of the scientific community in the last five decades. This great interest, causes in 1995, the creation of series of conferences PATAT (Practice and Theory of Automated Timetabling) with editions every two years [4] and the establishment of EURO (Association of European Operational Research Societies) WATT (Working Group on Automated Timetabling). In 2002 emerged with the support of PATAT, the International Competition of Timetabling [5].

In this work, the genetic algorithm is the method of solution. Designed by John Holland [6] at the end of the fifties years, uses a structure similar to that set out by Charles Darwin in "The Origin of Species'. It is based on two main operators: selection and reproduction, activated in the presence of a number of solutions, called population.

The formal model express using matrix representations. The application of genetic algorithm to matrix model, create exam timetables that meet the regulations imposed.

1.1 Related Works

Several works have approached the timetabling problem. Oliveira [7] presents a language for representation of the timetabling problem, the UniLang. UniLang intends to be a standard suitable as input language for any timetabling system. It enables a clear and natural representation of data, constraints, quality measures and solutions for different timetabling (as well as related) problems, such as school timetabling, university timetabling and examination scheduling.

Gröbner [8] presents an approach to generalize all the timetabling problems, describing the basic structure of this problem. Gröbner proposes a generic language that can be used to describe timetabling problems and its constraints.

Chan [9] discusses the implementation of two genetic algorithms used to solve class-teacher timetabling problem for small schools.

Fang [10], in his doctoral thesis, investigates the use of genetic algorithms to solve a group of timetabling problems. Presents a framework for the utilization of genetic algorithms in solving of timetabling problems in the context of learning institutions. This framework has the following important points, which give you considerable flexibility: a declaration of the specific constraints of the problem and use of a function for evaluation of the solutions, advising the use of a genetic algorithm, since it is independent of the problem, for its resolution.

Fernandes [11] classified the constraints of class-teacher timetabling problem in constraints strong and weak. Violations to strong constraints (such as schedule a teacher in two classes at the same time) result in a invalid timetable. Violations to weak constraints result in valid timetable, but affect the quality of the solution (for example, the preference of teachers for certain hours). The proposed algorithm, evolutionary, has been tested in a university comprising 109 teachers, 37 rooms, 1131 a time interval of one hour each and 472 classes. The algorithm proposed in resolving the scheduling without violating the strong constraints in 30% of executions.

Eley [12] in PATAT'06 presents a solution to the exam timetable problem, formulating it as a problem of combinatorial optimization, using algorithms Ant, to solve.

Analised the results obtained by the various works published, we can say that the automatic generation of schedules is capable of achieving. Some works show that when compared with the schedules manuals in institutions of learning real, the times obtained by the algorithms for solving the class-teacher timetabling problem are of better quality, since, uses some function of evaluation.

1.2 Organization of Paper

The concepts introduced in the Introduction are consolidated in the two chapters that follow. Thus, in Chapter 2, present the objectives of the exam timetables problem. Chapter 3 is devoted to the presentation of the method of solution used in this paper, describing the main concepts of Genetic Algorithms. Chapter 4, describes the activity of building exam timetables, it presents the model proposed, through its formalization, the model subjected to a simulated test with a simple problem, only for the purpose of demonstration and validation of the model and discuss the results, including results of larger problems and real. In Chapter 5, are pointed out the main conclusions and directions of future work.

2 Exam Timetables

The resolution of the exam timetables problem can be claimed by different areas, such as the School Administration, Artificial Intelligence, Mathematics or Operational Research. Probably, we must appeal the techniques of simulation imported from fields as diverse as physics or biology, to solve the problem.

The purpose of the exam timetable is scheduler exams, according to pre-defined periods of time; minimizing losses teaching for the students, such as realize examinations on the same day or on consecutive days. But here, it considers each student individually, since the choice may depend only of the route of each school students.

The importance of the constraints, the quantity and quality of which are, stems directly from the attempt to organize the problem. In this sense, we go classify, previously the constraints. Classified as constraints of the first order, or rigid, those are not being met, and it makes the scheduling illegal, calling themselves 'impossible solutions'. Other constraints, which should obey, and which, if not met, do not make illegal the scheduling, considered being of second order constraints, or flexible. So, we called the 'impossible solutions' the scheduling, that check the constraints of the first order, Regardless of check, or not, the constraints of second order.

This division represents two moments in the resolution of the exam timetables problem. The first, consisting in the search for possible solutions, in the development of heuristics to ensure that the scheduling chosen corresponds to a possible solution. The second, consisting in finding the best solution. The first runs in the space of all scheduling - which includes possible and impossible solutions; the second follows, just in the space of possible solutions.

3 Genetic Algorithms

The genetic algorithms distinguish themselves in the field of methods of optimization and search for the assimilation of the Darwinian paradigm of the evolution of species.

The genetic algorithms are processes of convergence [3]. Its structure is governed by import laws of the theory of evolution of species and concreteness in two fundamental concepts: selection and reproduction. The confrontation between genetic algorithms and the real problems is promoted by the need for optimization. It follows an space of enormous dimensions, in which each point represents a potential solution to the problem. In this maze of solutions, only a few, if not only one, fully satisfy the list of constraints that give shape to the problem.

The problems of optimization, usually associated with the satisfaction of constraints, define a universe of solutions, leaving the genetic algorithm to determine the overall solution, or a solution acceptable as a limitation on the time of action of the algorithm.

The genetic algorithms are search algorithms based on mechanisms of natural selection and genetics. Usually used to solve optimization problems, where the space of search is great and conventional methods is inefficient.

3.1 Characteristics

The terminology they are associated translate the import of essential concepts of genetics and guesses the importance attributed to the interaction of these concepts. The concept of population, like number of individuals of the same species, is extended to artificial species. Individuals are normally represented by sequences of numbers: the genotype. The numbers, or rather, a collection of numbers, is the genetic heritage of the individual, determining their characteristics, that is, its phenotype. The genetic algorithms differ from traditional methods of research and optimisation, mainly in four aspects:

1. Work with a codification of the set of parameters and not with their own parameters;
2. Work with a population and not with a single point;
3. Uses information from or gain cost and not derived or other auxiliary knowledge;
4. Uses rules of transition probability and not deterministic.

The solutions interact, mix up and produce offspring (children) hoping that retain the characteristics "good" of their ascending (parents), which may be seen as a local search, but widespread. Not only the neighbourhood of a simple solution that is exploited, but the neighbourhood of a whole population.

The members of the population are called individuals or chromosomes. As in natural evolution, the chromosomes are the base material (virtual, in this case) of heredity. Uses currently a function of evaluation that associates each individual, a real number that translates to adaptation.

Then, in a manner directly proportional to the value of their adaptation, are selected pairs of chromosomes that will cross themselves. Here, can be considered the selection with elitism, or ensure that the best solution is part of the new generation.

His crossing is the result of artificial selection, considering more adapted those that best meet the specific conditions of the problem. The crossing of the numerical sequences promotes the emergence of new sequences, formed from the first. With a probability established, after crossing, a mutation can happen, where a gene of chromosome changes.

These new individuals are the second generation of individuals and mark the end of cycle of the genetic algorithm. The number of cycles to perform depends on the context of the problem and the level of quality (partial or full satisfaction of the restrictions), which is intended for the solution.

A simple genetic algorithm describes the following cycle:

1st Generation of random n chromosomes that form the initial population;
2nd Assessment of each individual of the population;
3rd Verification of the termination criteria;
4th If verify termination criterion - cycle ending;
5th Selection of n/2 pairs of chromosomes for crossover;
6th Reproduction of chromosomes with recombination and mutation;
7th New population of chromosomes called new generation;
8th Back to step 2.

The cycle described above is illustrated in Figure 1.

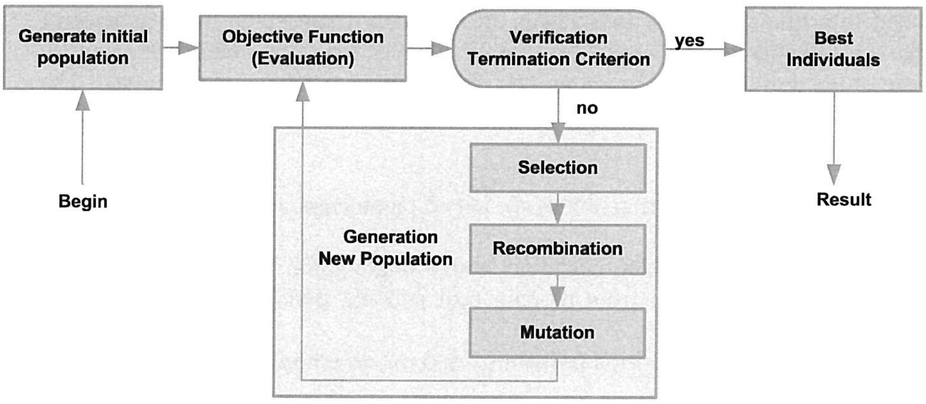

Generate initial population
Objective Function (Evaluation)
Verification Termination Criterion
yes
Best Individuals
no
Selection
Begin
Result
Generation New Population
Recombination
Mutation

Fig. 1. Basic structure of the genetic algorithm

4 Construction of Timetables for Exams

4.1 Model Proposed

The model we propose, matrix class, translates well the problem treated in this paper. Represents the allocation (or scheduling) of the examinations to the periods of time, supporting the limits that impose constraints conventional. The timetables are in the form of matrices of whole numbers, therefore easily manipulated by genetic algorithms.

4.1.1 Definitions

The construction of timetables for examinations requires the prior definition of some initial conditions. These conditions can be grouped into two broad areas: conditions of representation and conditions of constraints. The first is internal and coordinate the division of the periods of time and organization of resources; the second is external and limit the universe of scheduling.

As we presented the model proposed, it is subjected to a simulated test with a simple problem, only for the purpose of demonstration and validation of the model, the model is sufficiently broad to be confronted with real problems and more complex.

Conditions of representation

The scheduling of examinations assumes the prior organization of the days / periods of time that will be allocated exams. Admittedly, for example, that particular institution, for a certain period, defined two shifts - morning and afternoon - exams for day. Being assigned, respectively, to the turn of the morning and part of the afternoon, one (10h) and two periods of time (14h and 17h) for the conduct of examinations. In Table 1 we have the distribution of periods of time each day.

Table 1. Distribution periods of time

Days	Turn	Period of time
1st	afternoon	14h e 17h
2nd	morning + afternoon	10h, 14h e 17h
3rd	morning + afternoon	10h, 14h e 17h
4th	morning	10h
5th	morning + afternoon	10h e 14h

This definition by the institution, result 2 x 3 + 2 x 2 + 1 x 1 (11) periods of time, where 1st period corresponds to the first day available for examination at 14h and 11th period corresponds to the last day available, at 14h.

For each of the subjects (13) that will be subjected to exams we have the subscription from each student, indicating the code of subjects and of the number of student enrolled.

Conditions of constraints

We will divide the conditions of constraints into two classes: first-order constraints and second order constraints.

Constraints of the first order:

1. A student may not have more than one examination on the same day;
2. Maximum number of examinations (classrooms available);
3. Preference of teachers (pre-marked examinations).

Constraints of second order:

1. A student should not have exams on consecutive days;
2. Examinations of a student evenly spread.

4.1.2 Representation Model Exam Timetables

Definition

H - set of all the periods of time that can occur examinations.

$H = \{h_1, h_2, ...h_m\}$, Where m corresponds to the maximum number of periods of time. In the previous example would: $H = \{1,2,3,...9,10,11\}$

Definition

D - set of all subjects, in a given season, will be under examination.

$D = \{d_1, d_2, ...d_k\}$, Where k is the maximum number of subjects, in a given season will be under examination. In the previous example would: D = {1101, 1102, 1103, 1104, 1105, 1106, 2201, 2202, 2203, 2204, 2205, 2206, 2207}

Model

A matrix M with 1 line and k columns that represent, in order, the subjects (examinations), of the D set, which will be scheduling. Each column contains a value withdrawn from of the H set, indicating the time at which the exam was allocated.

4.2 Application of the Model

Each subject, is given a serial number, according to the subscription of students in examinations. Thus, each matrix with 1 line and 13 columns (number of subjects) of elements, that for each column is permutations of H set, is a solution to the problem of timetables for examinations.

Although the problem presented be extremely simple, the space of candidate solutions to global solution comprises 34 522 712 143 931 different points (arrangements with repetition of eleven elements taken thirteen to thirteen). The growth of the variables in real problems increases the number of potential solutions for values even more enormous, being almost impossible a systematic evaluation to all solutions. Now, we present two solutions of exam timetable, obtained at random – table 2.

Table 2. Tow solution

$$\begin{array}{l} \text{n nr} = [\ 3\quad 10\quad 9\quad 1\quad 9\quad 3\quad 9\quad 10\quad 2\quad 9\quad 1\quad 9\quad 7\ ] \\ \text{n n} = \underbrace{[\ 7\quad 2\quad 1\quad 4\quad 6\quad 9\quad 6\quad 10\quad 7\quad 9\quad 9\quad 7\quad 10\ ]}_{[\ \text{nnn nnn nnn nnn nnn nnn nnn nnn nnn nnnn nnnn nnnn nnnn}\ ]} \end{array}$$

In which each $\mathrm{n_n}$ is replaced by subject (examination), whose serial number corresponds to the index of n. To assign the examinations to the respective periods of time we used the following examinations mask for the allocation of the periods of time (H), table 3.

Table 3. Mask of examinations for periods of time (H)

	1st day	2nd day	3rd day	4th day	5th day
10h		H3(3)	H6(6)	H9(9)	H10(10)
14h	H1(1)	H4(4)	H7(7)		H11(11)
17h	H2(2)	H5(5)	H8(8)		

Thus, the first solution presented (n n), result in the following schedule of exams - Table 4. Example: subject nn(1001) allocated in position 3 (H3), etc.

Table 4. Calendar of examination for the solution n n

	1st day	2 nd day	3 rd day	4 th day	5 th day
10h		1101 1106		1103 1105 2201 2204 2206	1102 2203
14h	1104 2205		2207		
17h	2203				

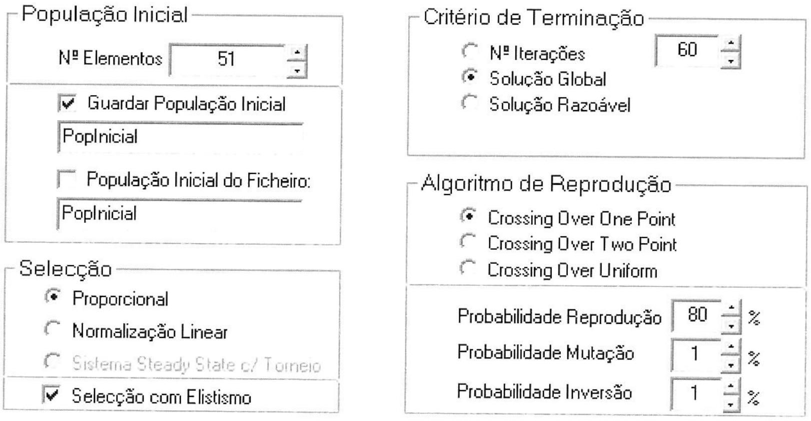
The next step of the algorithm is to evaluate each of the solutions (calendar) through a function of evaluation (1).

$$f(c)= P_1R_1 + P_2R_2 + P_3R_3 + P_4R_4 + P_5R_5 \tag{1}$$

Where P1, P2, P3, P4 and P5 is the value of the penalty for each constraint. R1, R2, R3, R4 and R5 represent the number of times the calendar *c* violates the restrictions 1,2,3,4 and 5, respectively.

Each solution, now is associated a numerical value that reflects their adaptation to the environment, or the conditions of constraints. The next stage that follows by the genetic algorithm, consist in the selection of n individuals, possibly with repetition, can also suffer mutation.

Before starting the computer generation of the schedule of examinations, it is necessary to customize the genetic algorithm. In the next figure we have the elements of customization of computational application – prototype.



População Inicial

Nº Elementos 51

☑ Guardar População Inicial

PopInicial

☐ População Inicial do Ficheiro:

PopInicial

Selecção

● Proporcional

○ Normalização Linear

○ Sistema Steady State c/ Torneio

☑ Selecção com Elistismo

Critério de Terminação

○ Nº Iterações 60

● Solução Global

○ Solução Razoável

Algoritmo de Reprodução

● Crossing Over One Point

○ Crossing Over Two Point

○ Crossing Over Uniform

Probabilidade Reprodução 80 %

Probabilidade Mutação 1 %

Probabilidade Inversão 1 %

Fig. 2. Customization of the genetic algorithm – Prototype

4.3 Evaluation

Test scenario

An institution with 4 courses with a total of 77 subjects. 250 students are enrolled in examinations. Many students are enrolled in tests of previous years (delayed subjects).

Were created two instances of the problem for testing. Then each instance was submitted to the prototype.

- Instance 1 – 45 days with 3 periods of time
- Instance 2 – 32 days with 3 periods of time

The computer of test had a 1.0 GHz Pentium III processor with 384 MB of RAM memory. The customization basis of the genetic algorithm was that is represented in figura 2. In each test, only changed the number of elements (solutions) of the initial population.

4.3.1 Results

In the various executions of the algorithm, we observed that the evolution of penalties from iteration to iteration (cycle of the algorithm) had a downward behavior - figure 3.

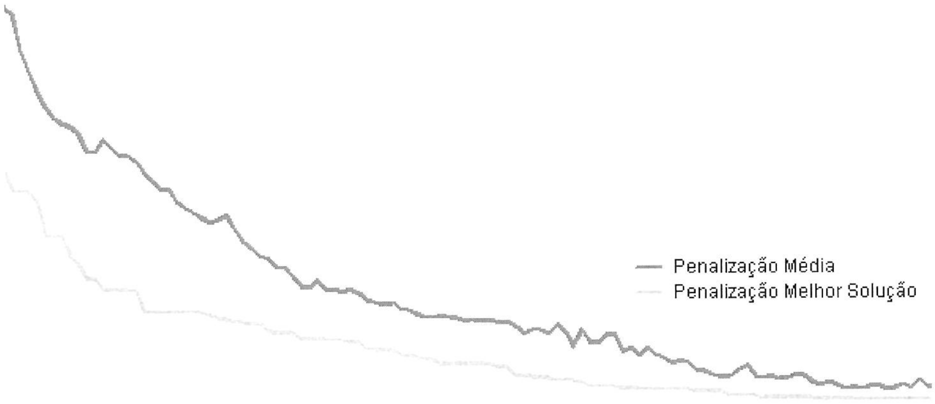


Fig. 3. Penalties evolution

In tables 5 and 6, have the results for the instances 1 and 2, respectively.

Table 5. Results of first instance

	Elements of the initial population	Penalty of 1st iteration (cycle)	Iteration (cycle) of the solution with penalty zero	Time
Test 1	51	12080	137	50 min
Test 2	101	8020	118	120 min

Table 6. Results of second instance

	Elements of the initial population	Penalty of 1st iteration (cycle)	Iteration (cycle) of the solution with penalty zero	Time
Test 1	51	14180	233	240 min
Test 2	101	10760	180	180 min

Increased amounts of elements (solutions) of the initial population, the one hand, it is more demanding for the computer, but, on the other hand, the overall solution is found with fewer iterations of the algorithm.

5. Conclusions and Future Work

The problem studied in this work, together with school timetabling, belongs to the class of more complex problems of combinatorial optimization with satisfaction of constraints. Its importance is well measured at each time of examinations, in each school year.

Our proposal focused on the preparation automatic exam timetables using computers. We define the problem and define the method of solution through genetic algorithms.

The foundations for the construction of an automatic generator has completed to the definition of an automatic model that represents the problem and the structure of the genetic algorithm. The model chosen, of nature matrix presents, in addition, other advantages: it is based on rigorous mathematical definitions, adjusting to an efficient analysis of the quality of the calendars, each matrix represents a solution (possible or impossible) and its elements belong to the set of integer numbers.

Under these conditions it has developed a prototype of automatic generation of schedules of examinations. All functions of the genetic algorithm were coded in the Visual Basic language.

The results achieved in all tests performed with real scenarios, in general, are satisfactory. The schedules of examinations meet the regulations imposed. When the algorithm does not determine a solution with zero penalty, can be explained by two reasons: this solution does not exist, that is, the overall solution admits some penalty, and / or the occupation, with examinations of all periods of time, that is, the inability to move an examination without changing the allocation of another examination.

In addition to the natural advantages in automating the process of the construction of timetables for examinations, it should be noted, also, the facilities at the editing of data that includes automation.

This work raises some clues for subsequent searches that can be topped. Thus, in the short term, the contemplation of the scheduling of examinations take into consideration the type of classrooms and also specify the number of vigilant required for each exam. In the long term, reconsideration of adaptive techniques, confronting the results of three types of algorithms: genetic algorithms, tabu search algorithms and simulated annealing algorithms.

References

1. Wren, Anthony: Scheduling, Timetabling and Rostering — a special relationship? Proceedings of the 1st International Conference on the Practice and Theory of Automated Timetabling, 474-495 (1995).

2. Thompson et al, Jonathan e Dowland, Kathryn: General Cooling Schedules for a Simulated Annealing Based Timetabling System. Proceedings of the 1st International Conference on the Practice and Theory of Automated Timetabling, (1995)
3. Queirós, F. H.: Construção automática de Horários de Aulas. Tese de Mestrado, Universidade Portucalense (1995).
4. PATAT, Conferences: The International Series of Conferences on the Practice and Theory of Automated Timetabling (PATAT) - http://www.asap.cs.nott.ac.uk/patat/patat-index.shtml
5. International timetabling competition: http://www.cs.qub.ac.uk/itc2007/index_files/overview.htm
6. Holland, John: Scheduling, Adaptation in Natural and Artificial Systems. The University of Michigan Press (1975).
7. Oliveira, E., Reis L.P.: A Language for Specifying Complete Timetabling Problems. 3th International Conference on the Practice and Theory of Automated Timetabling PATAT'2000 (2000).
8. Gröbner, M., Wilke P.: A General View on Timetabling Problems. 4th International Conference on the Practice and Theory of Automated Timetabling PATAT'2002 (2002)
9. Chan, H. W.: School Timetabling Using Genetic Search. 2th International Conference on the Practice and Theory of Automated Timetabling, PATAT'97 (1997)
10. Fang, H. L.: Genetic Algorithms in Tametabling Problems. PhD Thesis, University of Edinburgh (1994)
11. Fernandes, C.: Infected Genes Evolutionary Algorithm for School Timetabling. WSES International Conference (2002)
12. Eley, M.: Ant Algorithms for the Exam Timetabling Problem. 6th International Conference on the Practice and Theory of Automated Timetabling, PATAT'06 (2006)

Collaborative Ontology Specification

Carla Sofia Pereira

INESC Porto, Campus da FEUP, Rua Dr. Roberto Frias, 378,4200-465, Porto, Portugal
ESTGF - IPP, Casa do Curral, Rua do Curral, Apartado 205, 4610-156, Felgueiras, Portugal
csp@inescporto.pt

Abstract. The use of methodologies in the development of ontologies is a common practice. Until now several methodological proposals have been presented for building ontologies. Ontologies are forms of a priori social agreements on concepts. Therefore, reaching those agreements is a fundamental step to their success. Traditionally, the ontology engineering field has laid a lot of emphasis on the "specification of the conceptualization" as an engineering task, but the work developed about the social construction of the conceptualization itself has been scarce. In this paper, we present the state of the art in the collaborative ontology specification and a comparative analysis of the several existent approaches based on some criteria defined. The main conclusions are that up to now there are few detailed proposals for the collaborative construction of ontologies in (distributed) groups of human actors and there is no completely mature approach to support the collaborative specification.

Keywords: Ontology Engineering, collaboration, ontology specification, comparative analysis.

1 Introduction

Due to the industrial and economic environment, collaborative networks will tend to be formed and to exist for short periods of time, i.e., the time needed to complete a business opportunity. How to structure the information for purposes of supporting the activities of temporary collaborative networks will therefore be a major difficulty in the establishment of the semantic agreements that will be the cornerstone for sharing information and knowledge. In the last decade, research in this field has shown ontology engineering as the most promising technology to attain semantic interoperability of systems [5]. However, ontologies are forms of priori social agreements made about a conceptualization of a given part of the world. Therefore, reaching those agreements is a fundamental step to their success. Traditionally, the ontology engineering field has laid a lot of emphasis on the "specification of the conceptualization" but work developed about the social construction of the conceptualization itself has been scarce. This is even more noticeable in the application of ontology engineering to collaborative network contexts.

Ontology creation needs a social presence as it requires an actor to predict reliably how other members of the community will interpret the concepts of an ontology just based on their limited description. By incorporating the notion of semantics into the web architecture, we thus transform the users of the system themselves into a critical part of the design.

As it is known the word ontology was taken from philosophy, where it means a systematic explanation of being. In the last decade, as it was referred above, the word ontology became relevant for the knowledge engineering community. Today, many texts about what an ontology is can be found in the literature of several scientific areas, and it is possible even to trace how those definitions evolved over time.

Despite of this, one of the first definitions still reflects accurately the essence of ontologies as applied to the information systems area: "an explicit specification of a conceptualization" [11]. This definition gave origin to many other, and it is the reference definition in this paper.

One of the more comprehensive studies on what an ontology is [3] concluded that "ontologies aim to capture consensual knowledge in a generic and formal way, and that they may be reused and shared across applications (software) and by groups of people. Ontologies are usually built cooperatively by a group of people in different locations". This conclusion considers the importance of including the collaboration principles in the ontology development process, more precisely, in the specification phase. Other aspects already mentioned and considered as fundamental for our research work are the need of a social construction of the conceptualization and the application of ontology engineering to collaborative network contexts, reinforce the need of collaboration in this process.

Making an analogy with the information systems development process, the specification and conceptualization phase of an ontology is similar to information systems analysis which include the following activities: requirements elicitation, analysis and negotiation, and documentation. In this work, conceptualization and specification of an ontology are considered as one single phase, named specification.

For us, the specification phase includes the identification of the concepts to include in the ontology, their characteristics, definition and relationships, as well as the knowledge organization and structuring using external representations independent of the implementation language and environment.

The focus of this paper is not in the methodologies for building ontologies, but the study, in a detailed way, of the specification phase of each collaborative methodology existent. The other phases are not part of the goals of this work. Hence, this paper presents the state of the art in the collaborative ontology specification and a comparative analysis of the several existent approaches. The rest of the paper is structured as follows: Section 2, shortly reviews the most relevant methodologies for building ontologies and refers the importance of the specification phase in the development process. Section 3, presents a brief description of the work developed in this area up to now and finishes with a definition of collaborative ontology specification and some principles for collaborative specification. Section 4, presents a comparative analysis. Section 5, presents a brief discussion about the approaches for collaborative ontology specification. Section 6, provides some conclusions of this work and proposes future directions.

2 Ontology specification vs ontology development methods

An ontology can be developed collaboratively by distributed individuals and organizations with different expertise, goals, and interactions. Various communities of experts and practitioners examine problems from different angles and are concerned with different dimensions of the semantic contents and representation. These individuals all need to properly understand each other and meaningfully communicate their views of domain knowledge to form meaningful higher-level knowledge: the ontology [5].

An ontology development methodology comprises a set of established principles, processes, practices, methods, and activities used to design, build, evaluate and deploy ontologies. Several such methodologies have been reported in the literature. From the analysis of some surveys [8] concluded that: 1/ most ontology development methodologies that have been proposed focus on building ontologies; 2/ some other methodologies also include methods for merging, reengineering, maintaining, and evolving ontologies; and 3/ yet other methodologies build on general software development processes and practices and apply them to ontology development. The authors present two important observations that result of their brief survey of ontology development methodologies: 1/There are many common points in the various methodologies. Steps in different processes may be named differently, may also be of different granularity, or may only partially overlap, but the processes are still very much alike; 2/ Many of the principles and practices of ontology development are analogous to those of software engineering [8]. [7] present the following conclusions: it doesn't exist a completely mature methodological proposal for building ontologies, since there are some important activities and techniques that are missing in all of these methodologies; not all of the methodologies have the same degree of maturity; presents Methontology as a very mature methodology; although the work to unify proposals can be interesting, maybe several approaches should coexist and refer the lack of detailed description of the techniques used to build ontologies in all methodologies. They refer also the lack of approaches for collaborative development.

Just as in the information systems development the analysis and specification phase has great influence, or maybe it is the one that has more influence, in the success of the system. When we speak about ontology development, the question is the same, the specification phase is, in our opinion, the main responsible for the quality and success of the created ontology. Therefore, research questions proposed here relative to current ontology development methods are: 1/ how structured and how detailed is the specification process defined? 2/ which methods, techniques and tools are proposed? 3/ which actors and associated competencies are considered? 4/ how is collaboration considered within the specification process, including the characteristics of the used artefacts and the actors involved? The several definitions of ontology [3] sent for a process of collaborative conceptualization of the domain. This is fundamental if we want to apply ontology engineering to collaborative network.

3 Collaborative ontology specification: the state of the art

[12] reinforce the need of methodologies to support the phase of knowledge acquisition. The need of tools that support the knowledge conceptualization and that starting from this generate the code of the ontology. These authors consider that the ontology developers frequently pass directly from the knowledge acquisition to the implementation phase of the ontology. When most of the knowledge has been acquired, the ontologist has a lot of unstructured knowledge that must be organized. They present Methontology as a methodology that was created for building ontologies either from scratch, reusing other ontologies as they are, or by a process of reengineering them. The Methontology framework enables the construction of ontologies at the knowledge level. It includes: the identification of the ontology development process, a life cycle based on evolving prototypes and the methodology itself, which specifies the steps for performing each activity, the techniques used, the products to be output, and how the ontologies are to be evaluated. Related with the ontology specification it deals with the following aspects: the specification states why the ontology is being built, which are its intended uses and who are the end-users; a conceptualization that structures the domain knowledge as meaningful models at the knowledge level; the reutilization of other ontologies that are already available [3], [7] and [12].

[14] and [15], present the On-To-Knowledge methodology that is the result of the project with the same name. This methodology includes the phases of feasibility study, kickoff phase, refinement phase, evaluation phase and maintenance phase. In the kickoff and refinement phases the activities involved in the ontology specification are performed. The kickoff phase is then where ontology requirements are captured and specified, competency questions are identified, potentially reusable ontologies are studied and a first draft version of the ontology is built. The output product is an ontology requirements specification document. The goal of the refinement phase is to produce a mature and application-oriented target ontology according to the specification given by the kickoff phase.

The approaches selected in this review are those that consider, in some way, the collaborative ontology specification. The Methontology methodology was just selected because it is considered in the literature as the most complete and mature, although, in our opinion, it doesn't consider the collaborative ontology construction. The methodology On-To-Knowledge was selected because there is a report of at least one case study in which it was used to support a collaborative ontology construction, as described bellow.

[4] describe OntoShare, an ontology-based system for sharing information among users in a virtual community of practice and describe the deployment and evaluation of OntoShare in a particular community as part of a case study within the project On-To-Knowledge, OntoShare has been applied and evaluated using the On-To-Knowledge methodology. In this study it is interesting to analyze how the kickoff and refinement phase was executed, given that it is in this phase that the ontology is specified. The kickoff and refinement stages of the methodology were carried out at a workshop with key people from the user group. This was held at the company's premises and run by a knowledge engineer. It was very much brainstorming oriented during the kickoff phase [4]. The group was able to produce the ontology at the

workshop which meant that most of the refinement stage had been carried out in tandem with the kickoff stage.

[9] based on the work of [10] reviewed some of the most representative methodologies to build ontologies and claim to have identified good guidelines that may be applied in the Conceptualization, Knowledge Acquisition and Knowledge Representation phases. After doing the survey of methodologies and starting the conceptualization phase, they found several problems and needs: the lack of understanding of the domain terms and the lack of experts; the need to facilitate the ontology definition, using a formal representation language, by domain experts (users) that are not computer experts; and the need to structure the whole process of guidelines, tasks and support materials. To address these requirements the authors propose the following solutions: to obtain relevant concepts by processing written sources of knowledge, used as a guide for the next phases, using information retrieval and document structure processing techniques; to support human communication through conceptual structures, by representing knowledge by means of two layers: a user layer with an easy graphical language (CMAPs) and an internal layer with a formal representation language (Description Logics); to build up the ontology in an incremental manner, they define a refinement cycle based on the three main conceptual strategies, top-down, bottom-up and middle-out applied in different phases of the cycle. The tasks of the refinement cycle are repeated until all the participants reach a consensus for the semantic of the ontology.

[13] and [16] presents the DILIGENT that comprises five main activities of ontology engineering: build, local adaptation, analysis, revision, and local update. In DILIGENT methodology an initial ontology is made available and users are free to use it and modify it locally for their own purposes. This initial ontology is built by a small group of builders. There is a central board (the board should have a well balanced and representative participation of the different kinds of participants) involved in the process that maintains and assures the quality of the shared core ontology. This central board is also responsible for deciding to do updates to the core ontology. However, updates are mostly based on changes re-occurring at and requests by decentrally working users. Therefore the board only loosely controls the process. Due to the changes introduced by the users over time and the on-going integration of changes by the board, the ontology evolves. The ontology goes on being developed in an iterative and incremental manner. A central ontology exists together with several local ontologies, and the central ontology goes being readjusted in agreement with the local ontologies.

[1] and [2] propose a three-phased ontology construction procedure in which the knowledge engineer mediates between the differing conceptions experts or users may hold about a knowledge domain. This work approaches the question of the direct participation of the members of the organizations in the creation of the shared ontology. The procedure presented is derived from conflict mediation approaches and consists on three main phases: generation, explication and integration. The main objective of the procedure is the integration of contradictory knowledge and the establishment of a shared conceptualization as well as a sustainable ontological commitment among human users. In our perspective this is an interesting approach that addresses explicitly the social aspects of ontology development such as negotiation. The whole process described belongs to the ontology specification phase.

A quite recent methodology that addresses the question of the shared conceptualization is DOGMA-MESS [6]. The authors present DOGMA-MESS as a methodology that supports the process of organizational ontology engineering and the rapidly changing of collaborative requirements and the DOGMA-MESS (Meaning Evolution Support System) as a state-of-the-art system built on the DOGMA framework for scalable ontology engineering. The model suggested by the authors for Interorganizational Ontology Engineering (a generic model) is a conceptual model of the interorganizational ontology engineering process sufficiently specific, according to the authors, to derive and organize practical methodological guidelines, yet generic enough to represent and compare many different approaches and techniques from an application point of view. This model is the basis to the development of DOGMA-MESS methodology. This model shows that an interorganizational ontology consists of various related sub-ontologies. The engineering process starts with the creation of an upper common ontology, which contains the conceptualizations and semantic constraints that are common to and accepted by a domain. Each participating organization specializes this ontology into its own organizational ontology, thus resulting in a local interpretation of the commonly accepted knowledge. In the lower common ontology, a new proposal for the next version of the interorganizational ontology is produced, aligning relevant material from the upper common ontology and various organizational ontologies. The part of the lower common ontology that is accepted by the community then forms the legitimate upper common ontology for the next version of the interorganizational ontology.

[3] presents an overview of the main methodologies, tools and languages to build ontologies. In this work they present CO4 (Collaborative construction of consensual knowledge bases) as a single method that includes a proposal for collaborative construction. CO4 is a protocol to reach consensus between several KBs (knowledge Bases), which are organized in a tree. Its goal is for people to discuss and agree in the knowledge introduced in the KBs of the system. These KBs are built to be shared, and they have consensual knowledge, hence they can be considered ontologies. The user KBs does not obligatorily have consensual knowledge. Each group KB represents the consensual knowledge among its children (called subscriber KBs). A KB can subscribe to only one group. A human user can create several KBs (possibly subscribing to different group bases) representing different trends, and knowledge can be transferred from one KB to another. Also, it is possible that several human users share the same KB. When the users of a KB have enough confidence in a piece of knowledge of their KB, and they want to reach consensus about their knowledge with the rest of the users, the CO4 process is executed. The steps are repeated until all users accept the proposal, or some users definitively reject it. If a user makes a proposal that does not satisfy the other users, the users and the groups agreeing with the modification can add it to their KBs (see [7]).

[7] present CO4 and KA^2 as the methodologies for collaborative and distributed construction of ontologies. They say that the goal of the Knowledge Annotation Initiative of the Knowledge Acquisition (KA) community, also acknowledged as the (KA)2 initiative, is to model the knowledge acquisition community using ontologies developed in a joint effort by a group of people at different locations using the same templates and language. According the authors, KA^2 is an open-joint initiative where the participants are actively involved in the distributive ontological engineering

development. The ontology is generated with base in the knowledge introduced using the templates.

As we can see, few works exist in the area of collaborative ontology construction. Analyzing the specification phase of the main methodologies we verified that this question is still unsolved. Although some initiatives exist in this field, none of them seem sufficiently solid, complete and tested. The procedure presented by [1] and [2], the DOGMA-MESS methodology and DILIGENT methodology deserve in this area special attention.

A collaborative ontology specification process is defined as a set of practices and activities used to obtain a shared conceptualization of the domain with the participation of all stakeholders. Comprise the identification of the concepts to include in the ontology, their characteristics, definition and relationships, the knowledge organization and structuring using external representations independent of the implementation language and environment. In our opinion, a collaborative ontology specification/construction process should contain the following principles that should be considered the core values on which all of the collaborative ontology specification methods are designed: 1/ active participation of all interested parties; the process requires constant collaboration between the development team and the other stakeholders; 2/ propose efficient and effective methods to support the negotiation process, methods that support the consensus or agreement obtaining between groups of human actors about the ontology content; 3/ propose mechanisms that allow working with the several users perspectives presented; 4/ propose tools to support collaboration (communication, cooperation and coordination), for example, tools to support a graphic visualization of the contents proposed for the shared conceptualization during the negotiation process; 5/ propose a notation or language to be used by all to represent their perspectives (this can be supported by one tool); 6/ propose techniques for concepts/terms elicitation; and 7/ propose mechanisms that support the semantic and syntactic analysis of the ontology concepts, to guarantee a correct interpretation of the contributions of the several stakeholders, for example, how to work with situations where it exists the possibility of multiple definitions for the same concept (homonymy).

4 Collaborative ontology specification: comparative analysis

Considering the most representative approaches presented in the previous section, the activities of the specification phase of each one, the definition and principles by us proposed for collaborative ontology specification, we present in table 1 some aspects of the comparative analysis of the several approaches used in the collaborative ontology specification. Given the allowed number of pages for the paper we opted by presenting some of the criteria used in the comparative analysis, trying to present the one that we considered to be the most relevant.

Table 1. Analysis of the different approaches for collaborative ontology specification.

	"Two-layered approach to knowledge representation using conceptual maps and description logics" [9]	**On-To-Knowledge methodology [14], [15] and [4]**	**DILIGENT methodology [13] and [16]**	**"The knowledge mediation procedure" [1] and [2]**	**DOGMA-MESS methodology [6]**	**CO4 [3] and [7]**	**KA^2 [7]**
Actors	Clients; domain experts and knowledge engineers	Domain experts, User group representatives (some key people representing the interests of the user groups), knowledge engineers	Ontology users, domain experts, knowledge engineers, ontology engineers and control board editors	Knowledge engineer, human experts, direct or indirect ontology end-users	Core Domain experts, participating organizations (domain experts) and knowledge engineers	KB Users	Ontopic agents and ontology coordinating agents
Extraction of domain knowledge (concept elicitation techniques)	Interviews + Document Processing	Workshop + brainstorming techniques + competency questionnaires	Not specified	Brainstorming + use of automatic thesaurus generation tools	Templates	Not specified	Templates
Use of informal representation language to represent the actors proposals	Not proposed	Not proposed	Not proposed	Not proposed	Not proposed	Not proposed	Not proposed

Methods to reach consensus or agreements about the content should be included in the ontology		Repeat the refinement cycle until all the participants reach a consensus about the ontology semantics	Not proposed	Not proposed	Proposed the use of conflict mediation approaches	Not proposed	Steps 2, 3, 4 and 5 of CO4 protocol are repeated until all users accept the proposal, or some user definitively rejects it	Not proposed
Treatment ways of the several users perspectives		Not proposed	Not proposed	Allow the existence of local ontologies that result of the adaptation of core ontology by end-users	Not proposed	Each participating organization specializes the interorganizational ontology into its own organizational ontology, thus resulting in a local interpretation of the commonly accepted knowledge	Allow the existence of several KBs	Not proposed
Conceptual strategies use	**Top-down**	Interviews with the experts	Not proposed	Not proposed	Not proposed	Not proposed	Not proposed	Not proposed
	Bottom-up	Interviews with the clients	Not proposed	Not proposed	Not proposed	Not proposed	Not proposed	Not proposed
	Middle-out	Document Processing of the written documentation to extract the most relevant terms	Not proposed	Not proposed	The generation of terms is performed with a middle-out approach	Not proposed	Not proposed	Not proposed

5 Discussion

The comparative analysis presented in the previous section allows us to reflect about the current state of the approaches for collaborative ontologies specification. A conclusion to this analysis is that there is no completely mature approach to support this task.

The analysis led to conclude that the use of concept elicitation techniques is not consensual among the several proposals. There are presented structured techniques (templates and interviews, e.g.) and no structured techniques (simple sending of messages, e.g.). Will the structured techniques the right approach in what concerns collaborative specification? Of the presented proposals, workshops and brainstorming seem to be the more adjusted. No approach proposes tools to support the decision process of the concepts to include in the shared conceptualization. What decision criteria to use? For instance, something as simple as, in situations where agreement or consensus doesn't exist relatively to a concept, where several proposals exist, a system can support the decision, for example, showing the result of the use of the several concepts proposed in documents produced by the involved organizations. The extraction of knowledge concepts of the domain from organizational documents and systems to complement the capture of knowledge performed joint the human actors was not considered relevant for the great majority of the approaches. The use of techniques for graphic representation of the reached conceptualizations was little explored. The use of informal languages to represent the several proposals presented by the human actors, as well as the possibility of creating automatically a formal specification of the ontology and respective code based on these informal models was not yet explored. The use of informal languages, in our opinion, will help the visualization of the different perspectives proposed by the several actors and the obtaining of consensus or agreements during the ontology conceptualization phase. Few proposals approach the question of the reutilization of existent ontologies in the development process. However, this reutilization can make the ontology content completer and richer. The creation of support documentation as a result of the specification phase is out of the goals of the majority approaches proposed, that also hinders the future reuse of the resulting ontology, because most of the times this only exists codified. The social aspects involved in the ontologies construction have not been factors considered. The main concern of the existent approaches is focused on the engineering tasks, leaving to second plan the social questions. Up to now it was given little attention to the methods to support the negotiation process among human actors. However, analyzing the social questions involved in the ontologies specification, such as the need for approaches to support in the consensus or agreements obtaining, as well as ways to treat the different perspectives presented for the several users, the answer to the questions may pass through a detailed study of the techniques and strategies proposed by the social sciences for the consensus and agreements construction and the appropriate choice of them. In a large part of the approaches the generation of terms is accomplished without the resource to any conceptual strategy. The subjects related with the linguistic representation of the knowledge (semantic and syntactic analysis of the concepts and relationships to include in the ontology), that can help in the negotiation of meanings among human actors continue without being explored. The semantic analysis that can be defined as a method for elicitation and knowledge representation about organizations, in the perspective of the cognitive semantics (part of the cognitive linguistics) has been forgotten. However, the study of linguistic methods (cognitive semantics) during the knowledge elicitation can be a road to proceed. These models of the cognitive semantics can support in the consensual specification of the meaning and terms for the ontologies development, to support in the negotiation of meanings among human agents that belong to different communities and to establish consensus in a community that needs to adopt a new term (concept). Theories and approaches as conceptual blending theory, image schema theory, idealized cognitive models, conceptual metaphor theory, mental space theory, among others can have an important role in the negotiation of meanings, in the definition of the concepts to include in the ontology and in the generation of new concepts.

6 Conclusions

This reflection allows us to make an analysis on the current state of the approaches for collaborative ontology construction. Our main conclusion is that there is a long road to travel in this area. Up to now, there are few detailed proposals for the collaborative construction of ontologies in (distributed) groups of human actors. Some subjects that we intended to continue studying are: 1/ techniques of informal representation of the different perspectives presented by those involved and results reached during the collaborative process of ontology conceptualization; 2/ application of social sciences approaches to support in the consensus or agreements obtained about the content that should be included in the ontology and in the definition of ways to treat the different perspectives presented by users; 3/ approaches and theories of the cognitive linguistics (cognitive semantics) to support the consensual specification of the meaning and terms (concepts) to include in the ontology; 4/ creation of tools that support the knowledge conceptualization/specification and that starting from the reached conceptualization generate an ontology requirements specification document and the code of the ontology. These tools should support all the collaboration (all interaction existent) among the participants, as well as the whole negotiation process. Some of the ideas to explore were presented already in the discussion section.

Our main goal is to develop a tool to allow the creation of a shared taxonomy/ontology, developed almost exclusively by their users, quickly and efficiently.

References

1. Aschoff, F.: Knowledge mediation: A procedure for the cooperative construction of domain ontologies. Master's thesis, University of Heidelberg (2004)

2. Aschoff. F. R., Schmalhofer, F., van Elst, L.: Knowledge mediation: A procedure for the cooperative construction of domain ontologies. InProceedings of the ECAI-2004 Workshop on Agent-mediated Knowledge Management (AMKM-2004) (2004)
3. Corcho, O., Fernández-López, M., Gómez-Pérez, A.: Methodologies, tools and languages for building ontologies: where is their meeting point? Data Knowl. Eng., vol. 46, n. 1, pp. 41--64 (2003)
4. Davies, J., Duke, A., Sure, Y.: Ontoshare: a knowledge management environment for virtual communities of practice. In K-CAP '03: Proceedings of the international conference on Knowledge capture, pp. 20--27, New York, NY, USA. ACM Press (2003)
5. Devedzic, V.: Understanding Ontological Engineering. Communications of the ACM, vol. 45, pp. 136--144 (2002, April)
6. de Moor, A., Leenheer, P. D., Meersman, R.: Dogma-mess: A meaning evolution support system for interorganizational ontology engineering. Proc. of the 14th International Conference on Conceptual Structures (ICCS 2006), Aalborg, Denmark, July 17-21 (2006)
7. Fernández-López, M., Gómez-Pérez, A.: Overview and analysis of methodologies for building ontologies. The Knowledge Engineering Review, vol. 12, n. 2, pp: 129--156 (2002)
8. Gasevic, D., Djuric, D., Devedzic, V.: Model Driven Architecture and Ontology Development. Springer-Verlag, Berlin (2006)
9. Gómez-Gauchía, H., Díaz-Agudo, B., González-Calero, P.: Two-layered approach to knowledge representation using conceptual maps and description logics. In Concept Maps: Theory, Methodology, Technology. Proc. of the First Int. Conference on Concept Mapping A. J. Cañas, J. D. Novak, F. M. González, Eds. (2004)
10. Gómez-Gauchía, H., Díaz-Agudo, B., González-Calero, P.: Towards a pragmatic methodology to build lightweight ontologies: a case study. In Procs. of the IADIS International Conference, Applied Computing 2004, Lisboa, Portugal (2004)
11. Gruber, T. R.: A translation approach to portable ontology specifications. Knowledge Acquisition, vol. 5, pp. 199--220 (1993)
12. López, M., Gómez-Pérez, A., Sierra, J., Sierra, A.: Building a chemical ontology using methontology and the ontology design environment. IEEE Intelligent Systems, vol. 14, n. 1, pp. 37--46 (1999)
13. Pinto, S., Staab, S., Sure, Y., Tempich, C.: Ontoedit empowering swap: a case study in supporting distributed, loosely-controlled and evolving engineering of ontologies (diligent). In 1st ESWS 2004. Springer (2004)
14. Staab, S., Studer, R., Schnurr, H.., Sure, Y.: Knowledge processes and ontologies. IEEE Intelligent Systems, vol. 16, n. 1, pp. 26--34 (2001)
15. Sure, Y.: A tool-supported methodology for ontology-based knowledge management. In: ISMIS 2002, Methodologies for Intelligent Systems (2002)
16. Vrandecic, D., Pinto, H. S., Sure, Y., Tempich, C.: The diligent knowledge processes. Journal of Knowledge Management, vol. 9, n. 5, pp. 85--96 (2005)

Trends on Adaptive Object Models Research

Filipe Figueiredo Correia[1,2] and Hugo Sereno Ferreira[1,3]

[1] ParadigmaXis — Arquitectura e Engenharia de Software, S.A.,
Avenida da Boavista, 1043, 4100-129 Porto, Portugal
{filipe.correia,hugo.ferreira}@paradigmaxis.pt
http://www.paradigmaxis.pt/

[2] FEUP — Faculdade de Engenharia da Universidade do Porto,
Rua Dr. Roberto Frias, s/n 4200-465, Porto, Portugal
filipe.correia@fe.up.pt
http://www.fe.up.pt/

[3] MAP-I Doctoral Programme in Computer Science,
hugo.ferreira@di.uminho.pt
http://www.map.edu.pt/i

Abstract. An Adaptive Object Model (AOM) is a meta-modeling dynamic technique, where a runtime model is used in order to allow for fast prototyping and model experimentation. It uses several levels of abstraction but differs from generative approaches and reflection in it's degree of dynamism and application domain.
We present a set of common AOM-related design patterns, along with several open issues. We also present the current version of Oghma, an AOM-based system that aims to become a framework for information systems development. Our intent was to compare Oghma with other systems of this sort. We believed some of Oghma's solutions belong to the current state of the art, but also that some benefit could be taken from other researcher's experiences with AOMs.
We have verified our beliefs to some extent, and briefly documented some of Oghma's solutions that we have not yet found applied to other AOMs. However, Oghma is still not close of being a comprehensive solution.

Key words: Adaptive object models, AOM, Model driven engineering, Design patterns, Meta-modeling, UML virtual machine, Oghma

1 Introduction

Creating abstractions has been a recurrent solution in the process of building software systems, allowing developers to direct more attention to software design instead of the idiosyncrasies of the platform being used. Model Driven Engineering (MDE) takes abstractions further, focusing on abstracting particular business domains, rather than only technology related issues [1]. Using this approach, domain models may play an important role on the process of requirements engineering, but their usefulness is also extendable to other software engineering activities.

A lot of current MDE efforts concentrate on model transformations, namely, the generation of implementations, and other artefacts, that effectively support developer's work. Generative approaches cover some typical pitfalls that appear when using MDE, they allow for increased reuse and fewer bugs, easier to understand systems, up to date documentation, a shorter time-to-market and they help making systems that are easier to change [2]. As such, models have proven to be very useful also at software design time, and not only during requirements engineering activities. Their usefulness, however, is also extendable to further software stages, as we will see.

Software requirements change increasingly faster, as organizations have to frequently adapt their business processes to different realities, or they acquire new knowledge that lead to different ways of understanding their business and, therefore, what they expect from systems used to support it. Software systems are called upon being adaptive to these new perceived realities, something that traditional systems are not good at, but models, meta-models, and meta-data in general, may be used in this regard. As been said in [3], in the context of models, *MetaData is just saying that if something is going to vary in a predictable way, store the description of the variation in a database so that it is easy to change. In other words, if something is going to change a lot, make it easy to change.*

Generative approaches to modeling are usually confined to static usage, while runtime model-based adaptivity brings an additional advantage, namely, it greatly reduces the time taken to put a new, or modified, model into execution. It thus allows for rapid prototyping, supporting model experimentation and innovation [2], [4]. Another difference is that runtime models make model semantics explicitly available at runtime. Code generation also makes model semantics available at runtime, but in an implicit way, encoded into the generated code.

The Adaptive Object Model (AOM) architecture allows for runtime adaptivity. It consists on using a meta-model as a first-class model; classes, attributes, relations and behaviour are represented and stored as data. At runtime, this information is interpreted, instructing the system which behaviour to take. Changing the model data results on the system following a different domain model and a different behaviour [2], [4], [5].

This paper presents previous research results on AOMs, relating them to the development of Oghma; a system based on an AOM that is currently being developed at ParadigmaXis. We expected to find solutions better than those we've achieved so far, from which Oghma would benefit, though we also believed some of our solutions would constitute contributions to the state of the art.

We will start by showing the role of abstraction in AOMs (section 2). Section 3 describes what is at stake when designing AOMs, along with related design patterns. It also presents the concrete example of the Oghma system, highlighting some topics we believe to be of particular relevance. Section 4 concentrates on future work using two different perspectives, namely, open issues on AOM systems in general, and issues that will soon be addressed on the context of the Oghma system. Finally, in section 5 some concluding remarks are made.

2 Abstraction

The concept of abstraction, in the sense of object-oriented design, plays an important role in AOMs. There are several levels of abstraction in use in an AOM, which frequently makes them difficult to understand [5]. We will see how AOMs fit among other techniques that also take advantage of multiple abstraction levels, better explaining the differences and similarities between them.

2.1 Level of Abstraction

Object-oriented (OO) languages provide two levels of abstraction, namely, class level, and instance level, which correspond respectively to compile-time and runtime activities. OO systems are bound to these levels, although more conceptual levels can be considered and implemented using these two levels alone [4].

The Meta Object Facility (MOF) is a standard from de Object Management Group (OMG) [6], based on the Unified Modeling Language (UML) and supporting model driven engineering. It provides four modeling levels (M3, M2, M1 and M0), which define an architecture for MOF-based systems, each level describing the next lower one. M3 models constitute meta-meta-models, M2 is used to define meta-models, M1 handles class-level elements and, finally, M0 corresponds to concrete instances [2], [7].

All four levels are useful when taking a meta-modeling approach, as more than the two levels supported by the OO paradigm (M1 and M0) are needed to account for the additional abstraction levels that meta-modeling requires. Meta-modeling is a fundamental concept when building AOMs, and MOF is one way of supporting it [4].

2.2 Reflective and Meta-modeling Techniques

The before mentioned generative and adaptive approaches are, respectively, static and dynamic approaches to meta-modeling. Reflection, like the use of AOMs, is also an adaptive technique, it is a process by which software can alter it's own execution using meta-information about it's structure. Comparing reflection with AOMs, both techniques have the concern of introducing flexibility by allowing dynamic behavior, but reflection has a wider scope, acting at the language level and using meta-information in an ad hoc way, while AOMs act at the business domain level and use meta-information in a well structured fashion [4], [5], [7].

3 The Design of AOMs

Figure 1 shows the basic design of an AOM, as described in [4], [5] and [8].

Two different levels are presented, a *meta* level and an *operational* level. While the former is used to define new types of objects, their respective attributes, relations and behaviors, the later is used to represent concrete objects, attributes and relations, and to enforce the defined behaviors.

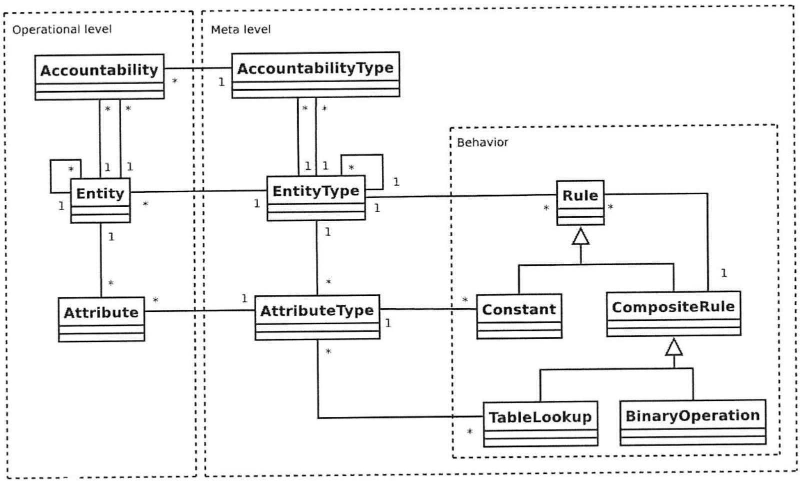


Fig. 1. Basic design of an Adaptive Object Model.

3.1 Patterns

When building AOMs, there are some typical issues that arise, as well as typical solutions for those very issues, which have been documented as *design patterns.*

A design pattern is a good solution for a recurring design problem. It's not meant to be a concrete solution, rather, it's meant to be a generic one that can be instantiated for a given type of problem, considering different contexts where it may arise. Solutions are presented in terms of interactions between elements of object-oriented design, such as classes, relations and objects [9].

There are many patterns useful in the context of an AOM, but the following ones are considered to be the most important when defining the essence of what an AOM is [4], [5].

Type Object. As described in [10], a *TypeObject decouple instances from their classes so that those classes can be implemented as instances of a class. Type Object allows new "classes" to be created dynamically at runtime, lets a system provide its own typechecking rules, and can lead to simpler, smaller systems.*

Property. The *Property* pattern gives a different solution to class attributes. Instead of being directly created as several class variables, attributes are kept in a collection, and stored as a single class variable. This makes it possible for different instances, of the same class, to have different attributes [11].

Type Square. The combined application of the *TypeObject* and *Property* patterns result in the *TypeSquare* pattern [11]. It's name comes from the resulting layout when represented in class diagram, as show in figure 1, with the classes *Entity*, *EntityType*, *Attribute* and *AttributeType*.

Accountability. Is used to represent different relations between parties, as described in [12], using an *AccountabilityType* to distinguish between different kinds of relation.

Composite. This pattern consists of a way of representing part-hole hierarquies. Is can be seen into practice in figure 1 with the *Rule* and *CompositeRule* classes [9].

Strategy. *Strategies* are a way to encapsulate behaviour, so that it is independent of the client that uses it. *Rules* are *Strategies*, as they define behaviour that can be attached to a given *EntityType* [9].

Rule Object. This pattern results from the application of the *Composite* and *Strategy* patterns, for the representation of business rules by combining simpler elementar constraints [13].

Interpreter. An AOM consists of a runtime interpretation of a model. The Interpreter pattern is used to extract meaning from a previously defined user representation of the model [9].

Builder. A model used to feed an AOM-based system is interpreted from it's user representation and a runtime representation of it is created. The *Builder* patter is used in order to separate a model's interpretation from it's runtime representation construction [9].

A lot of other patterns are used when building AOMs, though. Related issues like persistence, user-interfaces (UIs) and models maintenance can take great advantage of existing knowledge described as design patterns.

The patterns in [14], presented next, focus specifically on UI rendering issues when dealing with AOMs. In traditional systems, data presented in UIs is usually obtained from business domain objects, which are thus mapped to UI elements in some way. In AOMs business objects exist under an additional level of indirection, which has to be considered. In fact, it can be taken into our advantage, as the existing meta-information, used to achieve adaptivity, can be used for the same purpose regarding user interfaces. User interfaces can this way be adaptive to the domain model in use.

Property Renderer. Describes the handling of user-interface rendering for different types of properties.

Entity View. Explains how to deal with the rendering of *EntityTypes*, and how *PropertyRenderers* can be coordinated for that purpose.

Dynamic View. Approaches the rendering of a set of entities considering layout issues and the possibility of coordinating *EntityViews* and *PropertyRenderers* in that regard.

This growing group of patterns together describe a set of good practices for AOMs or, in other words, they constitute a *pattern language* for AOMs [15].

The following six categories include the patterns mentioned before, and where used while defining the pattern language presented in [15].

Core. This set of patterns constitute the basis for an AOM-supported system. The patterns included in this category are *Type Square*, *Type Object*, *Properties*, *Accountability*, *Value Object*, *Null Object* and *Smart Variables*.

Creational. These patterns are the ones used for creating runtime instances of AOMs: *Builder*, *AOM Builder*, *Dynamic Factory*, *Bootstrapping*, *Dependency Injection* and *Editor / Visual Language*.

Behavioral. Behavioral patterns are used for adding and removing behaviour of AOMs in a dynamic way. They are *Dynamic Hooks*, *Strategy*, *Rule Object*, *Rule Engine*, *Type Cube* and *Interpreter*

GUI. User-interface rendering patterns have already been mentioned: *Property Renderer*, *Entity View*, *Dynamic View*. Related to UI there's to add the *GUI Workflow* pattern.

Process. Includes the patterns used in the process of creating AOMs. An AOM has usually much of a framework in it. The following patterns are good practices when building a framework as well as when building an AOM: *Domain Specific Abstraction*, *Simple System*, *Three Examples*, *White Box Framework*, *Black Box Framework*, *Component Library*, *Hot Spots*, *Pluggable Objects*, *Fine-Grained Objects*, *Visual Builder* and *Language Tools*.

Instrumental. Patterns that help on the instrumentation of AOMs, namely, *Context Object*, *Versioning*, *History* and *Caching*.

3.2 The Oghma System

Oghma is a system based on an AOM. It is being developed at ParadigmaXis with the purpose of creating a framework for the development of information systems, although it hasn't yet been subjected to enough real-world cases in order to reach that status. It's development started without knowledge about existing research in AOMs and, as such, not all the design solutions employed match solutions described in literature on this topic, although a lot of them do. We find Oghma will benefit from some of these solutions, but we also believe some of the solutions in our system will constitute contributions to the current state of the art.

A detailed description of the system is outside the scope of this paper, but will certainly be further described in a future one. We will, however, highlight some of it's characteristics, which we haven't yet seen discussed to this extent in other AOM-related literature.

Modeling language. UML, as an executable language, presents some difficulties. Supporting it in a way that any UML model may be used by an AOM is a difficult task, as the UML specification is not formal, in order to be executed in a concise, standard way, and the several model types are not always seamlessly connected. As such, being a complete UML virtual machine is not one of the purposes of Oghma, although it uses a subset of UML, that allows for enough expressiveness.

Previous work exists on UML-based AOMs [2], but few details have been given about the extent of the supported UML specification. Oghma supports common AOM meta-model elements such as Classes, Attributes and Relations, along with relations' Cardinalities, but it also supports some UML-specific concepts, like Interfaces, Associative Classes and Navigability. These structures have shown to greatly simplify executing models that had been previously created using UML, while not over complexifying our models by trying to cover all the details in the UML specification.

Persistence. Persistence has before been pointed out as a typical issue when building AOMs. The most simple form of persistence may be achieved by using an object-oriented database, although using a relational database is also possible, in spite of the impedance mismatch between the relational and object-oriented worlds [2], [5].

The way to achieve persistence, when comparing with other Object-Relational Mapping (ORM) approaches [16], may be simplified in AOMs. In Oghma's case, a relational database is used. A runtime relational meta-model was conceived, along with rules to map between it and the existing runtime object-oriented meta-model. In this way we are combining model transformation techniques, common in generative approaches, with the dynamic technique which is an AOM.

Client-server. Oghma has a client-server architecture, and both kinds of nodes (clients and servers) take advantage of the possibilities offered by the underlying AOM. The way clients and server communicate with each other can be made independent of the fact an AOM is being used or not, but we have found that, the existing meta-model, allows the schema of messages, exchanged between clients and the server, to also be made adaptive.

The server accepts requests for both meta-level elements and operational-level elements. Allied with the fact that REST/XML (over HTTP)[17] was used as a communication architectural principle, we have obtained a server interface that is simple to use and constitutes a general purpose API, available for establishing interoperability with other systems. Using REST/XML over HTTP has also some additional advantages, namely, it simplifies debugging, provides caching mechanisms, and makes available standard ways of dealing with authentication and communication security.

Queries. Queries in the context of object-oriented environments have been addressed before in different perspectives [18], [19].

In Oghma, the way data is persisted is completely hidden from the server interface. In order not to break that abstraction, a querying model was designed, that allows queries to be defined in an object-oriented way. Instances of this object-oriented querying model can be transformed into an analogous relational-oriented querying model, in a similar way the relational meta-model is transformed to the object-oriented meta-model, and vice-versa. The relational-oriented querying model is directly translatable to SQL code, which is used to actually execute the intended query.

Because data is exchanged between the client and server in a RESTfull way, queries fit into this communication architecture encoded into URLs, and query results are returned as a set of resources.

Addressing. Something that directly derives from the adopted RESTfull communication architecture, is the fact that (meta-level and operational-level) model elements are seen and made available as resources. As such, by using the meta-model information, an adaptive and URL-based resource addressing scheme was defined. Considering an hypothetical model, the next example would obtain the model schema for "laptop" element types:

```
http://oghma.paradigmaxis.pt/computer/laptop/@schema
```

These examples would return existing information for a specific laptop, and a list of all of it's parts, respectively[4]:

`http://oghma.paradigmaxis.pt/computer/laptop/4A3615F1-5A91-22E4-0B1D-1416F93D4412`

`http://oghma.paradigmaxis.pt/computer/laptop/4A3615F1-5A91-22E4-0B1D-1416F93D4412/parts`

As mentioned before, queries also take part of the addressing scheme. The following example consists on a query that returns all the instances' of laptop computers bought before 2005:

`http://oghma.paradigmaxis.pt/computer/laptop[yearbought lt 2005]`

Business rules. Business rules in the context of AOMs are frequently made as pluggable components, based on the Strategy design pattern (see section 3.1), and these components' implementation vary according to the domain in use [11], [5], [4]. Oghma's business rules don't follow this approach, as they are added to the model in a declarative way, making them simpler to define and more reusable, although less powerful than using Strategies.
The runtime model enforces these business rules, and it does so both on the client and server sides. On the server side, business rule enforcement is done to ensure semantic integrity according to the model, while on the client side it is done to validate user input, giving quick feedback to the user and avoiding roundtrips to the server as much as possible. Validations exclusively on the client side are not sufficient, as the server is used concurrently by multiple clients, but also because it may be used by third party client software as well, which may not fully validate their input data.

User-Interface. Adaptivity is a pervasive concept when it comes to AOMs, and it reaches user interface issues too. Some solutions have recently been documented [14] that take an adaptive approach to UI issues in AOMs (see section 3.1), but some additional advancements can be found in Oghma's approach. Namely, PropertyRenderers are used to present not only value-properties (attributes), but also instance-properties (relations). When adapting the interface for a specific context, PropertyRenderers are chosen based on several meta-model characteristics. For value-properties, the kind of the AttributeType, as well as related business rules are used to determine which renderer should be applied. For instance-properties, the cardinality and navigability are used for the same purpose; there are specialized renderers for one-to-many, many-to-many and one-to-one relations. Has mentioned before, user-interface feedback is also based on business rules.
System navigation is also taken into account. Types may be declared as Entry Points, and modeled as belonging to specific Subsystems. Doing so makes such Types directly accessible from the system's menu, under the established subsystem structure.

[4] There is another important detail, that although not a direct consequence of the addressing scheme, shows in these examples: all instances are identified by Global Unique Identifiers (GUIDs). This makes decentralization easier; clients can create new objects, with their respective identifiers assigned, without having to request them to the server.

4 Future Work

Advantages of using AOMs stand out when using a domain model that changes frequently, but these advantages come at a cost. A lot of issues remain to be solved in their design and development, making them a fertile research area.

We will first address common open issues in AOMs, followed by some areas we will specifically like to explore in Oghma, and that we believe may also be of interest in the context of other AOM-based systems.

4.1 General Open Issues in AOMs

AOMs generally require a higher initial development effort, as they are more complex than traditional systems. This complexity makes them also harder to understand, specially by those who haven't had previous contact with this kind of architecture, and thus, they may be harder to maintain. Although AOMs are *adaptive* to model changes, they are not easily *adaptable* to new functional requirements. It is important to consider the degree of adaptivity that is in fact needed when starting the development of an AOM, as the introduced flexibility will tend to increase the system's complexity [5], [20], [21].

Model maintenance may also be an issue. Using Visual Editor tools or Domain Specific Languages (DSLs) may support model creation and later modifications, but these tools have usually to be developed from scratch. When developing a language, as when developing a DSL, other needs also arise, such as debuggers, version control and documentation [5]. However, we do believe the use of standard languages and tools may ease these issues.

Also related to modeling, running systems are limited to the expressiveness of the modeling language used, specially concerning behavior modeling, and it is an open issue to find the right level of abstraction the model should have [2].

Model evolution should also be taken into account. As models evolve, instances created according to the previous model version have to be migrated, which may require intervention of external tools or, alternatively, will require the system to handle different model versions simultaneously [2].

Being part of the production system, an AOM affects it's execution. AOMs are usually slower than traditional systems, and even other (generative) meta-modelling techniques [2], [5], [20]. The approach followed in [2] is an interesting combination of AOMs and generative techniques, at a cost of an additional initial development effort: an AOM is used for system prototyping and model experimentation, but code is generated for the production system.

Our own experience with Oghma has show us some of AOM's advantages and disadvantages, but it is not trivial to assess the entire impact of using an AOM over a traditional approach. Yet, we haven't found a concrete analysis of how software quality metrics are affected by the use of this architecture. We believe such study would be of great interest, and it would also ease the comparison between different AOM implementations, including the Oghma system.

4.2 Future Work in Oghma

In the context of Oghma, the following AOM-related concerns would be of most interest to explore in the future.

Meta-model transformations. The transformation process between the object-oriented and relational-oriented models, as described in section 3.2, is done in a monolithic way. This process could be improved by taking a more modular approach, using a set of rules that together define the transformation, which would allow to prove the bijectiveness, or injectiveness, of transformations. It will also be useful to take this same approach towards the transformations between object-oriented models and their xml-oriented representation, which is also currently made in a monolitic manner.

Model evolution. Migration of object instances, between different model versions, is an issue we believe may be solved, to a certain degree, by taking benefit from the respective meta-models.

Elements from the source model will have to be mapped to the respective elements from the target model, but this mapping can be a complex process, considering the number of differences that may exist between two given models. Current knowledge on refactorings [22] and database evolutionary transformations [23] can be used in this regard. By simplifying complex model migrations into a more restricted set of standard object-oriented *refactorings*, we obtain an additional level of abstraction, from which a model-migrations-oriented language can be developed. The developer would be able to use this language to conceive a migration process between two given models. The next step would be to assist the developer in more easily creating these migration processes by automatically identifying model refactorings from the source and target models. In the best case scenario, the described assistant would be able to identify the entire process with minimal intervention from the developer, although that may not be possible for models very different from each other.

Ontology. An ontology is a form of knowledge representation[5]. It is a data model, and it can be used to describe classes, attributes, relations and events. This makes ontologies very similar to the subset of UML that Oghma uses, and thus, makes them a possible candidate to replace the current Oghma meta-model.

We believe ontologies may provide a richer and more formal meta-model than the one currently used by Oghma. It would also serve the purpose of standardizing the XML dialects used for model representation as well as for communication between the server and its clients. Well established formats such as the Ontology Web Language (OWL) can be used for this purpose, instead of the XML dialects we have developed [24], [25].

[5] We refer to the concept of ontologies in context of computer science, as it has a different meaning in the context of philosophy.

5 Conclusion

By using Model Driven Engineering (MDE) developers can more effectively focus on business domain modeling, as well as the modeling of more technology-related concerns. Generative approaches have shown to be very helpful in this context, but they are not appropriate for fast prototyping and model experimentation, while Adaptive Object Models (AOMs) present a way to achieve these objectives.

A characteristic of AOMs is that they use several levels of abstraction, as other techniques also do. However, AOMs are dynamic in nature (while generative approaches to meta-modeling are not) and act at the business domain level (while reflection act at the language level). AOMs take into account an operational level and a meta level. The first works at the instance level, while the second works at the class level.

Design patterns are a valuable resource when conceiving AOMs. Several patterns are used in the basic underlying architecture of an AOM, and many more can be used to solve related issues, such as persistence and user-interface adaptivity, among others. The set of patterns, that describe the ways an AOM may be designed, form a Pattern Language for AOMs.

The Oghma system is based on an AOM. It aims to be a framework for information systems development, and uses approaches that we believe to be of notice, namely, on the areas of the modeling language used, persistence, client-server interaction, queries and business rules enforcement.

AOMs still have a lot of unsolved related issues, although some of them have been being addressed on AOM's literature. Oghma also has a long way to go before we may see it as a comprehensive solution, and not all of the topics we will like to explore have already been covered before, in the context of AOMs. We believe we will find such topics to be fruitful research paths.

References

1. Schmidt, D., Schmidt, D.: Guest editor's introduction: Model-driven engineering. Computer **39** (2006) 25–31
2. Riehle, D., Fraleigh, S., Bucka-Lassen, D., Omorogbe, N.: The architecture of a UML virtual machine. In: OOPSLA '01: Proceedings of the 16th ACM SIGPLAN conference on Object oriented programming, systems, languages, and applications, Tampa Bay, FL, USA, ACM (2001) 327–341
3. Yoder, J.: Adaptive object models and metadata definition (2008) http://www.adaptiveobjectmodel.com/Define_Adaptive_Object_Models.html Accessed January 5, 2008.
4. Revault, N., Yoder, J.W.: Adaptive object-models and metamodeling techniques. In: ECOOP '01: Proceedings of the Workshops on Object-Oriented Technology, London, UK, Springer-Verlag (2002) 57–71
5. Yoder, J.W., Johnson, R.E.: The adaptive object-model architectural style. In: WICSA 3: Proceedings of the IFIP 17th World Computer Congress - TC2 Stream / 3rd IEEE/IFIP Conference on Software Architecture, Kluwer, B.V (2002) 3–27
6. OMG: OMG's metaobject facility (MOF) home page (2008) http://www.omg.org/mof/ Accessed January 5, 2008.

7. Costa, F.M., Provensi, L.L., Vaz, F.F.: Using runtime models to unify and structure the handling of meta-information in reflective middleware. Volume 4364., Springer Berlin / Heidelberg (2006) 232–241
8. Welicki, L., Lovelle, J.C., Aguilar, L.J.: Meta-specification and cataloging of software patterns with domain specific languages and adaptive object models. In: EuroPLoP, Irsee, Germany (2006)
9. Gamma, E., Helm, R., Johnson, R., Vlissides, J.: Design Patterns. Addison-Wesley Professional (1995)
10. Johnson, R., Woolf, B.: The type object pattern (1997)
11. Yoder, J.W., Balaguer, F., Johnson, R.: Architecture and design of adaptive object-models. ACM SIG-PLAN Notices **36**(12) (2001) 50–60
12. Fowler, M.: Analysis patterns: reusable objects models. Addison-Wesley Longman Publishing Co., Inc, Boston, MA, USA (1997)
13. Arsanjani, A.: Rule object: A pattern language for adaptive and scalable business rule construction. (2000)
14. Welicki, L., Yoder, J.W., Wirfs-Brock, R.: A pattern language for adaptive object models: Part i - rendering patterns. In: PLoP 2007, Monticello, Illinois (2007)
15. Welicki, L., Yoder, J.W., Wirfs-Brock, R., Johnson, R.E.: Towards a pattern language for adaptive object models, Montreal, Quebec, Canada, ACM (2007) 787–788
16. Fowler, M.: Patterns of Enterprise Application Architecture. Addison-Wesley Professional (2002)
17. Fielding, R.T. In: Representational State Transfer (REST). University of California, Irvine (2000)
18. ODMG: Object data management group home page (2008) http://www.odmg.org/ Accessed January 5, 2008.
19. Microsoft: The linq project (2008) http://msdn2.microsoft.com/en-us/netframework/aa904594.aspx Accessed January 5, 2008.
20. Dantas, A., Yoder, J., Borba, P., Johnson, R.: Using aspects to make adaptive object-models adaptable. In: RAM-SE'04-ECOOP'04 Workshop on Reflection, AOP, and Meta-Data for Software Evolution, Oslo, Norway (2004) 9–19
21. Crous, T., Danzfuss, T., Liebenberg, A., Moolman, A.: Adaptive object modelling using the .NET framework (2005)
22. Fowler, M., Beck, K., Brant, J., Opdyke, W., Roberts, D.: Refactoring: Improving the Design of Existing Code. Addison-Wesley Professional (1999)
23. Ambler, S.W., Sadalage, P.J.: Refactoring Databases: Evolutionary Database Design. Addison-Wesley Professional (2006)
24. W3C: Owl web ontology language overview (2004) http://www.w3.org/TR/owl-features/ Accessed January 5, 2008.
25. Knublauch, H.: Ramblings on agile methodologies and ontology-driven software development, Galway, Ireland (2005)

An Approach to Improve Speed and Objectivity in Audits

Paulo Alves

Faculdade de Engenharia da Universidade do Porto
Rua Dr. Roberto Frias, s/n 4200-465 Porto, Portugal
alves.paulo@fe.up.pt

Abstract. Yet at many companies today, there is still an enormous amount of inertia around implementing the tools, technologies, processes and training to meet increasingly in their business and innovation processes. This paper present and discusses a tool for Consulting, Audit company's, that made external audit or for any company that made internal audits. Our approach is intended to increase the speed of audit process and convert knowledge in capital, using mobile platforms such as PDA, TabletPC and Laptop. After testing in real world with some entities, proved that can reduce time of audit process.

Keywords: Mobile, Audit, Networking, Synchronization.

1 Introduction

The heavy competitive pressure of the market forces all competitors to design strategies of continuous adaptation to business environment, creating agile and flexible structures for responding, with the highest total quality level, to market demands. Each enterprise operates in the market as a node in the network of suppliers, customers, service providers and partners and to track them and not lose customers they need to improve their technologies and processes [1].

The main difficulty when we talk about audits is to get the same audit assessment changing auditor [2]. An auditor expert can bring capital to the enterprise and customers thrust, because he has the knowledge and experience. When an auditor expert goes way, the enterprise loses knowledge, customers and capital [2].

With the intent of answering the exposed problems, the demand is strong and because in the market the tools that exist are specialized only in one audit type [3][4][5]. The approach described in this paper had as objectives reduce audit process time, the customer must receive the final report faster and materialize the knowledge into a model. When tacit knowledge is converted into explicit knowledge with Information Systems a materialized knowledge was occurred, personal knowledge was transferred to the group or organization. The base of knowledge of auditors is the pyramid of an audit's company, and all the companies do not want to lose knowledge.

Nowadays, because the market competition is not always easy to keep an audit expert, or any kind of collaborator, and for do not lose their knowledge, this approach put the knowledge into a model (materialize the knowledge), which will be used in the audit creation. This model must be created always by one auditor expert or a team with experience in Audits, to create a good model.

One model can be applied to several audits, contains all questions and possible answers, and depending on the answers is requested justification and the possible justification is presented. In case of negative answers it's possible to allocate clues and corrective measures to the questions (items).

Instead to do an audit with checklist in paper, the auditors work with a PDA or TabletPC. At the end of the audit, the auditor synchronize the data with the server using a wireless connection (Internet, GPRS, etc), and can send a *pdf* with audit results to customer in the moment.

Using models which materialize the knowledge, audit effectiveness will be improved, variation between different auditing experts will be reduced, and will facilitate decision making during an audit.

The paper is structured as follows: Section 2 contains current method and related work. In the section 3 describe the development environment and the system developed. Section 4 contains the results, and section 5 contains conclusions, critical analysis and future work.

2 Current Method and Related Work

Some of the companies that support us do audits, and the process that they use can be seen in Fig 1. The process is based on 3 steps. First step, the auditor take the checklist, in paper, goes to the customer and execute the audit, by checking the list (questions) and answer that question taking notes by hand. Second step, the auditor goes to their office and passes to computer the audit checklist with notes and conclusions. After the report created is sent to the customer where is described the audit strengths and weaknesses. All the process can take for 15 to 30 days at least.

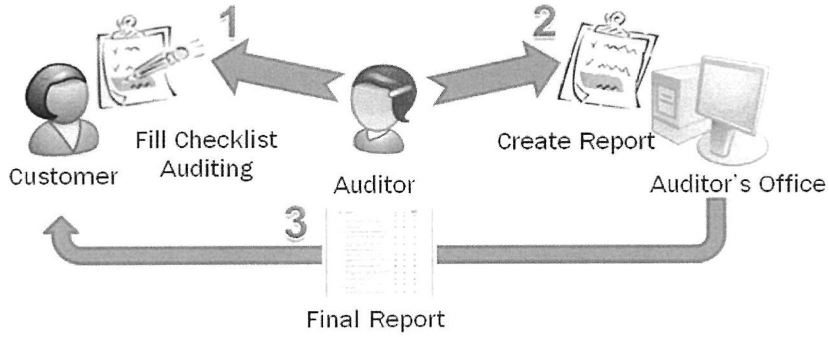


Fig. 1 - Current Audit method

In the market (National and International) there is tools for this purpose [3][4][5], but they are very specific, for example, HACCP (Hazard Analysis and Critical

Control Point) audits where includes Restaurants, Butchers, Bakeries, etc. Tools ago referenced, has been tested with some of audit models by a group of auditors, some are expert other no, and the results was different in some audit topics. The different results occur because these tools allow subjective answers. To create new models, or change the structure of them it is difficult because the systems are based on one owner model structure base.

In addition, *European Foundation for quality Management*, *Malcom Baldrige* (EFQM) [8] and *Prémio da Excelência – Sistema Português da Qualidade* (PEX-SPQ) [7], that are essentials management models for a certified company. PEX-SPQ is based on EFQM that is one of the best models for:

- Self-Assessment;
- Benchmark;
- Identify areas for Improvement;
- A common Vocabulary and a way of thinking.

3 Our Approach and Development Environment

3.1 Our Approach

Our approach to the problem was been based on possibility to create large models, very complexes, very detailed and can integrate management models like EFQM. This approach does not change much the auditing process, but changes the way the audit model is created in order to achieve the audit effectiveness, reduction of the variation between different auditing experts, and facilitate decision making during an audit.

For reduction of the variation between different auditing experts and to facilitate decision making during an audit it is essential remove the subjectivity of the answers. To remove subjectivity of the answers, the questions of the model need to be very detailed, simple and objective for the answers can be at atomic level. For example, to the question: "The table has meat on top?" the possible answers will be: "Yes" or "No". Removing subjectivity to the answers putting them at atomic level is to materialize the knowledge.

In some cases, the question does not make sense exists, for these cases coexisting, a new possible answer is added for example: "Not Applied". The auditors usually do not use answers like: "Yes" or "No", generally the question are at level of satisfaction so usually the answers are: "Satisfactory", "Not Satisfactory" and "Not Applied".

In Fig. 2 it is possible to see the process of auditing, since the auditor goes to the customer until the customer receives the final report. Similar to the older process, but instead of having a checklist (in paper), now the auditor have a mobile equipment. At the end of audit execution, the auditor can send a temporary final report (digital format) with audit results to the customer, where the customer can see the weaknesses, strengths and corrective measures recorded. This report is carried out on pre-defined report templates for the purpose. With the temporary report the customer can make the necessary changes immediately after the audit, and not need to wait a lot of days.

The final report, despite being able to perform in mobile equipment, it is done in auditor's office for validation, certification and to make an authenticated printed version.

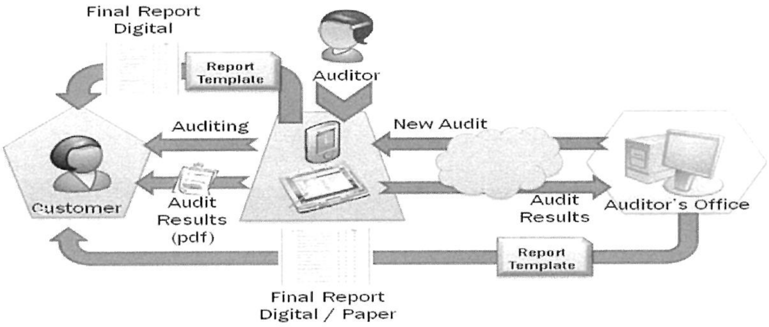


Fig. 2 - Audit process developed

In Fig. 3 is described which contains a model. A model contains all the questions and possible answers for each question for the audits. Depending on what answers is given, the system can ask to the user (auditor) for a justification, if during the model construction, the model constructor require a justification, it will not be possible answers without a justification.

In case of negative answers, it is possible to allocate clues and corrective measures to the questions (items). Clues have the goal of helping to find solutions to detected problems and relate problems with possible causes. The auditor as performs the audit, when find a problem can add Clues to that question and relatc it with other issues or questions. A Clue can be used for advice and to call attention for a topic or issue during then audit execution.

The Corrective Measures like Clues, when a problem is found, the auditor writes a corrective measure, if is applied, in order to solve the problem, based on legal solutions or not. Corrective Measures are classified on: Non-legal compliance or legal compliance. If occurs a corrective measure classified on Legal compliance, the auditor need to add which law refers that corrective measure and some description of the law.

There is a base of knowledge (lexical database) in the system, which contains all words written in the models (questions, answers, justifications, etc), with the aim of assisting in the creation new text by completing words or phrases. This base will grow up, until the administrator so wishes, because stores phrases, and there is a lot of possible combinations of words and phrases.

Audit Framework is a Framework[1] which contains re-usable components, interfaces, code libraries used in all development layers and it was all developed for this work.

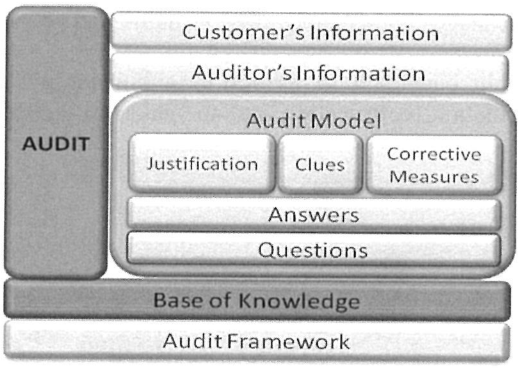


Fig. 3 - Core of Business Layer

Fig. 4 shows the overview of entire system. There is 4 ways that the user can operate the system. Eye2PDA and Eye2AuditTablet are the applications that the auditors use for auditing the customers. Eye2AuditWeb and Eye2AuditDesktop are used for audit management, to create or change models.

The Data Layer has been tested with Mysql, Oracle and Sql Server, all the companies are working with Sql Server 2000 or 2005 because already have the software.

For data synchronization between mobile equipment and the Server has created a module, part of Audit Framework, for confidentiality reasons do not will be described in this paper, which receives encrypted data from secure XML Web Service to Data Layer.

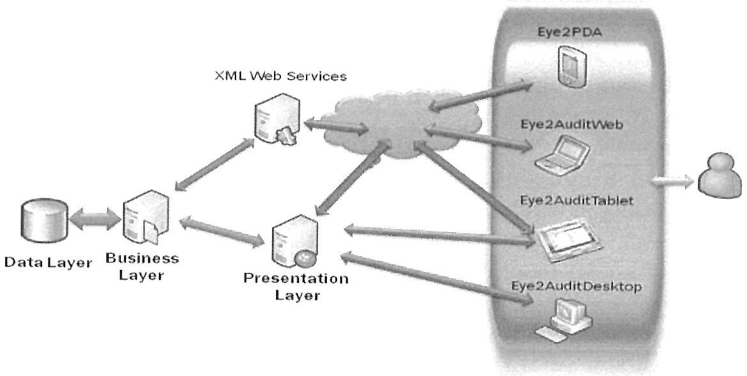


Fig. 4 - General view of the System

[1] A software framework is a re-usable design for a software system (or subsystem). May include support programs, code libraries, a scripting language, or other software to help develop and *glue together* the different components of a software project.

3.2 Development Environment

The main development tool used in this work was Microsoft Visual Studio 2005 [6], using C# language with .NET Framework 2.0. Was used .net framework because it´s a requirement of the companies, and .NET Framework offers a number of advantages like [6]:

- Consistent Programming Model;
- Direct Support for security;
- Simplified Development Efforts;
- Easy Application Deployment and Maintenance.

Was used the traditional n-tier application architecture, more information can be found on [9].

3.3 Mobile Transaction Processing

Database transaction processing conforms for several years now to the criteria of atomicity, consistency, isolation and durability (ACID). Techniques like two-phase commit (2PC) and locking (2PL) [10], in turn, are used by almost every transaction to achieve the atomicity and isolation properties and preserve the consistency of shared data. Two-phase commit protocol between the transaction manager and all the resources enlisted for a transaction ensures that either all the resource managers commit the transaction or they all abort.

Although 2PC guarantees the autonomy of the transaction, the required processing load is quite heavy, creating frequent update conflicts, especially when data is duplicated across multiple sites. Replication of data is a way to alleviate this conflict problem and is usable only when transaction-based update propagation is not required. Most distributed systems adopt these two methods in parallel to judiciously match the requirements of the application [11].

The basic Two-Phase Locking protocol is the most common locking protocol in distributed transactional systems to accomplish update synchronization and concurrency control. Often vendors combine concurrency control techniques like 2PL, consistency control techniques like 2PC, and timeout for deadlock resolution into a single implementation for global distributed transaction management [11]. With 2PL, a transaction execution consists of *two* phases. In the first phase, locks are acquired but may not be released. In the second phase, locks are released but new locks may not be acquired.

In mobile computing environments, transaction processing faces new challenges due to typical characteristics of wireless networks such as low bandwidth, frequent disconnections by mobile hosts (MH), very low processing power as well as limited storage capacity of the mobile devices.

Moreover, we adopt the assumption of [11] that handoff delays pose a severe challenge for database transactions, hence we recognize the need for a novel transaction model to counter their effects. In addition, the mobile devices that are used today operate as I/O and communication devices primarily with low processing capabilities and battery life, while they rely on proxies working on their behalf and residing at their mobile-support station (MSS) of the current cell.

A novel model for transaction execution in such environments may not use the traditional techniques of 2PC and 2PL, as transactions would only get a small fraction of useful work done due to frequent aborts which owe to network disconnections.

An effort towards this direction defines such a model (so called Kangaroo Transactions [12]) by building upon the concepts of split and global transactions, which ensures the successful execution of transactions despite the occurrence of handoffs, a reference models layers are given in Table 1. Following this model, a number of solutions have been proposed by other authors [13][14][15] that address issues related to roaming, disconnections, data availability and transaction throughput. Kangaroo Transactions is transaction model to capture the movement behavior of transactions in a multi-database environment where mobile transactions do not originate and end at the same site.

Table 1. Reference models layers (Kangaroo Transactions)

Layer	Location	Purpose
Source System	Fixed Host Base Station Mobile Unit	Provide services defined by specific software.
Data Access Agent	Base Station	Coordinate access to data in source system and facilitate recovery. Manage mobile transaction.
Mobile Transaction	Base Station Mobile Unit	Grouping of operations needed to perform user request initiated at a mobile unit.

While combining the requirements of security and mobility, we are also concerned with other relevant issues like concurrency and performance (of an individual transaction and an entire system too).

Fig 5 shows the effect of these factors on each other within the context of a mobile transaction. The dotted arrows denote undefined effects for which different views can be presented (we keep this discussion out of the scope of this paper). Though concerned with the security of shared data during the execution of mobile transactions, we are conscious not to do this at the cost of reduced concurrency and degraded performance (a more elaborate analysis can be found in [16]).

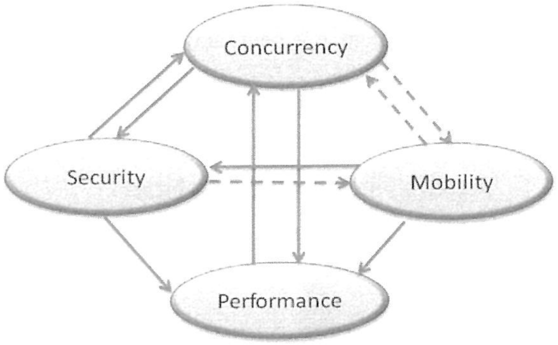


Fig. 2 - Interdependence among various factors in mobile processing

4 Experiments and Results

Experiments take place in Castelo Branco, in Aquimisa [17]. Aquimisa is a consulting company in Food Industry and is a laboratory analyses that provide services of assistance and control of quality. The work was developed in 2007 and ended in September.

After prepare the system in Aquimisa installations, which correspond to install SQL Server 2005 Database, install the application "Eye2PDA" on three PDAs (Qteck 9100), one "Eye2AuditTabletPC" on TabletPC (Asus R2HV) and one version "Eye2AuditDesktop" on Desktop PC for Audit Management and to create new models.

During the first month, 2 auditors began to perform the audits with the PDA and checklist in paper to compare with which method they were faster, to find possible problems with the application. At the beginning, they was faster to execute an audit with the checklist in paper (not prepare report) than in PDA, because they are not familiarized with PDA.

The three months later, they are already familiarized with the PDA, and they take the same time using the checklist in paper and the PDA, in this moment are 3 auditors working only with PDA. When they are auditing they need to see all items (questions), so is normal, that they having spent the same time with checklist and PDA.

Another part of the experiment was the creation of the final report to send to the client. With the Report template, the final report is automatically created, missing only introductions and conclusions, but the Strengths and weaknesses of the audit was already separated, which with the checklist in paper, they need to spend hours to separate one by one in the computer. The Report Template can create statistics based on present and old audits that belongs to the same customer, comparing the results and display corrective measures and advices to overcome the problems.

After testing in real world, was proved that can reduce audit process time like we can see in Fig. 6 (Data provided by Aquimisa), were has presented the same audit with different process. The process using the paper checklist subtitled "Before" and with PDA subtitled "After". An improvement has been achieved, by average, of one day and a half for half an hour.

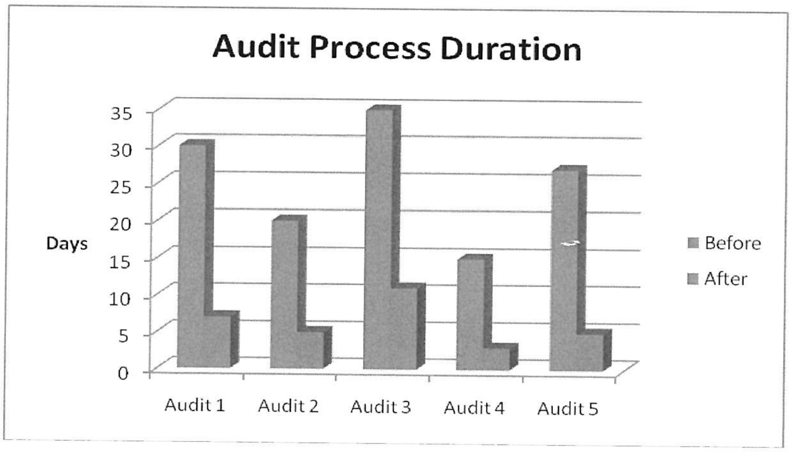


Fig. 6 - Compare Audit Executions (Aquimisa November 2007)

5 Conclusions and Future Work

5.1 Critical Analysis

Our approach was intended to increase speed to audit process and convert knowledge in capital, using mobile platforms such as PDA or TabletPC and new audit models. The speed of audit process was increased because the process to create the final report was optimized, so that at the end of the audit execution the final report was already prepared. The speed of audit execution was not increased because in checklist or in PDA have the same questions and answers, so it takes the same time.

Another objective was convert knowledge in capital. The tacit (implicit) knowledge has two dimensions: the technical and cognitive. The technical dimension concerns the practical knowledge to know execute a task. The cognitive dimension was based on schemes, mental models, beliefs and perceptions that reflect our image of reality (which is) in our vision of the future (which should be) [18]. The explicit knowledge is the knowledge formal, often encrypted in Mathematical formulas, rules, specifications, etc. It is that knowledge that can be formally expressed with the use of a system of symbols and based on objects and rules and can therefore be easily communicated or disseminated [18].

Convert the tacit knowledge to explicit knowledge, through Information Systems, we are transferring the individual knowledge to the group, to the organization. In this case we convert, transfer the tacit knowledge to an explicit knowledge and stores that knowledge into an audit model very detailed. And when we materialize the knowledge we are transforming them into a tool to be used by the organization to make profits, so we can consider that was converted knowledge into capital. If one organization loses an auditing expert, do not will lose all knowledge, because that knowledge was already materialized.

This objective is only achieved if the model created was well constructed, if the answers were at atomic level, i.e. there is atomicity in the answers, if the tacit knowledge was well converted to explicit knowledge.

An Audit with objective answers does not need a specialist Auditor, and is not sensitive to subjective answers. Consequently different auditors can obtain the same audit assessment.

5.2 Future Work

The approach presented, requires the creation of audit models very detailed, which requires time and costs. One way to recover the investment made on creating models, is get profit by selling owner audit models to other organizations.

It proposed an on-line platform (Fig 7), where companies, that have this tool, can share their audit models, not for free, but to take advantages of this tool and make profitable their audit models, their knowledge.

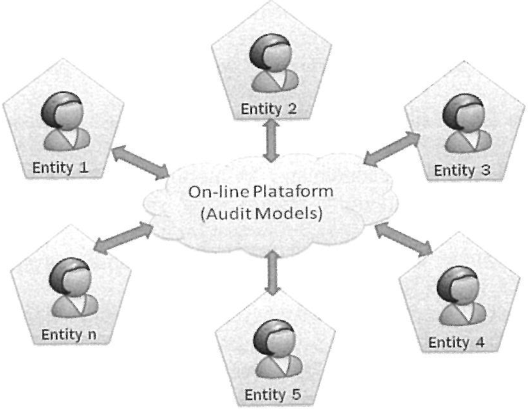


Fig. 7 - Online platform (Future Work)

With the platform it will be possible recover the investment, or part of it, when selling the models. When selling models, it is sharing audit models with other auditing experts, that they can find mistakes in the models and make improvements on them, and continuing materialize new knowledge coming from diverse organizations.

Acknowledgment

The authors would like to thank DeepEye [19] and Netsigma [20] for granting access to technologies, infrastructures and some expenses that they had with the project. A special thanks to C4G [21], for their availability and advice about evaluation systems. Finally, the authors would like to thank Aquimisa, especially to their auditors for all the support and willingness to test the tool in the real world.

References

1. C. H. Kim et al.: “A modelling approach for designing a value chain of virtual enterprise. *In International Journal of Advanced Manufacturing Technology”*. Springer-Verlag, Vol. 28, No 9, pp. 1025-1030, 2005.
2. Rita Teixeira d'Azevedo.: “Auditorias de Qualidade e/ou Ambiente: preparação e documentação”, Pluridoc, 2007.
3. http://www.haccpnow.co.uk/downloads.asp, Dec. 2007.
4. http://activequality-iso-9000-software.open-mind-solutions.qarchive.org/, Dec. 2007.
5. http://www.iglink.com.au/index.php/igl/software, Dec. 2007.
6. Microsoft Visual Studio, http://msdn.microsoft.com/vstudio/, Dec. 2007.
7. EFQM, http://www.efqm.org/Default.aspx?tabid=35, Dec. 2007.
8. PEX-SQP, http://www.ipq.pt/custompage.aspx?modid=1296, Dec. 2007.
9. Alexandre G. Valente.: Analysis of N-Tier Architecture Applied to Distributed-Database Systems, ISBN-10: 1423544536, Storming Media (1999).
10. G. Samaras, G.K. Kyrou, and P.K. Chrysanthis.: "Two-phase commit processing with restructured commit tree," in *Proc. Nat'l. Greek Conf. on Inform.*, pp. 82-99, *LNCS 2563*, 2003.
11. T. Imielinski, and B.R. Badrinath.: "Mobile wireless computing." *Commun. ACM*, vol. 37, pp. 18-28, Oct. 1994.
12. M.H Dunham, A. Helal, and S. Balakrishnan.: "A mobile transaction model that captures both the data and movement behaviour." *Mobile Netw. Appl.*, vol. 2, pp. 149-162, June 1997.
13. S.A. Patricia, C.L. Roncancio, and M. Adiba.: "Analyzing mobile transactions support for DBMS," in *Proc. DEXA*, pp. 595-600. *LNCS 2113*, 2001.
14. E. Pitoura, and B. Bhargava.: "Data consistency in intermittently connected distributed systems." *IEEE Trans. Knowl. Data Eng.*, vol. 11, pp. 896-915, Nov. 1999.
15. G.D. Walborn, and P.K. Chrysanthis.: "Supporting semantics-based transaction processing in mobile database applications," in *Proc. IEEE SRDS*, pp. 31-40, 1995.
16. P. Serrano-Alvarado, C.L. Roncancio, and M. Adiba.: "A survey of mobile transactions." *Distrib. Parallel Dat.*, vol. 16, pp. 193-230, Sept. 2004.
17. Aquimisa, http://www.aquimisa.pt, Dec. 2007.
18. NONAKA, I; TAKEUCHI, H.: Criação de Conhecimento na Empresa. 12a. Edição. Rio de Janeiro: Campus, 1997.
19. Deepeye, http://www.deepeye.pt/, Dec. 2007.
20. Netsigma, http://www.netsigma.pt, Dec. 2007.
21. C4g, http://www.c4g.pt/noticias/default.asp?IDN=78&op=2, Dec. 2007.

Aspect-Oriented Web Development in PHP

Jorge Esparteiro Garcia

Faculdade de Engenharia da Universidade do Porto
jorge.garcia@fe.up.pt

Abstract. Aspect-Oriented Programming (AOP) provides another way of thinking about program structure that allows developers to separate and modularize concerns like crosscutting concerns. These concerns are maintained in aspects that allows to easily maintain both the core and crosscutting concerns. Much research on this area has been done focused on traditional software development. Although little has been done in the Web development context. In this paper is presented an overview of existing AOP PHP development tools identifying their strengths and weaknesses. Then we compare the existing AOP PHP development tools presented in this paper. We then discuss how these tools can be effectively used in the Web development.
Finally, is discussed how AOP can enhance the Web development and are presented some future work possibilities on this area.

Keywords: Aspect Oriented Programming, Web Development, AOP, PHP

1 Introduction

As web applications become more complex, it becomes harder to separate independent concerns. Aspect oriented programming (AOP) [9] paradigm offers various ways to separate concerns which can help us to reduce time and complexity of applications. AOP better separates concerns than previous methodologies (object oriented, procedure, etc.), thereby providing modularization of crosscutting concerns [10].

Despite AOP being a programming paradigm that can be used with the most common object oriented languages, much of the research has been done on developing standalone applications and little has been applied to Web development. Therefore, we believe Web development can be improved using aspect-oriented techniques.

AspectJ [8] is one of the most popular AOP proposal tools that offers the possibility to develop web applications using JSP (Java Server Pages). However, nowadays PHP is becoming the most widely used Web scripting. PHP has an edge over locked-in solutions such as JSP and ASP for most Web development work because it is a cross-platform technology.

PHP is, nowadays, one of the best and most popular script programming languages for innumerable web applications. Over 20 millions domains on web use PHP as the web programming language [12]. Specially suited for Web development, it´s recognized as one of the most used programming languages in the world.

Therefore in this work are presented the existing AOP PHP development tools and is made a comparison of these tools showing their strengths and weaknesses on the web development. It´s also discussed the impact of AOP Web development with a language with such a wide-spread use.

The rest of the this paper is as follows. Section 2 gives a brief overview of Aspect Oriented Programming. In Section 3 are presented the existing AOP PHP development tools. In Section 4 is made a comparison of these tools in the context of the web development. Section 5 concludes the paper and discusses the future work.

2 Aspect-Oriented Programming Web development

AOP is a new technology for separating crosscutting concerns into single units called aspects. An aspect is a modular unit of crosscutting implementation.

It encapsulates behaviors that affect multiple classes into reusable modules. With AOP, we start by implementing our project using our OO language (for example, Java), and then we deal separately with crosscutting concerns in our code by implementing aspects.

Finally, both the code and aspects are combined into a final executable form using an aspect weaver. As a result, a single aspect can contribute to the implementation of a number of methods, modules, or objects, increasing both reusability and maintainability of the code.

Figure 1 explains the weaving process. The original code doesn't need to know about any functionality the aspect has added; it only needs to be recompiled without the aspect to regain the original functionality.

2.1 Aspect-Oriented Programming Web Development Concerns

Although aspect oriented programming is a very young area of research, from the very beginning there have been concerns, such as synchronization or distribution, that have had the attention of researchers due to their clear crosscutting nature. However, there are other concerns that don´t crosscut so clearly, and haven´t had the focus of the aspect oriented community.

Following the proposal by Kilesev [8], Reina et al. [2] have addressed the following concerns on the development of web applications:

- **Security**. This concern is really authentication. Authentication is the process of determining whether someone is who is declared to be. This aspect tries to prevent that unauthenticated users have access to some web pages.

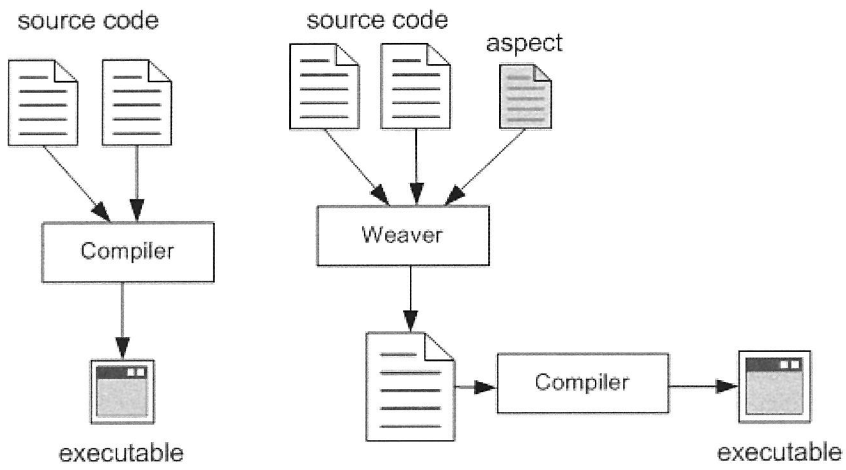


Fig. 1. Aspect Weaver

- **Design by Contract**. A contract is something that should be guaranteed before calling a method on a class, but, also, the class should guarantee certain properties after the call. This is a way to check if certain conditions are fulfilled before executing a method. Some programming languages have implemented this concern using the notion of assertion.

- **Exception Handling**. It is a simple way of applying an exception handling policy, in such a way that all exceptions should be handled by notifying the end user that something went wrong.

- **Logging**. This concern encapsulates the logger behavior. When certain points during the execution of a program are reached, a message is printed out. Tracing. It is a debugging tool very similar to a logger, but it only tracks one type of event, a method execution.

- **Profiling**. A debugging concern which measures the execution time consumed in some methods. This concern can be very helpful for detecting some bottlenecks. Pooling. Pooling is a strategy to obtain faster database connections. When a database and all its associated files are closed, the connection and server resources are released. If the same application needs the database services again, a new connection will have to be established and server resources will have to be asked for again, wasting resources, and, of course, slowing down the application. If we maintain a pool of connections and server resources, we will obtain faster database connections.

- **Caching**. Caching is the retention of data, usually in the application, to minimize network traffic flow and/or disk accesses. If database information is cached on the application server, the database server can be relieved of its repetitive work.

In the development of an web application there are some aspects that are crucial for the success of the final product. These key aspects are: pooling, caching and security.

On the one hand, pooling and caching are very important because they can have influence on response time, which is an important requirement, because a user can be bored waiting for a response, specially in a web application, where the response time can very exasperating to the user.

On the other hand, authentication is really important, because a web application can easily be altered by an intruder, and needs to be protected. But there are other key concepts, such as navigation [1], that should be addressed during the web development.

2.2 PHP Web development

There are several programming languages for the development of Web Interfaces. These programming languages are used primarily for developing server-side applications and dynamic content. Microsofts ASP.NET, PHP, Java, CGI, Perl are some of the technologies used on this area. PHP is currently one of the most popular server-side scripting systems on the Web.

One major part of PHP which has helped it become popular is that it is a very loose language; in particular, it is dynamically typed. The key technical contributor to PHP success is its simplicity, which translates into shorter development cycles, easier maintenance and lower training costs.

That is, the rules aren´t as strict with variables - they don´t have to be declared and they can hold any type of object. Further, unlike many other languages (like C++ and Java), arrays are able to hold objects of varying types, including other arrays.

PHP, like Perl or Javascript is a dynamically weakly typed interpreted language. However, like Java, classes (and in this case functions) are special entities within the language, they can't be directly referenced.

3 Aspect-Oriented PHP development tools

In this section are presented the existing tools for web development in PHP. There also presented some code examples of each implementation. We can consider two main methods of implementation of the extensions to support AOP in PHP. The Pre-Processing Implementation and the Runtime Weaving implementation.

In the Pre-Processing Implementation a preprocessor is used to perform source code transformations and carry the weaving process. Then the PHP source code produced can then be deployed in a standard PHP environment.

In the Runtime Weaving implementation, there is no previous source code generation, the weaving process is done dynamically and PHP code is executed normally as a "traditional" PHP web application.

3.1 Pre-Processing Weaving Implementations

There are already some different AOP implementations for PHP. The first implementations of all required pre-processing, this means it is necessary to run the PHP source through a processor first or patches against the engine. This pre-processing applies code transformations and integrates the aspects to generate final PHP code for the application.

This method has some disadvantages to classic PHP developers because pre-processing adds an extra step to the development. This means developers can´t write the code and then test it, the source code has to be processed after written and then only after that can be tested. Another issue of this method is that PHP becomes no longer an interpreted language. The code that is produced won´t run natively on any interpreter, the PHP developer has to program on a Java fashion way to be able to develop AOP web applications.

Though, this implementations are very useful because they can really implement AOP in PHP and bring to the language other capabilities that were not present until now.

PHPAspect The phpAspect [11] compiler weaves aspects implementing cross-cutting concerns as shown in the weaving chain in Figure 2. Inspired in AspectJ, the phpAspect is one of the most used solutions to develop PHP web applications using the proposals of AOP. As previous referred, the weaving process is static and based on Lex and Yacc analysis to generate XML parse trees. XSLT is used to perform the source code transformation on those trees. PHPAspect contains the usual jointpoints on AOP like method execution/call, attribute writing/reading, object construction/destruction, exception throwing, etc.

One important plugin on this tool is the PHPAspect Builder that provides check of aspect syntax and also offers the possibility to weave each php file contained in the project. Figure 3 shows a screenshot of this plugin.

Aspect-Oriented PHP Aspect-Oriented PHP(aoPHP) [3] uses a preprocessor for the PHP programming language written in Java. This preprocessor is responsible for the weaving process of the aspects and the base-code. aoPHP contains support to the methods: execution/call, field read and write, among others.

The aspects on this implementation are stored in files with the extension .aophp. aoPHP plans to evolve in direction of an Aspect-Oriented Language with a rich joint point model such as AspectJ.

AOP Library for PHP This library, developed by Dmitry Sheiko [13], can implement Aspect Oriented Programming (AOP) by executing the code of classes that enable orthogonal aspects at run-time.

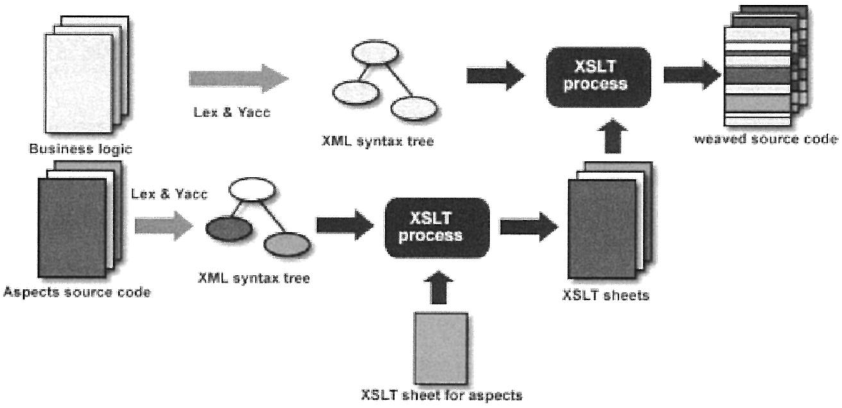
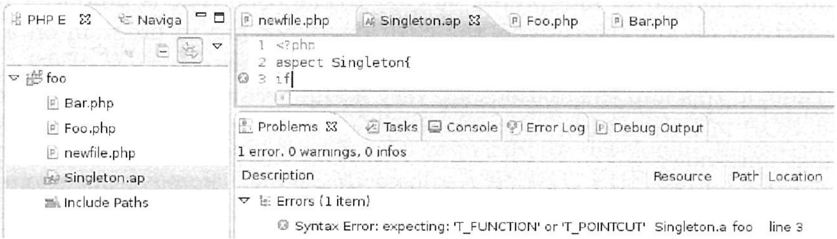


Fig. 2. PHPAspect´s weaving chain



```
<?php
aspect Singleton{
if
```

1 error, 0 warnings, 0 infos

Description	Resource	Path	Location
Errors (1 item)			
Syntax Error: expecting: 'T_FUNCTION' or 'T_POINTCUT'	Singleton.a	foo	line 3

Fig. 3. PHPAspect Builder Screenshot

The intention is to provide and implement orthogonal aspects in separate classes that may be interesting to add, without affecting the main business logic to the application, like logging, caching, transaction control, etc.

The package provides base classes for implementing defining point cuts where the code of advice class is called to implement actions of the orthogonal aspects that an application may need to enable.

3.2 Runtime Weaving

The extensions that implement this method use aspect weavers that work in application runtime, taking advantage of the interpreted nature of the PHP language, and weaving the aspects on demand.

aspectPHP aspectPHP [4] is another implementation that works in application runtime, taking advantage of the interpreted nature of the PHP language.

A first version of this tool was adapted from aoPHP implementation described on Section 3.1. On this version they assume all the aspects are already located in the same directory, as separate files "*.aspect". The aspects were then loaded in sequence by the weaver, to weave them with the original code.

Another version based on the Zend compiler was released. This version was released due to in the previous implementation is unlikely that the PHP call-site will be captured, especially when the functions are not inside a script file, but included from other program files. Zend compiler was then adopted changing the compiler to support aspects.

GAP: Generic Aspects for PHP GAP [6] is the first implementation in AOP PHP that supports dynamic weaving, genericity and an extensible pointcut language.

This is a project under development by Sebastian Bergmann, the author of PHPUnit [5] (software testing framework for PHP based on JUnit of Java), for the creation of an AOP extension that takes full advantage of the new functionalities made available by PHP 5, such as the Reflection API or the overloading methods call(), get() and set(). This project, which is not available for public use yet, uses the PHP Runkit [7] extension, a new and powerful reimplementation of the Classkit extension mentioned previously. Figure 5 shows the aspect declaration with the AOP Library for PHP in GAP.

Figure 4 illustrates the GAP weaving chain. This chain uses a streams filter written in PHP, the GAP Weaver hooks into the loading of the source code of classes (represented in green color) and aspects (represented in blue color). The first weaving stage performs source code transformations then, it passes the generated source code to the compiler, integrated in the PHP Interpreter. The second weaving stage operates on the bytecode generated by the compiler. It uses the Runkit extension to complete the insertion of generic hooks into the classes.

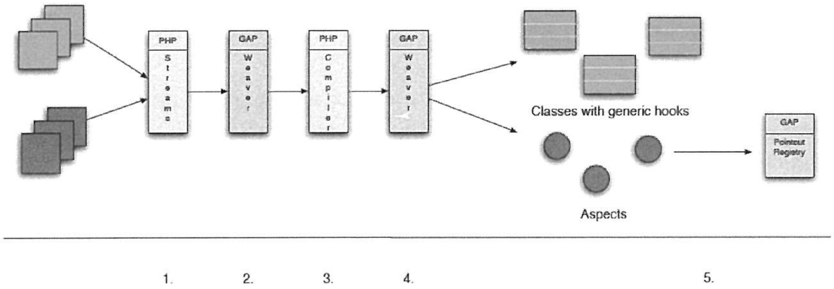


Fig. 4. The weaving chain of GAP

In this implementation, the definition of aspects is obtained through the creation of classes whose methods correspond to the kind of advice to apply (before, after or around). The creation of pointcuts and association of advice to the joint points is carried through with the aid of annotations that precede the class definition. An example of GAP aspect that logs all method calls can be seen in Figure 6.

```
<?php

require_once 'Class.php'
require_once 'aop.lib.php'
$aspect = new Aspect;
$pointCut = $aspect->pointcut('all AClass::aMethod';
$pointCut->_before('... before advice code ...';
$pointCut->_after('... after advice code ...';
$pointCut->destroy();
$object = new AClass($aspect);
?>
<?php class AClass {
private $aspect;
public function __construct($aspect) {
$this->aspect = $aspect;
}
public function aMethod() {
Advice::_before($this->aspect);
// ... base code ...
Advice::_after($this->aspect);
}
} }?>
```

Fig. 5. Aspect declaration with the AOP Library for PHP in GAP.

An example of GAP aspect that logs all method calls can be seen in Figure 6.

4 Comparison of the Tools

All tools presented in this paper are in a very early stage of development. Therefore we cannot yet make some deep considerations about this AOP PHP development tools.

Although, we can make some comparison based on the weaving process method used by each tool and the possibilities of improvement of each tools features and capabilities.

4.1 Weaving Process Method

In the Pre-Processing Weaving Implementation method there are some disadvantages when compared to the Runtime Weaving process method, due to the need to use a preprocessor to perform source code transformations and carry the weaving process.

There are two main issues. Preprocessing adds an extra step to the development: it's no longer code then test as in traditional php development; but has become code, preprocess then test. The benefit of php being an interpreted

```
<?php
/* @pointcut allInvocations : method(* *->*(..));
* @after allInvocations : Logging->log();
*/
class Logging {
public function log($joinPoint) {
printf(
"%s->%s() called %s->%s()\n",
$joinPoint->getSource()
->getDeclaringClass()
->getName(),
$joinPoint->getSource()
->getName(),
$joinPoint->getTarget()
->getDeclaringClass()
->getName(),
$joinPoint->getTarget()
->getName()
);
}
} ?>
```

Fig. 6. GAP aspect that logs all method calls

language is lost. The second issue is that you have stopped writing PHP, but have started writing a dialect of PHP; your new dialect won't run natively on any interpreter.

Some preprocessors like Aspect-Oriented PHP move the step of preprocessing out of the programmers hands to the hands of apache, so that apache will preprocess the PHP before handing it off to the Zend engine, but it´s still a Pre-processing that is executed in the PHP Hypertext PreProcessor language. This is highly redundant and very slow.

Developing Web applications using a tool with the Runtime Weaving process method can take some advantages. The main advantage is, of course, not having to deal with the step of preprocessing that allows to apply the aspects in runtime. It also can be very annoying to a php developer to deal with the world of class libraries, frameworks or applications that are need to the Pre-Processing Weaving Method.

5 Conclusions

This paper presented various solutions and approaches to support the AOP paradigm in the PHP Web development. Depending the method used for the weaving process the implementations were classified as Pre-Processing implementations or Runtime Weaving implementations.

Despite AOP being a programming paradigm that can be used with the most common object oriented languages, much of the research has been done on developing standalone applications and little has been done applied to Web development. Therefore, we believe Web development can be improved using aspect-oriented techniques and this paper shows some good web development tools to help the improvement of Web development.

On Section 4 we shown that the Runtime Weaving method can be very interesting applied on the Web Development, in particularly the fact of offering the developer the possibility to apply aspects in runtime without the need of the step of preprocessing.

Despite the possibilities of applying PHP to AOP Web development, the solutions that were presented and studied in this paper, are all on a very early stage of development. This means, that doesn´t exists a PHP tool that can explore and implement the AOP paradigm like an OO language as Java. Although, these tools seem to very promising specially GAP, that can change the PHP web development in the next few years.

References

1. J. Torres A. M. Reina. Separating the navigational aspect. In *Proceedings of the Workshop of Aspect-Oriented Programming for Distributed Computing Systems*, Viena, Austria, 2002.
2. M.Bonilla A. M. Reina, J. Torres. Aspect-oriented web development vs. non aspect-oriented web development. In *Workshop AAOS2003: Analysis of Aspect Oriented Software*, Darmstadt, Alemania, 2003.
3. aoPHP. Website visited on January 3th 2008 at http://www.aophp.net/.
4. aspectPHP. Website visited on January 5th 2008 at http://www.cs.toronto.edu/ yijun/aspectPHP/.
5. S. Bergmann. PHPUnit website Visited on January 6th 2008 at http://www.phpunit.de.
6. Sebastian Bergmann and Günter Kniesel. Generic aspects for php. In *Proceedings of EWAS 2006*, Netherlands, 2006.
7. S. Golemon. Runkit extension for PHP website, Visited on January 6th 2008 at http://pecl.php.net/package/runkit.
8. Gregor Kiczales, Erik Hilsdale, Jim Hugunin, Mik Kersten, Jeffrey Palm, and William G. Griswold. An overview of AspectJ. *Lecture Notes in Computer Science*, 2072:327–355, 2001.
9. Gregor Kiczales, John Lamping, Anurag Menhdhekar, Chris Maeda, Cristina Lopes, Jean-Marc Loingtier, and John Irwin. Aspect-oriented programming. In Mehmet Akşit and Satoshi Matsuoka, editors, *Proceedings European Conference on Object-Oriented Programming*, volume 1241, pages 220–242. Springer-Verlag, Berlin, Heidelberg, and New York, 1997.

10. C. Lopes and W. Hursch. Separation of concerns, 1995.
11. phpAspect. Website visited on January 5th 2008 at http://phpaspect.org/.
12. PHP.net. PHP.net website visited on January 24th 2008 at http://www.php.net/usage.php.
13. D. Sheiko. *Aspect Oriented Software Development and PHP*, volume 5. In php — architect, 2005. issue 4, pages 17-25.

Comparing Three Aspect Mining Techniques

Fernando Sérgio Barbosa

Faculdade de Engenharia da Universidade do Porto
fsergio@est.ipcb.pt

Abstract. Even in well designed software systems there are some concerns that are spread over many units. Aspect Oriented Programming is a new programming paradigm that enables modularisation of these crosscutting concerns as aspects. Some crosscutting concerns are obvious to spot, but many others are not. In large systems or frameworks, identifying them is a hard task that needs the help of tools. The identification of such crosscutting concerns is called aspect mining. In this paper we compare three aspect mining techniques: Fan In analysis, Dynamic analysis and Clone detection. We compared them using a common target: the JHotDraw framework. We then discuss each of the techniques strengths and weaknesses by comparing the results. Finally we analyse the possibilities of combining the techniques to achieve better results.

Keywords: aspect mining, aspect oriented programming, fan in analysis, dynamic analysis, clone detection

1. Introduction

The tyranny of the dominant decomposition [1] states that there are always some concerns, called crosscutting concerns, that, even in well decomposed systems, will not neatly fall in any of the units thus being spread over many of them.

Aspect Oriented Programming (AOP) [2] is an emerging programming paradigm with primitives that allow modularisation of the so-called cross-cutting concerns, removing much of the code scattering and tangling, placing them in aspects. Because of this programmers seek to use AOP in their programs. But, as with migrating to a new technology, they have to deal with a large base of installed artefacts based on the old technology and must keep on using it or refactor [3] it to aspects.

While some crosscutting concerns are plain obvious, like the traditional ones referred in AOP literature as logging, persistence, security, memory management, etc, others are not so obvious and require a much more detailed look at the source code. In large systems or frameworks this is a hard task that not all programmers, if any, wishes to do. This calls for the help of tools that can identify possible aspects from the source code. This aspect finding activity is called aspect mining [4]. Current tools allow for semi-automatic identification for aspects only, but in the future this can be automated. If automatic aspect mining is used together with refactoring techniques then the conversion of OO code into AO code also becomes automatic [5].

In this paper we compare three aspect mining techniques: Fan In analysis [6], Dynamic analysis [7] and Clone Detection [8]. We compared them using a common target: the JHotDraw framework. We do not aim to decide which of the techniques is best but to identify their strengths and weaknesses. A best/worst categorization would need well established criteria for good aspect modularisation and that is not yet available. Because of this we limit ourselves to a qualitative comparison.

Another objective of this paper is to see if the combination of the techniques can improve the results or if they overlap. Either way it is good to know: if they overlap we can discard one of them, if they complement each other then we can combine them to achieve better results. Our conclusions support the fact that the three techniques are combinable as they have different strengths and weakness.

This paper is structured as follows: In Section 2 we present the three aspect mining techniques and briefly present the tools used for the comparison. The experiment is presented in Section 3 as well as the results of the experiment. In Section 4 the comparison of the three techniques is made based on the results from Section 3. Section 5 presents related work and Section 6 concludes the paper.

2. The Three Aspect Mining Techniques

In this paper we will compare the following 3 aspect mining techniques: fan-in analysis [6], dynamic analysis [7], and clone detection [8]. These are not the only proposed aspect mining techniques but they are supported by publicly available tools. The respective tools are briefly presented within each technique.

2.1. Fan-In Analysis

The fan-in of a method *M* is defined as the number of calls to method *M* made from other methods [6]. Because of polymorphism, one method call can affect the fan-in of several other methods. A call to method *M* contributes to the fan-in of all methods refined by *M* as well as to all methods that are refining *M [9]*. The more places the method is called from the more likely it is that the method implements a crosscutting concern so the amount of calls (fan-in) is a good measure for the importance and scattering of the discovered concern.

The analysis follows three consecutive steps: (1) Automatic computation of the fan-in metric for all the methods in the targeted source code. The result is stored as a set of "method-callers" structures that can be sorted by fan-in value. (2) Filtering of the results of the first step, by restricting the set of methods to those having a fan-in above a certain threshold; filtering getters and setters from this restricted set. Get/Setters on static fields are not eliminated because these can be used in the Singleton design pattern; filtering utility methods, like toString( ), collections manipulation methods, etc. (3) Analysis of the remaining set of methods. The elements considered at this step are the callers and the call sites, the method's name and implementation, and the comments in the source code.

A tool that supports Fan In is FINT[1]. FINT is implemented as an Eclipse[2] plug-in [10]. The current implementation of FINT includes three source code analysis

[1] http://swerl.tudelft.nl/bin/view/AMR/FINT

techniques to identify crosscutting concerns: Fan-in analysis, Grouped calls analysis and Redirections finder. Since this paper only concentrates in the first one the others were disabled.

The results of the analysis are displayed in the Fan-in analysis view as a tree structure of callee-callers elements, sorted by name or fan-in value. From this view the user can inspect the source code of each of the displayed elements and apply the already mentioned filters to restrict the elements to the most relevant candidates.

2.2. Dynamic Analysis

The technique of Formal Concept Analysis (FCA) is fairly simple [11]. Starting from a (potentially large) set of elements and properties of those elements, FCA determines maximal groups of elements and properties, called concepts.

FCA is used for aspect mining according to the following procedure: Execution traces are obtained by running an instrumented version of the program under analysis for a set of use cases. The execution traces associated with the use cases are the objects, while the executed class methods are the attributes. In the resulting concept lattice, the concepts specific of each use case are located, when existing. The use case specific concepts are those labelled by at least one trace for some use case (i.e. the concept contains at least one specific property) while the concepts with zero or more properties as labels are regarded as generic concepts. When the methods that label one concept crosscut the principal decomposition, a candidate aspect is determined. More specifically, a concept is a candidate aspect if:

- scattering: more than one class contributes to the functionality associated with the given concept (i.e., the methods labelling the concept belong to more than one class);
- tangling: the class itself addresses more than one concern (i.e., appears in more than one use case specific concept).

The first condition alone is typically not sufficient to identify crosscutting concerns, since it is possible that a given functionality is allocated to several modularised units without being tangled with other functionalities. In fact, it might be decomposed into sub-functionalities, each assigned to a distinct module. It is only when the modules specifically involved in a functionality contribute to other functionalities as well that crosscutting is detected, hinting for a candidate aspect.

Dynamo[3] is a tool for the identification of aspects in existing Java classes by means of a dynamic code analysis [7]. Execution traces are generated for the use cases that exercise the main functionalities of a given application. The relationship between execution traces and executed computational units is subjected to concept analysis. In the resulting lattice, potential aspects are detected by determining the use case specific concepts and examining their specific computational units.

[2] http://www.eclipse.org/

[3] http://star.itc.it/dynamo/

2.3. Clone detection techniques

Clone detection techniques aim at finding duplicated code, which may have been adapted slightly from the original. Several clone detection techniques have been described and implemented [8]: Text-based techniques attempt to detect identical or similar lines of code. Token-based techniques use tokens as a basis for clone detection. AST-based techniques build an Abstract Syntax Tree which is searched for similar subtrees. PDG-based approaches construct a Program Dependence Graphs (PDGs) that contains information such as control and data flow of the program. Metrics-based techniques calculate, for each fragment of a program, a number of metrics which are used to find similar fragments. Information Retrieval-based methods exploits semantic similarities present in the source code.

Clone detection techniques are promising in aspect mining due to two likely causes: First, by definition, scattered code is not well modularised so developers are unable to reuse concern implementations through the language module mechanism. Therefore, they are forced to write the same code over and over again, typically resulting in a practice of copying existing code and adapting it slightly to their needs. Second, they may use particular coding conventions and idioms to implement superimposed functionality, i.e., functionality that should be implemented in the same way everywhere in the application. This is even more so with the adoption of patterns [12], where similar code is found in various implementations of a pattern.

CCFinder[4] is a multilanguage clone detector [13]. CCFinder makes a token sequence from the input code through a lexical analyser and applies a, language specific, rule-based transformation to the sequence. The purpose is to transform code portions in a regular form to detect clone code portions that have different syntax but have similar meaning. Representing a source code as a token sequence enables the detection of clones with different line structures, which cannot be detected by line-by-line algorithm. After the detection the user has several browsing capabilities.

3. Running the Experience

As a common ground for comparison of the three aspect mining techniques the JHotDraw[5], v5.4b, framework was used. JHotDraw is a Java GUI framework for technical and structured graphics that has been developed as a "design exercise". Its design relies heavily on design patterns [12]. JHotDraw's original authors have been Erich Gamma and Thomas Eggenschwiler. JHotDraw is considered well designed and so has been used in both aspect mining [9] and aspect refactorings efforts [14].

3.1. Fan In Experience

The setting up of the experience with FINT is easy. All we had to do was create a project in Eclipse and run the plug-in. The results came in about 1,5 seconds. The

[4] http://www.ccfinder.net/index.html

[5] http://www.jhotdraw.org/

results comprised 479 filtered methods (with all filters on). After applying a threshold of 10 to the fan in metric the final result gave 120 methods that had to be analysed.

For each method two steps were taken: first we inspected the method's body to get a perception of its functionality, second we analysed the call sites in order to get the context of the methods usage. After this manual examination 45 methods were considered seeds for possible aspects. This doesn't mean that 45 aspects were found as many methods contributed to the same functionality. For example an instantiation of the Composite pattern [12] had 10 methods associated with it. But some concerns were associated with a single method. An example is util.Undoable.isRedoable() that indicates a consistent behaviour because every redo action must perform a isRedoable() test before execution.

A summary of the crosscutting concerns found is presented in Table 1. The grouping of the consistent behaviour and contract enforcement is done for it is sometimes hard to distinguish between the two.

Table 1. Summary of the results from the Fan In experiment

Description	methods	Concern type
When a tool has finished execution method toolDone() must be called	3	Consistent Behaviour
Border decorator	1	Decorator
Composite pattern in Figure hierarchy	10	Composite
FigureChangedListener	7	Observer
After execution of commands/tools the view must be checked for damages	4	Consistent Behaviour
After execution of commands (specially redos) the selection must be cleared	2	Consistent Behaviour
Every tool class must call the superclass constructor	1	Consistent Behaviour
Tool activation and deactivation	2	Consistent Behaviour
Every tool calls super.mouseDown()within their own mouseDown()	1	Consistent Behaviour
many figures use readInt, writeInt and such methods to write to a stream	6	Persistence
Before doing a redo the method isRedoable() must be called	1	Consistent Behaviour
Every undo operation must call its superclass undo()	1	Consistent Behaviour
Every UndoableAdapter calls its superclass constructor	1	Consistent Behaviour
UndoableComand uses the decorator pattern	1	Decorator

From the experience we could see that Fan In analysis can detect crosscutting concerns that are implemented in one of three ways: (1) The crosscutting concern is implemented by a single method that is called from several call sites, as is the case with the consistent behaviour where the method has to be called before some other execution. (2) The crosscutting concern is implemented by a single method but that method is used from several call sites for the same basic functionality. An example is persistence where methods to read/write several types of data are used by most classes that implement a figure. (3) When several methods contribute to a crosscutting concern it is likely that the crosscutting concern is derived from a pattern superimposed role. Examples of these patterns are Composite (10 methods) or Observer (7 methods).

Fan In analysis helped to find some crosscutting concerns by "mere chance". An example is the Decorator pattern [12] that was found because the methods from the decorator class had a high Fan In. If we limited the examination to the call sites such concern would be undiscovered because they were addressing the class main concern. The method code, however, clearly indicated that the class was a decorator. If it were not for the fact that the methods had high Fan In AND the names of identifiers and methods clearly indicated the purpose this pattern would be unnoticed.

3.2. Dynamic Analysis Experience

This was the most difficult experience of all. Before the analysis could be done the code had to be "instrumented". This meant that every ".java" file class had to be copied to a ".oj" file. Afterwards every file had to be edited and, for every class, (inners classes too) some code had to be inserted. Every file should also add an import clause for the dynamo package. This work could be done automatically by a tool but there wasn't, unfortunately, none available. We had to write a piece of code to do the work (the idea of doing it manually was refused). In the end those ".oj" files had to be compiled with a special compiler (OpenJava[6]) and the resulting ".java" files had also to be compiled. The OpenJava compilation is not fast, taking in JHotDraw more than 5 hours in a 2.4 Ghz, Pentium IV with 512MB. This "instrumentation" has only to be performed once and the tests are done with the instrumented version.

After the successful instrumentation of the framework some use cases were run. The time spent (more than 1 hour) running the use cases was considerable. The concept lattice generated had more than 1500 nodes. This amount of information cannot be dealt with manually so the aspect mining tool of Dynamo was used.

Table 2. Summary of the results from the Dynamic Analisys experiment

Description of concern	Concepts	Methods
Undo	2	36
Bring to front	1	3
Send to back	1	3
Connect text	1	18
Persistence	1	30
Handle manipulation	4	60
Figure Observer	4	11
Command executability	1	25
Connecting figures	1	55
Manage figures outside drawing	1	2
Get Attribute	1	2
Set Attribute	1	2
Managing view rectangle	1	2
Visitor	1	1

A summary of the concerns found is presented in Table 2. As can be seen from Table 2 several concepts contribute to the same concern. This is the case for the handle manipulation concern in which four concepts contribute to that concern.

3.3. Clone Detection Experience

CCFinder is an easy to use tool. It is enough to specify which files to analyse. The analysis was done in 15.3 seconds. To that time it must be added the time to calculate some metrics that were needed latter, in about 20 seconds.

The first result included 433 clone sets. We then filtered the sets with a population less than 4, reducing the clone sets to 79. These 79 clone sets were manually inspected. We selected the clone set to inspect and the tool showed the various places where the clone was reproduced. It must be mentioned that some clones do overlap,

[6] http://www.csg.is.titech.ac.jp/openjava/

i.e. a larger code clone may include a smaller code clone. Some clones are also similar to others, and because the tool doesn't take in account identifiers names some clones have unrelated code (false clones). These clones were removed.

After inspection 46 clones were discarded leaving a final 33 clone sets. Much like Fan In analysis a clone set does not have a one to one relation with a crosscutting concern. This is the case with the Undo concern that has 14 clone sets associated with its implementation. Some clones do represent a single crosscutting concerns as for example the clone that shows that tools must update the view after a mouseDown( ).

A summary of the concerns found is presented in Table 3.

Table 3. Summary of the results from the Clone Detection experiment

Description	Clone sets	Concern type
Tools must update the view after a mouseDown( )	1	Consistent behaviour
1 subject roles in Observer pattern: Command Listener and Tool Listener	4	Observer (2x)
Undo	14	Undo
Reading and writing data	1	Persistence
1 observer role and 3 subject roles in FigureListener	2	Observer
2 subject roles of ViewListener	1	Observer
1 subject roles of Figure Listener and 2 subject roles of DesktopListener	1	Observer (2x)
Handle manipulation	1	Handles
Event dispatcher + undo	1	undo + event dispatcher
Connecting figures	1	Connecting figures
5 subjects roles of FigureListener	1	Observer
Before executing commands isExecutable() must be called	1	Consistent behaviour
Every handle checks owner's display box	1	Consistent behaviour
3 instances of decorator	1	Decorator
Many commands have to perform isExecutableWithView	1	Consistent behaviour
1 ViewChangeListener subject role + 2 DesktopListener subject roles	1	Observer

From the experience we could see that Clone Detection analysis can detect crosscutting concerns that are implemented in one of three ways: (1) the code that deals with the crosscutting concern is implemented in every class that addresses the crosscutting concern. This is the case with the consistent behaviour where every class must perform a call to a specific method. (2) The code that deals with a particular crosscutting concern is implemented by several classes with minor variations, as is the case with the undo concern. (3) The code is part of a superimposed role from the participation in a pattern. This is the case with the various instances of the Observer pattern where every class must implement addObserver() like methods which are very similar. It must be said that the detection of some patterns depends on the number of the classes that participate. For example the detection of the Observer pattern was possible because there were several subjects. If it was only one subject none would be detected because only one instance of addObserver( ) method existed. The way each observer reacts upon an event depends on the observer itself and so the code is unique, which explains why only one observer was found.

It must be stressed that sometimes the clone itself didn't provide enough information to achieve a conclusion, but examining the surroundings of the clones for a wider vision was enough to detect some crosscutting. The most expressing one is undo: while several clone sets contributed to this concern it was when the surroundings where examined that we found that every class implementing undo had a UndoActivity as inner class. This clearly indicated Undo as a crosscutting concern.

4. Comparing the techniques

For the comparison of the techniques we selected some concerns that were detected by all the techniques and some that were discovered by two or one of the techniques. For each concern we discuss why the various techniques failed/succeed to identify it.

4.1. Concerns Used in Comparison

Observer

Every technique succeed to identify this concern, even if they didn't find all instances of the Observer pattern. The Fan In and Dynamic analysis identified only one while the clone detection technique discovered four. Note however that Clone Detection only identifies subjects (with exception of one observer), this is because the Observer pattern uses very similar code between its instantiations, namely to manage observers. The Fan In succeeds in the FigureChangedListener because there are many observers thus the methods to register in the subject surpasses the Fan In threshold. But because there are few DesktopListener and ViewChangeListeners those methods didn't pass the Fan In threshold and it failed to identify them.

Composite

Only the Fan In analysis succeeded to identify this concern. Clone detection failed because there are few instances of the pattern so clone population didn't arise above the threshold. Dynamic analysis failed to identify it due to the fact that every trace uses the composite pattern because every drawing is a composite figure and so this is exercised in every use case.

Undo

Both Clone Detection and Dynamic analysis detected this concern, but Fan In failed. Even though Fan In detected that all redo must do a isExecutable() this falls in the category of consistent behaviour. It can be argued that detecting this behaviour one could suspect that an undo concern is present so Fan In detects it too, but that is the annalist deduction/perception/experience that enables it and not a direct consequence of the Fan In. Clone Detection succeed because every undo activity has similar code (most notorious are the undo and redo methods of an inner class UndoActivity). Dynamic analysis detected it because undo is considered a use case and so is exercised in the application.

Handle Manipulation

Both Clone Detection and Dynamic analysis detected this concern, but Fan In failed. Handle manipulation is used in the use cases that manipulates figures so the Dynamic analysis detects it fairly well. Clone Detection succeeded because much of the handles code is similar. Fan In failed because there are many methods related to the handle manipulation but each is called from very few points.

Consistent behaviour

Every technique identified this type of concern but they identified it in very different ways and not all the consistent behaviour was discovered by all the techniques. Namely Dynamic analysis only captured the command executability concern. The Fan In was the one that discovered more cases (9). Clone detection also discovered a few (4). This is clearly the goal of the Fan In method were a specific concern is done by a method that is called from several places in the code. Clone detection also checks this because some of the code is very similar (often involving an if statement with a method invocation followed by a return statement). But since this code is usually very short Clone Detection can fail to capture it because the clone has to be bigger than a minimum length in size. To capture this code the minimum clone length must be small but this in turn increases the size of the clone sets found and the noise generated is much greater (for example in one of the experiments every for was considered a clone). Dynamic analysis failed to detect most of the concerns because they are exercised in almost every use case.

Bring to Front/Send to back

Only Dynamic analysis detected this concern. Fan In failed because the methods dealing with it are not called from many places as it is a rather specific concern. Clone detection failed because the concerns code is specific to two classes and not repeated.

Table 4. Summary of the comparison between the tree techniques with the selected concerns.

Concern	Fan In Analysis	Dynamic Analysis	Clone Detection
Observer	+	+	++
Composite	+		
Undo		+	++
Handle Manipulation		++	+
Consistent Behaviour	++ (9)	+ (1)	+ (4)
Bring to Front/Send to back		+	

Table 4 summarizes the previous discussion. In Table 4, concerns that are discovered by a technique are marked +, if they are not discovered they are not marked. If a technique was more efficient than others in discovering a concern it is marked ++. This way we can see the strengths and weaknesses of the techniques.

4.2. Limitations of the techniques

Comparing the techniques in a common ground enables us to identify their strengths and limitations. While the strengths are outlined in the techniques presentation area their limitations are uncovered in the experiment. Next we present the limitations found for each technique.

Fan IN

Mainly addresses crosscutting concerns that are largely scattered and have a significant impact on the modularity of the system. More: it depends on the correct modularisation of those concerns in methods. If the code is not placed in methods but "copy/pasted" (unfortunelly this is not uncommon) the technique fails completely. This means that concerns with a small code footprint and thus with low fan-in values, will be omitted. For example, as debated before, the identification of Observer design pattern instances is dependent on the number of classes implementing the observer role. The number of observer classes will determine the number of calls to the registration method in the subject role. A collateral effect is the anticipated unsuitability of the technique for analysing small case studies.

Dynamic Analysis
Among the known limitations of this technique, the two most important ones are that it is partial (i.e., not all methods involved in an aspect are retrieved) and it can determine only aspects that can be discriminated by different execution scenarios (e.g., aspects that are exercised in every program execution cannot be detected). Additionally, it does not deal with code that cannot be executed (e.g., code that is part of a larger framework, but is not used in a specific application).

Clone Detection
This technique addresses code that is similar in crosscutting concerns, but if a concern is handled differently by those who implement it the technique fails. Even one of the best results obtained by the technique depends on the amount of code, as is the case with the Observer patterns of which it encountered several instantiations (mainly subjects). If there was only one instance of this pattern then Clone Detection would fail to discover it. The undo concern is another example: if few classes implemented the undo or didn't follow a common usage it would be undetected.

4.3. Combining the techniques

Overall, fan-in analysis and dynamic analysis show largely complementary result sets. This is an expected result [9], since the first technique focuses on identifying those methods that are called at multiple (scattered) places. However, when a method is called multiple times in a system, it is likely to occur in most the execution traces, so that no specific use case can be defined to isolate the associated functionality.

Clone detection is also a very good technique capable of complementing the other two, specially in the case of superimposed roles that have similar implementation. As can be seen from Table 4, the best results may be obtained from the combination of the three techniques and not by a single technique alone. The example of the Observer pattern also suggests that Clone Detection technique could be used together with Fan In to achieve better results. The Fan In identified the observers roles while Clone Detection identified the subject role. This is also the case with the badly written modularity (using copy/paste instead of method calling) where Fan In would fail but Clone Detection would succeed. Using Dynamic analysis would bring to the scene the concerns that are not detected by the other techniques.

5. Related work

Ceccato et al compare three aspect mining techniques: Fan In analysis, Identifier analysis and Dynamic analysis and provides some thoughts in combining them [9]. We added a, minor, extension to their work using a different technique and our results are somewhat different. This is explained by the fact that stating that a given concern is a crosscutting concern and not a class specific concern can vary. Such an example is the move figure concern identified as a crosscutting concern in [9] and as a class concern by us. Others include the detection of the Command pattern that we dismissed based on the assumption that the Fan In technique does not directly identify this concern. The identified method didn't address a crosscutting concern and the call sites reflected that, but its implementation suggested that the Command pattern was being used, mostly by the identifiers names. To be fair in the comparison we marked that concern as not identified. Identical decisions were made in the Dynamic analysis. The same authors give a full description on how to combine the techniques in [15].

Bruntnik et al compare three clone detection techniques in aspect mining [8]. They evaluated the suitability of the three techniques for identifying crosscutting concern code. The evaluation considers token, AST, and PDG-based clone detection techniques and provides a quantitative comparison of their suitability. In their case study they manually identify five crosscutting concerns and evaluate to what extent the crosscutting concern code is matched by the three clone detection techniques.

Breu suggests to enhance DynAMiT, a aspect mining tool based on dynamic analysis, with static information and generating the traces using call pointcuts [16].

Cojocar and Serban propose several criteria to be used in comparing aspect mining techniques [17]. But most criteria take into account the detected aspects versus the number of existing aspects. This means that the comparison must be made using a well known and previously aspectized code. Such a code base is not yet available but some steps are being taken in order to completely refactor the JHotDraw framework into an aspect oriented framework named AJHotDraw [14].

6. Conclusion

A major problem in re-engineering legacy code based on aspect-oriented principles is to find and to isolate crosscutting concerns, a problem known as aspect mining. The detected concerns can be re-implemented as separate aspects, thereby improving maintainability and extensibility as well as reducing complexity.

In this paper we presented three aspect mining techniques and compared them using a common target. Each of the techniques has its strengths and weaknesses. Fan In deals better with concerns that are implemented in methods that are called from the various places dealing with the concern. It cannot handle concerns that are not much scattered or that are not implemented with methods calls but with in site code. Dynamic analysis deals better with concerns that are associated with some of the application use cases but fails to discover the concerns that are common to all use cases and also fails to detect unused concerns making it rather unsuitable to detect concerns in a framework for example. Clone Detection finds the crosscutting concerns that are implemented via replicated code but fails to address those concerns that have different code for each place that deals with it, as was the case with the observer role in the Observer pattern.

The combination of the techniques would benefit the results because each technique has its strengths were the others have their weakness.

References

1. Tarr, P., Ossher, H., Harrison, W., Sutton, J. S. M.: N degrees of separation: Multi-dimensional separation of concerns. Proceedings of the 21st international conference on Software engineering, (1999) 107-119
2. Kiczales, G., Lamping, J., Mendhekar, A., Maeda, C., Lopes, C., Loingtier, J., Irwin, J. Aspect-oriented programming. In Proceedings of ECOOP 1997, Finland, (1997) 220–242.
3. Monteiro, M.P., Fernandes, J. M.: Towards a Catalogue of Aspect-Oriented Refactorings. In proceedings of AOSD 2005, Chicago, USA (2005) 111-122
4. Kellens, A., Mens, K., Tonella, P.: A Survey of Automated Code-level Aspect Mining Techniques. Transactions on Aspect-Oriented Software Development, Special Issue on Software Evolution, 2007, to appear.
5. Deursen, A., Marin, M., Moonen, L.: Aspect mining and refactoring. In Proc. of the 1st International Workshop on REFactoring: Achievements, Challenges, Effects. (2003)
6. Marin, M., Deursen, A., Moonen, L.: Identifying aspects using fan-in analysis. In Proc. of the 11th IEEE Working Conference on Reverse Engineering (WCRE 2004). (2004)
7. Tonella, P. , Ceccato, M.: Aspect mining through the formal concept analysis of execution traces. In Proc. of the 11th IEEE Working Conference on Reverse Engineering (WCRE 2004), Delft, The Netherlands (2004).
8. Bruntink, M. Deursen, A. Engelen, R. Tourwé, T., On the use of clone detection for identifying crosscutting concern code, IEEE Transactions on Software Engineering, Vol. 31, No. 10, (2005)
9. Ceccato, M., Marin, M., Mens, K., Moonen, L., Tonella, P., Tourwe, T.: A qualitative comparison of three aspect mining techniques. In Proceedings on the 13th International Workshop on Program Comprehension, (2005) 13–22
10. Marin, M., Moonen, L. Deursen, A.V.: FINT: Tool Support for Aspect Mining, 13th Working Conference on Reverse Engineering (WCRE 2006), (2006) 299-300,
11. Ganter, B., Wille, R.: Formal Concept Analysis: Mathematical Foundations. Springer-Verlag, (1999)
12. Gamma, E., Helm, R., Johnson, R, Vlissides, J.: Design Patterns, Elements of Reusable Object-Oriented Software. Addison-Wesley, (1995).
13. Kamiya, T., Kusumoto, S., Inoue K.: CCFinder: A Multi-Linguistic Token-based Code Clone Detection System for Large Scale Source Code," IEEE Transactions in Software Engineering, vol. 28, no. 7, pp. 654-670, (2002-7).
14. Deursen, A.V., Marin, M., Moonen, L.: AJHotDraw: A Showcase for Refactoring to Aspects. In Proceedings of the Workshop on Linking Aspects and Evolution (LATE05). 4th International Conference on Aspect-Oriented Programming, (2005)
15. Ceccato, M., Marin, M., Mens, K., Moonen, L., Tonella, P., Tourwé, T.: Applying and combining three different aspect Mining Techniques, Software Quality Control, v.14 n.3, (2006), 209-231
16. Breu, S.: Extending Dynamic Aspect Mining with Static Information, Proceedings of the Fifth IEEE International Workshop on Source Code Analysis and Manipulation (SCAM'05), (2005) 57-65
17. Cojocar, G. S., Şerban, G. 2007. On some criteria for comparing aspect mining techniques. In Proceedings of the 3rd Workshop on Linking Aspect Technology and Evolution (Canada,. LATE '07). (2007)

A NEW GROUPING METHOD FOR XSL 1.0

Isidro Vila Verde

Feup, Faculdade de Engenharia da Universitdade do Porto
Rua Dr. Roberto Frias, 4200-465 Porto Portugal.
jvv@fe.up.pt

Abstract. One of the classic problems in XSL 1.0 is the nodes grouping. The known solutions are not trivial and have limited range of application. Additionally, the problem is not easy to state and until now all that has been done so far, is presenting it as use cases. In the this article we present a first approach to stating the problem in a formal way and introduce an algorithm based on a method known as "method of Muenchian", we identify XML document structures where this does not work well and we propose a new method for overcoming those limitations. In the end we identify the limits of our solution but we show that it generalizes the Muenchian method.

Keywords: XSL, XSLT, nodes grouping problem

1. Introduction

In XSL 1.0 there are several grouping techniques [1], [2], [3] and in XSL 2.0 there are support in proper elements of the language [4]. However, the support for XSL 2.0 is still not widely implemented in the current XSLT processors and does not solve all situations. As such it makes sense to analyze these techniques in XSL 1.0. This article appears as a consequence of the necessity to create a XSLT to transform into HTML list elements, the items of XML document that appear in mixing elements:

```
<!ELEMENT PARA (#PCDATA|item)*>
```

The search for a solution to this problem led to the research of grouping algorithms. In elapsing of this research some methods were found, but a more detailed analysis of these allow to conclude that these are limited to some very specific XML structures. This article intends to make a revision of one of these methods and with some alterations present a new algorithm.

We start by stating the problem in a formal way but it is not completely formalized. This is due to the fact the XSL specification was done in natural language and until now, so far we know there are not any formal specification[1]. We based our problem's formalization with ideas from some related works [7,8,9,10,11].

[1] Very recently and after finishing this paper, we found a formal specification for XSL on W3C Draft [12] but it is a document with 191 pages and we have not had the time to study it yet.

A grouping algorithm in XSL 1.0 is presented using the Muenchian method [1] and we show where it fails. To overcome those limitations we propose another method and show how it works even when the Muenchian method fails.
However, the new proposed method does not solve all grouping problems. Nevertheless, it is more generic than the previous one. The situations where it will not work are also identified.
In the following section the partial formalization of the problem is presented, using ideas from others [7,8,9,10,11]. In section 3 two algorithms are presented, one based on Muenchian method and our proposal for a new method. We discuss both on section 4 and finish with conclusions on section 5.

2. Stating the Problem

A XML document can be represented in a tree structure. Each DTD specifies the possible structures of a set of trees. These structures can be seen as being a type t of trees. For a given type t of trees zero or more XML documents d of type[2] t can exist and, theoretically, it can have an infinite number of types t .
Therefore, we can define the following sets:

$$D_t = \{d:\ d\ is\ a\ XML\ tree\ of\ type\ t\ \text{and}\ t\ \in T\}$$
$$T = \{t:\ t\ is\ some\ type\ of\ XML\ tree\}$$

D_t is the set of all XML documents of type t and T is the set of all possible types[3]

A XML document is composed by a content c and represented in a tree of type t . However, the same content[4] can be represented by trees of different types without modifying its meaning.
Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<persons>
        <person><age>20</age><name>Ana</name></person>
        <person><age>25</age><name>Joana</name></person>
        <person><age>20</age><name>Pedro</name></person>
        <person><age>25</age><name>Sofia</name></person>
</persons>
```

Fig. 1: A XML Document $d1$ representing a given content c

[2] A XML document is said valid, according to one given DTD, if its structure is conform to type t defined by that DTD.

[3] A tree structure can be explicitly defined or implicitly assumed from a XML document when it is not associated with any DTD or XSD or any other language.

[4] Here, by content , we mean the set of real world facts that the XML text represent

The type $t1$ for the XML document above is defined in Fig. 2

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT persons (person+)>
<!ELEMENT person (age,name)>
<!ELEMENT age (#PCDATA)>
<!ELEMENT name (#PCDATA)>
```

Fig. 2: Type Definition $t1$

We can say document $d1$ belongs to set D_{t1} , that is, $d1$ is one instance of type $t1$. D_{t1} is the set of all possible instances of type $t1$.
That is:

$$d1 \in D_{t1}$$

However the same content c can be represented as in Fig. 3, which is a tree of type $t2$ (defined in Fig. 4)

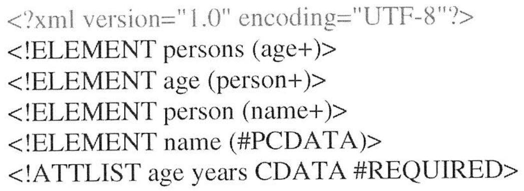
```
<?xml version="1.0" encoding="UTF-16"?>
<persons>
        <age years="20">
                <person><name>Ana</name></person>
                <person><name>Pedro</name></person>
        </age>
        <age years="25">
                <person><name>Joana</name></person>
                <person><name>Sofia</name></person>
        </age>
</persons>
```

Fig. 3: Another XML Document $d2$ representing the same content c as $d1$

To define this kind of relation between $d1$ and $d2$ we use the similar operator modified as follows:

$$d1 \underset{c}{\simeq} d2$$

This means $d1$ is similar to $d2$ in content.
Here, on this example, we transform the document of type $t1$ to a document of type $t2$ by grouping the elements person, which share the same value in its leaf age, under a new element age with an attribute years assigned to value of the old leaf age.



```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT persons (age+)>
<!ELEMENT age (person+)>
<!ELEMENT person (name+)>
<!ELEMENT name (#PCDATA)>
<!ATTLIST age years CDATA #REQUIRED>
```

Fig. 4: Type Definition $t2$

On a generic transformation we need to identify the leaves for grouping and to choose a name for the new grouping element in a such way that:

$$d1 \underset{c}{\simeq} d2$$

$$\textit{where} \quad d1 \in D_{t1} \text{ and } d2 \in D_{t2}$$

$$\text{and } t1 \neq t2 \text{ and } (t1, t2) \in (T x T)$$

Assuming we are able to identify the leaves, the problem is finding a function F that transform any XML document of type t1 into another document of type t2 such as:

$$F : D_{t1} \rightarrow D_{t2}$$

$$F(d1) = d2$$

$$\textit{where}\, d1 \underset{c}{\simeq} d2$$

$$\text{and } t1 \neq t2 \text{ and } (t1, t2) \in (T, T)$$

$$\text{and } d2 \textit{ groups some elements of } d1$$

$$\textit{based on leaves with the same value}$$

Obviously this is not a completely formal problem statement but it is a start and we can see what kind of problems we are dealing with.

3. XSL Grouping Methods

As the XSL[5, 6] is the main programing language to transform XML documents, the function F must be implemented in XSL.

We do not investigate how easy it is to implement function F in an imperative Language like Java or C++, but we suppose it will not be difficult. However, in XSL 1.0 the solution is not immediate because XSL has some particularities.

A XSL Transformer (XSLT) is set of template rules [8] where each rule consists of a pattern matching and a template. A XSLT processor receives as input a XML tree and starting from the root node it will transform each node according to a template defined in the XSLT. The selection of the template to use is made by the pattern matching of the rules with the processed node.

The XSL algorithm to implement the function F is not trivial due to not being enough to declare a set of templates which are applied to the nodes. It is also necessary to identify the groups of the target elements (person) that have the same value in the grouping node (age) and to guarantee that a template exists that will be executed for each one of those groups and not for each node of that group.

In favor of the clearness of the text the following conventions are assumed for the remaining portion of this article:

- The element that we intend to group will be called "target element"
- The sub-node (or leaf) used to group ascendant nodes will be called "grouping node"
- The parent element of the intended elements will be called "context".

Occasionally, to remember and to reinforce the idea, the name of the referred node will be placed between parentheses.

3.1 The Muenchian Method

A solution for the function F in XSL 1.0 was proposed by Steve Muenchian and published by Jeni Tennison on the book *"XSLT and XPath On The Edge"*[1].

A possible implementation of F to convert any XML of type $t1$ (defined in Fig. 2) to a XML tree of type $t2$ (defined in Fig. 4) based on the method of Muenchian, is presented in Fig. 5.

The target element here is the element person, the context is the element persons and the grouping node is the element age.

In Fig. 5, a key k is created (in line 3) for the target elements (person) and indexed by the value of the grouping node (age). Each entry of this key identifies a set of the target elements that have the same value in the grouping node. Thus this key will contain the sets of the target elements to be grouped together. This solves the first part of the problem: group identification.

From line 4 to 8 a trivial template, which applies to any node, is defined. This template simply copies the node and applies the correspondent rules to its children. Eventually, depending on the objectives, these 4 lines can be omitted or replaced.

In line 9 a template is defined which applies only to one of the elements of each set indexed in key k (in this implementation we chose the first target element in each set to match the template rule, but any element of these sets can be selected). The xpath expression on the match attribute guarantees that the template will be executed only once (and only once) for each set. It is in this template that the grouping will be performed. This solves the second part of problem: a template applied only once to each group.

Note that in the predicate the cardinal is only one ([count (...) = 1]), as it is known, when the union (|) of the current element (.) with the first element indexed by the key (key(' k ', age)[1]), results in a set of only one node, i. e., when the current element is the first element indexed by the key k. In all other cases the union result will contain two elements (the current and the first indexed element) therefore, the condition will not be verified. An alternative method to the use of the count function can be seen in reference [1].

From line 10 to 17 the new grouping element (age) is generated with the attribute years assuming the corresponding value, which is nothing more nothing less than the value of the grouping node.

From line 12 to 16 all target elements sharing the same value on the grouping node are copied. As those are joint together in the same set and indexed by key k it is enough to perform the copy operation over each element in that key.

```
1       <?xml version="1.0" encoding="UTF-8"?>
2       <xsl:stylesheet version="1.0"
      xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
3        <xsl:key name="k" match="person" use="age"/>
4        <xsl:template match="*|@*">
5         <xsl:copy>
6          <xsl:apply-templates select="@*|node()"/>
7         </xsl:copy>
8        </xsl:template>
9      <xsl:template match="person[
                         count(. | key(' k', age)[1]) = 1]">
10        <xsl:element name="age">
11       <xsl:attribute name="years">
             <xsl:value-of select="age"/></xsl:attribute>
12         <xsl:for-each select="key('k', age)">
13          <xsl:copy>
14           <xsl:apply-templates select="@*|node()"/>
15          </xsl:copy>
16         </xsl:for-each>
17        </xsl:element>
18       </xsl:template>
19       <xsl:template match="person"/>
20       <xsl:template match="person/age"/>
```

Fig. 5: A grouping algorithm based on the Muenchian Method

In line 19 an empty template for those target elements that are not the first ones in the respective set indexed in the key k. It is guaranteed, thus, that those elements are not copied a second time for the result as they are already present.

Finally in line 20 omits the copy of grouping nodes as the information contained in those elements is already present in attribute years. One notices that this is a decision that can be not taken in other situations.

In this section we presented a possible algorithm and noted, however, that there are other XSL grouping algorithms using the Muenchian Method as well. In the reference [1] there is a slightly different algorithm from the algorithm described here. Nevertheless, this one is a little bit more generic.

3.2 Two Keys Method

The Muenchian Method works well when the target elements belong all to the same context (that is, they are all children of the same parent element) and the parent does not have other children than the target elements. That is:

count((xpnodes)/parent:: *) = 1
and count((xpnodes)/parent::*/nodes()) = count((xpnodes)) (1)
where xpnodes is the xpath expression for selecting target elements.

In a real case, the probability of this scenario occurring, limits the use of this method. It will be most likely to find diverse elements that we intend to group, distributed by several nodes of the tree, that is, belonging to different contexts.

Let us see the case shown in Fig. 6, where we have elements person, that we intend to group, distributed by more than one context.

```
<?xml version="1.0" encoding="UTF-8"?>
<persons>
  <group n="1">
    <person><age>20</age><name>Ana</name></person>
    <person><age>25</age><name>Joana</name></person>
    <person><age>20</age><name>Pedro</name></person>
  </group>
<group n="2">
    <person><age>20</age><name>Rita</name></person>
    <person><age>20</age><name>Tiago</name></person>
    <person><age>25</age><name>Sofia</name></person>
  </group>
</persons>
```

Fig. 6: Example of a XML structure where Muenchian Method does not work well

In this example we have the target elements (person) in two different contexts. We have some elements in the context of the first element group and others in the context of the second element group. If this XML document is transformed by the XSLT in Fig. 5, the result will not make any sense as shown in Fig. 7. As it can be verified, the elements person of group 2 appear in the result as being of group 1. This result will hardly be desired because it breaks the information relations.

The result that, certainly, is expected from a grouping algorithm for this case is represented in Fig. 8. The XSL code in Fig. 5 does not work with this structure for two reasons. First, the group identification occurs only with the first element on the set, independently of the context used. So the groups whose value of the grouping node has already occurred in a previous context are simply ignored by the template in line 19. Second, this set omits the context of the elements. Thus, these are selected indistinctly and appear in the first group whose grouping node shares the same value.

The elements in key k are grouped by the value of the grouping node independently of the context where they exist, when they should be grouped by the value of the grouping node and by the context to which they belong. Thus it is necessary to modify the XSLT to take into consideration the context.

```
<persons>
  <group n="1">
    <age years="20">
      <person><name>Ana</name></person>
      <person><name>Pedro</name></person>
      <person><name>Rita</name></person> <!--wrong
-->
      <person><name>Tiago</name></person> <!--wrong
-->
    </age>
    <age years="25">
      <person><name>Joana</name></person>
      <person><name>Sofia</name></person> <!--wrong
-->
    </age>
  </group>
  <group n="2" /> <!--wrong -->
</persons>
```

Fig. 7: Result when use the XSLT in Fig. 5 to transform XML Document of Fig. 6

The XSL algorithm to transform the XML in Fig. 6 into the XML showed in Fig. 8, having in consideration the context where the target elements exist, is represented in Fig. 9.

This algorithm has the base structure of the algorithm on Fig. 5 but it contains some alterations derived from a different problem approach. The key of the Muenchian Method is divided in two, one to index the first occurrences on each context and other to index the remaining occurrences on that same context. These two keys identify the group (that is, the first element of that group) and the remaining elements of that same group. The way the keys are created allows the groups to be identified in context and not as it happens with the Muenchian Method.

The first key (k1) is created in line 3a for the first occurrence of a target element (person) which has a grouping node (age) with a distinct value of all others which have already occurred on the same context. The key is indexed by the ID generated for the element. This ID will be used for later verification of the existence or not of an element in this key. That is, it will be used to verify if an element identifies a group or not. It must be noticed that to each key it corresponds to one and only one element. Also, it must be noticed that there is an index for each distinct value of the grouping element in each context. So, if it has *m* contexts and each context has *n* distinct values the key will have *nxm* indexes which identify *nxm* distinct groups.

The second key (k2) contains all the target elements (person) sharing the same value in the grouping node (age) and being in the same context which are not the first occurrences. That is obtained by the use of the predicate [preceding-sibling::*/age = age] which guarantees the existence of, at least, a previous element in the same context and with the same value in the grouping node. In this key those elements are indexed by the ID of the first occurred element. Thus both k1 and k2 keys share the same indexes. And for a given index key k1 returns the element that occurs in first place, while key k2 returns the set of the elements that occur after the first and all share the same value in the grouping node. It can be said that in certain way they are complementary.

```
<persons>
  <group n="1">
    <age years="20">
      <person><name>Ana</name></person>
      <person><name>Pedro</name></person>
    </age>
    <age years="25">
      <person><name>Joana</name></person>
    </age>
  </group>
  <group n="2">
    <age years="20">
      <person><name>Rita</name></person>
      <person><name>Tiago</name></person>
    </age>
    <age years="25">
      <person><name>Sofia</name></person>
    </age>
  </group>
```

Fig. 8: Expected result after grouping XML document in Fig. 6

To generate the ID of the first occurred element to also index key k2, we use the axis preceding-sibling, with the predicate [age = current()/age]. This returns the set of the preceding elements sharing the context and with the same value in the grouping node. The first one of these is the element that we are interested in but, as it is known, when this set is gotten by the axle preceding-sibling the first one to occur in the XML document appears in the last position, reason why we use a second predicate with function last().

Once these two keys defined, it is now easy to define a XSL template for each group and, inside of this, select the elements sharing the context and with the same value in the grouping node. In line 9 the predicate is used to impose the condition of the element being present in key k1, that is, to guarantee that it represents the group. We have, thus, a template executed for each group that we want to generate.

The selection of the elements to be copied to this new element is performed in line 12 using the second key k2. As this key is indexed by the ID of the first occurrence, that is, by the current element, to get all the elements for the group it is enough to index this key by the current element generated ID. Since the current node must also be copied and it is not present in the key k2, it is also included on the XSL match attribute using the union operator.

The remaining part of the algorithm, as it can be observed, remained relatively unalterable with respect to Fig. 5.

```
   <?xml version="1.0" encoding="ISO-8859-1"?>
   <xsl:stylesheet version="1.0"
   xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
  <xsl:key name="k1" match="person[
        not(preceding-sibling::*/age = age)]"
   use="generate-id()"/>
    <xsl:key name="k2" match="person[
    preceding-sibling::*/age=age]"
   use="generate-id(preceding-sibling::*[
 age=current()/age][last()])" />
    <xsl:template match="*|@*">
     <xsl:copy>
      <xsl:apply-templates select="@*|node()"/>
     </xsl:copy>
    </xsl:template>
    <xsl:template match="person[key('k1',
                                generate-id())]">
     <xsl:element name="age">
      <xsl:attribute name="years">
        <xsl:value-of select="age"/>
      </xsl:attribute>
      <xsl:for-each select=".|key('k2',
                                generate-id())">
       <xsl:copy>
        <xsl:apply-templates select="@*|node()"/>
       </xsl:copy>
      </xsl:for-each>
     </xsl:element>
    </xsl:template>
    <xsl:template match="person"/>
   </xsl:stylesheet>
```

Fig. 9: Two keys method

4 Discussion

There are other solutions for this kind of grouping problems but as they don't use keys they have complexity of order O(N^2) (see of Michael Kay's argument in thread initiated in the reference [2]). This solution, being based on keys as the Muenchian Method, has complexity order equivalent, that is, O(N log N). However, we have not made any performance tests over large datasets yet. This is essential but at the present stage of our work, we have not had time to perform those tests.

The presented solution removes the limitation of all target element sharing the same context. Thus, the first condition in (1) could be eliminated. That is, this solution works for any XML document structure where:

count((xpnodes)/parent::*/nodes()) = count((xpnodes)) (2)
where xpnodes is the xpath expression for selecting target elements.

Of course it still has the limitation of all children elements of contexts being target elements and nothing else. But this is a little more common situation than the restriction for Muenchian Method.

5 Conclusions

In this article we present an algorithm of grouping based on the method of Muenchian and show what conditions must be satisfied for it to work. We show why it is too limited for the real world and we present an alternative method to solve one of those limitations. We also show the limitations of our own solution although the range of amplitude of applicability is much more. Finally we must say our solution is not a generic solution nor even resolve our original problem (grouping continuous items under mixed elements) but it is a step forward and we think we can work on a solution for that problem based on the two keys methods with some changes at predicates level. This will be our next goal and we hope to present the solution on a future article.

The complexity issue is particularly relevant for large datasets and the performance tests have not been done yet. We need to work urgently on that component to consolidate our solution.

However, the big challenge for the future will be stating the problem in a formal way. For achieving that, we need to study and learn from the W3C draft XQuery 1.0 and XPath 2.0 Formal Semantics [12]

Acknowledgments

This work was based on a previous paper presented in "IADIS International Conference WWW/Internet", Oct 2006 and was partially support by Departamento de Engenharia Electrotécnica e de Computadores of FEUP and was done as a part of assignments of the course Methodologies for Scientific Research of Doctoral Program in Informatics Engineering

References

1. Jeni Tennison, "*XSLT and XPath On The Edge*", 2001, Unlimited Edition, USA.

2. Sergiu Ignat , "Recursive grouping - simple, XSLT 1.0, fast non-Muenchian grouping method", In a mailing list, Dec, 2004, http://www.biglist.com/lists/xsl-list/archives/200412/msg00865.html
3. M. David Peterson, "New Alternative to Muenchian Method of Grouping", In a mailing list, Dec, 2004, http://www.xsltblog.com/archives/2004/12/new_alternative.html
4. Bob DuCharme, "Grouping With XSLT 2.0", In XML.com, Nov, 2003, http://www.xml.com/lpt/a/2003/11/05/tr.html
5. W3C, "XSL Transformations (XSLT) Version 1.0", W3C Recommentation, Nov, 1999, http://www.w3.org/TR/xslt
6. W3C, "XML Path Language (XPath) Version 1.0", W3C Recommentation, Nov, 1999, http://www.w3.org/TR/xpath
7. P. Wadler. "A Formal Semantics of Patterns in XSLT and Xpath" In Markup Tecnologies, Philadelphia, Dec, 1999. Revision version in Markup Languages, MIT Press, Jun, 2001.
8. G. J. Bex, S. Maneth, and F. Neven. "A formal model for an expressive fragment of XSLT". 1st International Conference on Computational Logic, pp :1137-1151, Lecture Notes in Artificial Intelligence, volume 1861. Springer, 2000
9. S. Flesca, G. Manco, E. Masciari, L. Pontieri, and A. Pugliese. "Fast detection of XML structural similarity", In Transactions on Knowledge and Data Engineering, Volume 17, Issue 2, pp:160-175, Feb, 2005
10. T. Milo, D. Suciu and V. Vianu, "Typechecking for XML transformers " In Journal of Computer and System Sciences, Volume 66, Issue 1, pp:66 - 97, Feb, 2003
11. G. Gottlob, C. Koch, and R. Pichler. *"Efficient Algorithms for Processing XPath Queries"* In Proc. VLDB'02, 2002
12. W3C, "XQuery 1.0 and XPath 2.0 Formal Semantics", W3C Draft, Jan, 2007, http://www.w3.org/TR/xquery-semantics/

Using Learning Styles and Neural Networks as an Approach to eLearning Content and Layout Adaptation

Jorge Mota[1]

[1] PRODEI - Faculty of Engineering of University Of Porto, Portugal
Museu8bits@gmail.com

Abstract. The eLearning's trend is changing; learning content has become the key issue of current eLearning. The eLearning in Portugal as in many other countries is not yet so widely used as an alternative to other forms of training: as is the case of traditional classroom. This is because learners don't identify their own learning style in the way the presentation of education content are done in the majority of eLearning material produced today, or not feel enough customization in the content to their own needs. This paper describes the design, development and implementation of the model of an adaptive course player that uses Kolb learning styles[1] and neural networks to model learners and dynamically generates navigation paths and layout adaptation. The system implements adaptation of individual recommendations and content adaptation based on learning styles, previous learner knowledge, learner's progress and persistence of their own preferences. This is a ongoing work, and we are using our own experience producing eLearning content and an actual eLearning project to evolve the way difficult domain content can be presented to different individuals or stereotyped groups (similar conceptual understanding) with a disparity of objectives, different kind of professional roles, dissimilar previous knowledge and different context.

Keywords: Adaptive Hypermedia; eLearning; Adaptive Educational Systems, navigation support, user modeling, intelligent tutoring systems, student models.

1 Introduction

eLearning has emerged as a prime topic in Portuguese educational strategy more than a decade ago, but it hasn't yet gained sufficient stakeholders and satisfactory results to be accepted without restrictions in all kind of educational contexts: long life learning, universities, schools, companies and other kind of organizations. Learning content has become the key issue of current e-Learning.

The first adoption phase of eLearning in Portugal was focused on platform's technology. The most important universities, yearly technology adopters in companies, business associations and research labs invest in testing, experimented and developed platforms for eLearning. Unfortunately, much time, money and enthusiasm were lost in these programs forgetting the most important: quality eLearning content in Portuguese language and well trained professionals in the area.

Modern developments in the field of content standardization for learning objects and metadata (LOM, SCORM)[2, 3] open new possibilities for adaptive educational media to work with masses of content and learning objects[4]. From our point of view, the appropriate modeling of the learner's needs and preferences, representation of pedagogical strategies, learning designs and assets as well as the runtime reconciliation of these elements, are the key issue for next generation eLearning. This can be done with the help of some kind of learning styles classification and a mechanism to produce personalized content.

In our own experience producing and implementing eLearning content, the previous knowledge of the subject matter, predominant learning style, and progress results combined with user control for a particular content presentation style are the main adaptive attributes to model a successful eLearning 2.0 content. In our work, we design and implement a learner model based on Kolb learning style inventory classification[1] and a dynamically generated presentation, Personalized Learning Paths, based on learning styles, previous knowledge of the subject, progress results and persistence of learner educational elements preferences. Some other work was done in this field using similar strategies: this is the case of ALE system developed at Fraunhofer Institute for Applied Information Technology[4] based on Felder-Silverman learning style classification, INSPIRE[4] an AEH-System that uses a learning style model based on Honey and Mumford[5], TANGOW[6, 7] based on learning styles by Felder and Soloman[8] which represents the profile in the model. Our model innovates in a way that we use not only a different learning style model based on Kolb inventory styles[9] but also four axis of adaptive attributes used on fly by a learning neuronal network engine that promotes recommendations on presentation layout and permits that all the time the learner has optional control in the GUI to allow users to adapt the content presentation. Based on individuals' previous experiences, the system adapts the weights in the learner model and suggests the new recommendations based on the new model parameters.

2 Personalized and Dynamic Content Presentation and Navigation

Learners' pedagogical and contextual parameters are inputs to the reconciliation engine that creates the personalized content in the sense of picking the right learning designs and activities [10]. Adaptivity in learning experience is accomplished by choosing the learning paths that suit the knowledge level and the acquired competencies of the learner. The core concept of our design is the Adaptive Hypermedia (AH) System, this is build as a model of the individual user and apply it for adaptation to that same user[11].

In our design, we use two types of adaptability:

1. Adaptive Presentation

2. Adaptive Navigation

For the first type we use three methods of adaptivity: Kolb Learning Styles, individual and global performance and user's preferences. For the second type we use a subject matter pre-test mapped to each learning object in the repository.

The most important adaptive methods are the learning styles and we design a learner model, which is determined with the Kolb (1984) learning styles inventory[1].

2.1 Kolb Learning Styles[1]

Kolb set out four preferences for learning:

- Feeling ("Concrete Experience" – CE)
- Watching ("Reflective Observation" – RO)
- Thinking ("Abstract Conceptualization – AC")
- Doing ("Active Experimentation – AE")

The combination of these styles gives us four learning styles or types:

- Reflector (Watching and Doing, Concrete-Reflective)
- Theorist (Watching and Thinking, Abstract-Reflective)
- Pragmatist (Thinking and Doing, Abstract-Active)
- Activist (Doing and Feeling, Concrete-Active)

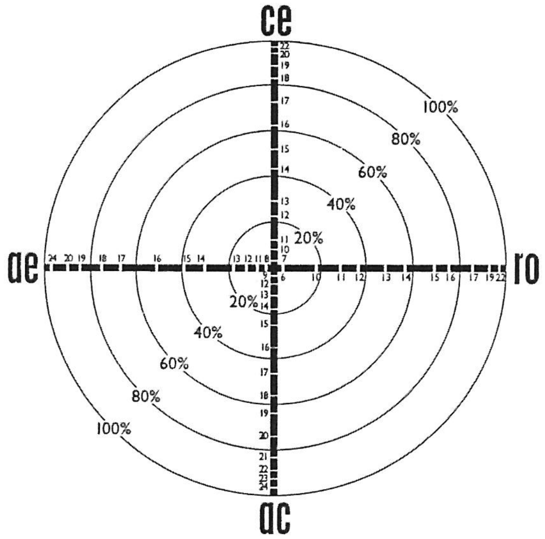
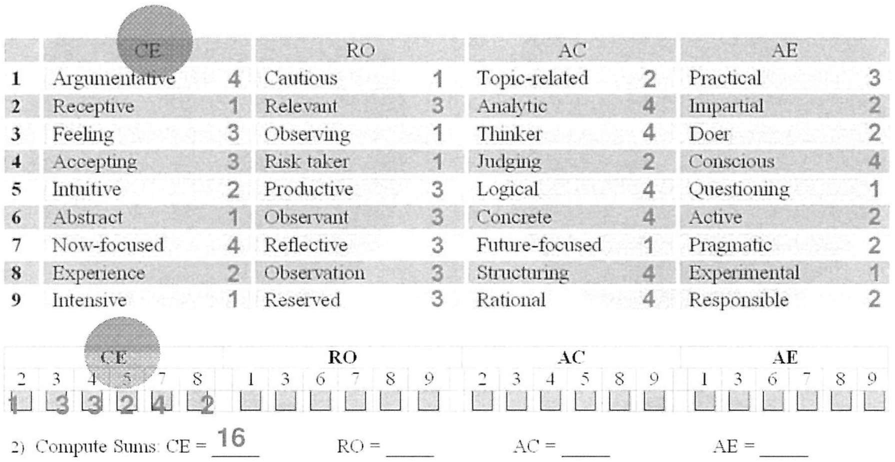


Fig. 1. Kolb's Learning Style Inventory Graph. Reflexive-Active and Concrete-Abstract dimensions[1].

The Kolb inventory uses 9 sets (columns) of 4 words (rows) to locate the learner on 2D space. The learner must arrange each row of 4 words assigning a 1, 2, 3 or 4 value to the words that better suit their learning feeling. In the end we must transport the values to corresponding semi-axis, using a pattern of words.

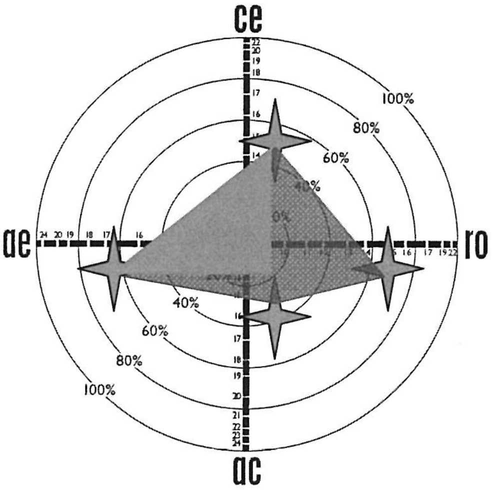


	CE		RO		AC		AE	
1	Argumentative	4	Cautious	1	Topic-related	2	Practical	3
2	Receptive	1	Relevant	3	Analytic	4	Impartial	2
3	Feeling	3	Observing	1	Thinker	4	Doer	2
4	Accepting	3	Risk taker	1	Judging	2	Conscious	4
5	Intuitive	2	Productive	3	Logical	4	Questioning	1
6	Abstract	1	Observant	3	Concrete	4	Active	2
7	Now-focused	4	Reflective	3	Future-focused	1	Pragmatic	2
8	Experience	2	Observation	3	Structuring	4	Experimental	1
9	Intensive	1	Reserved	3	Rational	4	Responsible	2

CE						RO						AC						AE					
2	3	4	5	7	8	1	3	6	7	8	9	2	3	4	5	8	9	1	3	6	7	8	9
1	3	3	2	4	2	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□	□

2) Compute Sums: CE = 16 RO = _____ AC = _____ AE = _____

Fig. 2. Kolb's Learning Style Inventory words. The red numbers are an example.



RO: 13
AC: 14
AE: 17
CE: 17

Fig. 3. Example of Kolb's Learning Style Inventory graph[1]. The area means the predominant learning style.

In his research Kolb concludes that no learner has one single style, we can even say that the limit has as many styles as there are individuals. In our design we use the following designations for the kolb learning styles: Reflector, Pragmatist, Theorist and Activist[5]. (fig 4).

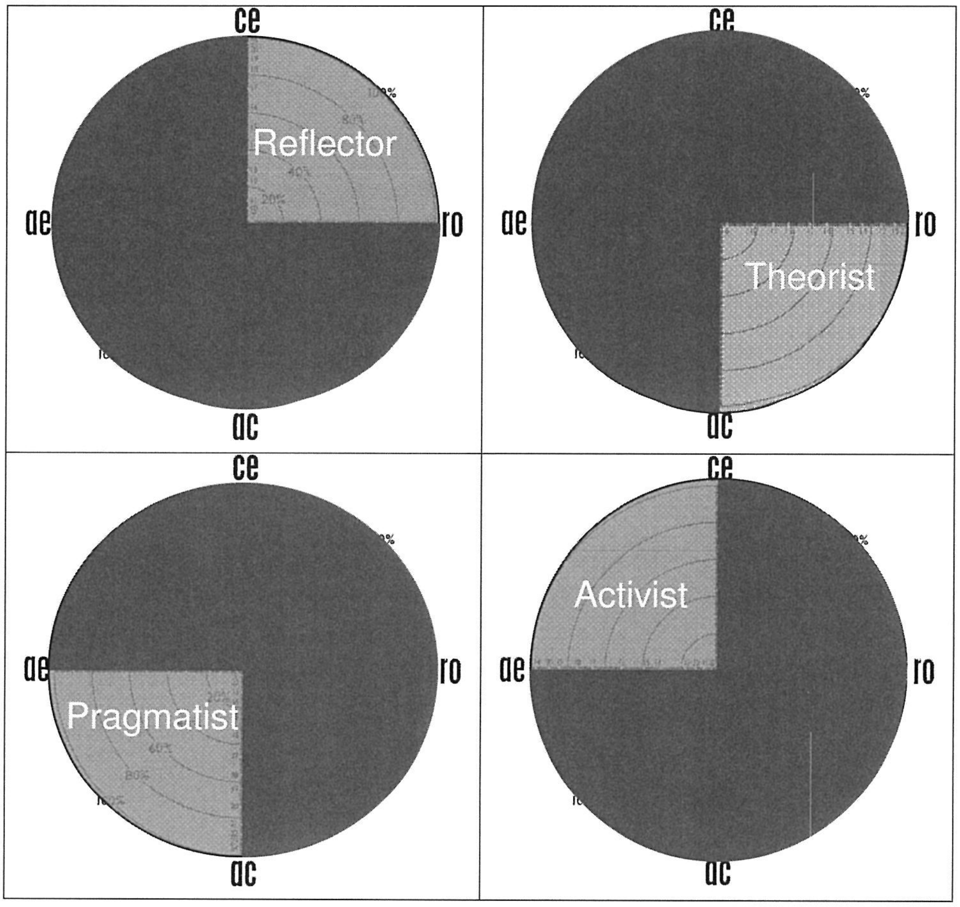


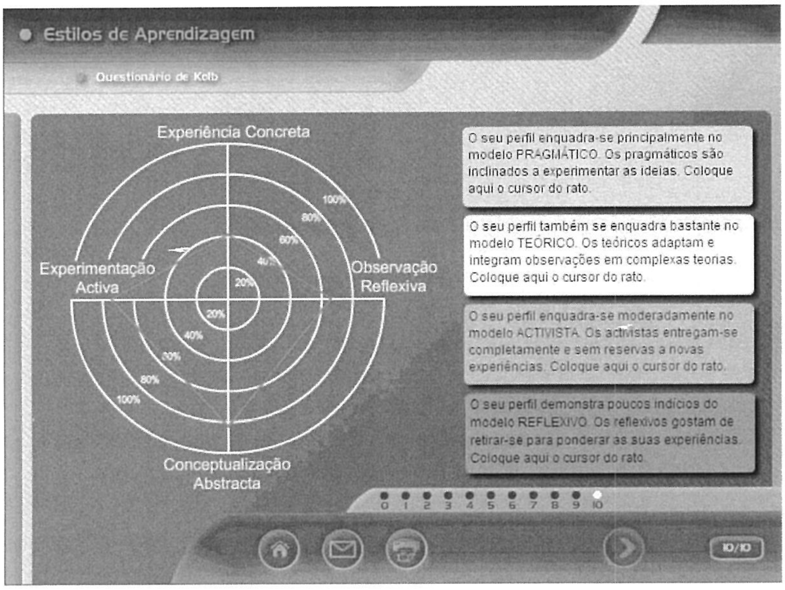
Fig. 4. Modified Kolb's Learning Styles[5].

Our design uses a Drag and Drop interface to process the self-administered questionnaire at the beginning of a new course. We present the results using a graph (figure 5) and we use color coding to distinguish the most predominant learning styles from the others. The results are then saved to a XML file as adaptive attributes.

```
<rede.neuronal>
  <proximo.aprendente>100</proximo.aprendente>
    <Aprendente Id=”1”>
      <Kolb.Inicial>
            <Data>07012008</Data>
            <Pragmatico>1</Pragmatico>

              <Teorico>0</teorico>
              <Reflexivo>0</Reflexivo>
              <Activo>0</Activo>
```

```
        </Kolb.Inicial>
            ...
```



O seu perfil enquadra-se principalmente no modelo PRAGMÁTICO. Os pragmáticos são inclinados a experimentar as ideias. Coloque aqui o cursor do rato.

O seu perfil também se enquadra bastante no modelo TEÓRICO. Os teóricos adaptam e integram observações em complexas teorias. Coloque aqui o cursor do rato.

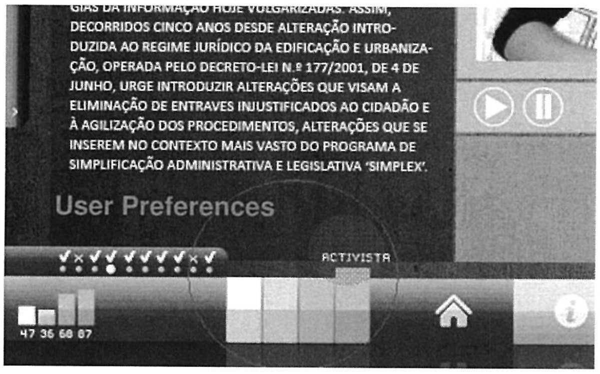
O seu perfil enquadra-se moderadamente no modelo ACTIVISTA. Os activistas entregam-se completamente e sem reservas a novas experiências. Coloque aqui o cursor do rato.

O seu perfil demonstra poucos indícios do modelo REFLEXIVO. Os reflexivos gostam de retirar-se para ponderar as suas experiências. Coloque aqui o cursor do rato.

Fig. 5. Our implementation of Kolb's self-administered questionnaire[9].

2.2 Learning element Sequencing

Adaptivity in learning experience is accomplished by choosing the learning paths that suit the knowledge level and the acquired competencies of the learner[12]. In our design this is measured by the engine service based on the assessment results and on the learner's consumption performance of the Los (Learning Objects). Learning paths are portions of the concept domain ontologies. These ontologies represent essentially the curriculum constructs.



GIAS DA INFORMAÇÃO HOJE VULGARIZADAS. ASSIM, DECORRIDOS CINCO ANOS DESDE ALTERAÇÃO INTRODUZIDA AO REGIME JURÍDICO DA EDIFICAÇÃO E URBANIZAÇÃO, OPERADA PELO DECRETO-LEI N.º 177/2001, DE 4 DE JUNHO, URGE INTRODUZIR ALTERAÇÕES QUE VISAM A ELIMINAÇÃO DE ENTRAVES INJUSTIFICADOS AO CIDADÃO E À AGILIZAÇÃO DOS PROCEDIMENTOS, ALTERAÇÕES QUE SE INSEREM NO CONTEXTO MAIS VASTO DO PROGRAMA DE SIMPLIFICAÇÃO ADMINISTRATIVA E LEGISLATIVA 'SIMPLEX'.

Fig. 6. Player learner Preferences Manual Graph Options.

Adaptability in learning experience is accomplished by choosing learning activities that suit the learner's pedagogical parameters and preferences. Being adaptable implies that the learners assume responsibility within the designated limits, and the also have freedom, yet guidance[12]. In our design we have a manual option graph (figure 5) that allows the learner to choose any of the available layouts for the content.

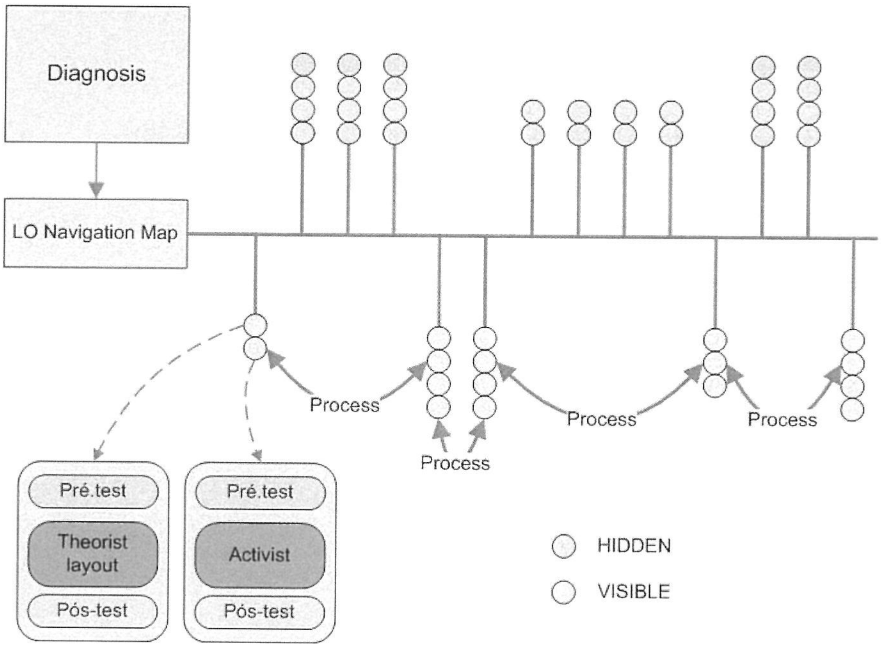


Fig. 7. CeLIP (Cesae eLearning Intelligent Player) Map Navigation Strategy.

Depending on the subject, their topic might have more or less presentation layout options. Learners with different learning styles react in different ways therefore they require different types of support when consuming the same learning object. This demarcation in support is provided not only for the search of an appropriate learning object, but also for the consumption of that learning object. Other important sequencing strategy is imposed by the kind of hidden options imposed by the initial diagnosis and ontological maps representing the curriculum.

2.3 Content/Presentation Adaptation

Targeting personalization, being adaptive and adaptable constrain the learning content to be developed and exploited by CeLIP (Cesae eLearning Intelligent Player). In our actual design each resource/page is developed, by authors and instructional designers, coding them in template pages manually. Each of these resources has metadata and can be reused in the development process of other courses. CeLIP can use contents like video, audio, text/graphics and interactivity simulations. CeLIP exploits the standardized technologies, such as SCORM 1.3 for learning objects.

The development of learning objects and learning designs should be coherent in order to prevent disharmony between these two. To overcome the Frankenstein effect [12] CeLIP employs only four types of final assembly layouts and the neuronal network engine tries to preserve the same style during the entire course using different weights for learning styles, performances as well as manual user preferences.

CeLIP determines the sequences of the learning objects at the very beginning, and an adaptive hidden strategy occults any LO that is considered not need to obtain the goals and objectives of the course. A primary aspect of content creation involves the curriculum analysis and accordingly the development of the ontological domain maps.

Another content creation's aspect is the development of knowledge representations for domains and learners. In order to match learner's knowledge to the knowledge designated for the domain, there should be a common representation model. However the representation for the learner will be let to evolve while the domain representation is bonded by the curriculum[12].

CeLIP uses four type of pedagogical layout strategies mapped to the four basic main styles defined by Kolb. We use sets of didactical elements composed in a way that the learner "feels at home".

3 CeLIP Player Architecture

CeLIP – Cesae eLearning Intelligent Player integrates new principles and tools in the field of Learning Design and Artificial Intelligence. This player uses a MLP (Multilayer Perceptron) neural network (figure 8) to predict the next presentation layout. This neural network is composed by layered arrangement of artificial neurons in which each neuron of a given layer feeds all the neurons of the next layer. This model forms a complex mapping from the input to the output. Our model is trained with the back propagation (BP) learning algorithm. This neuronal network is the core of our AI engine. Each time de engine process a new select, the state of each parameters is saved in a repository as a xml file. Our actual design only permits that the neural network operates on the behavior of one learner, don't permit global interaction between learners' models.

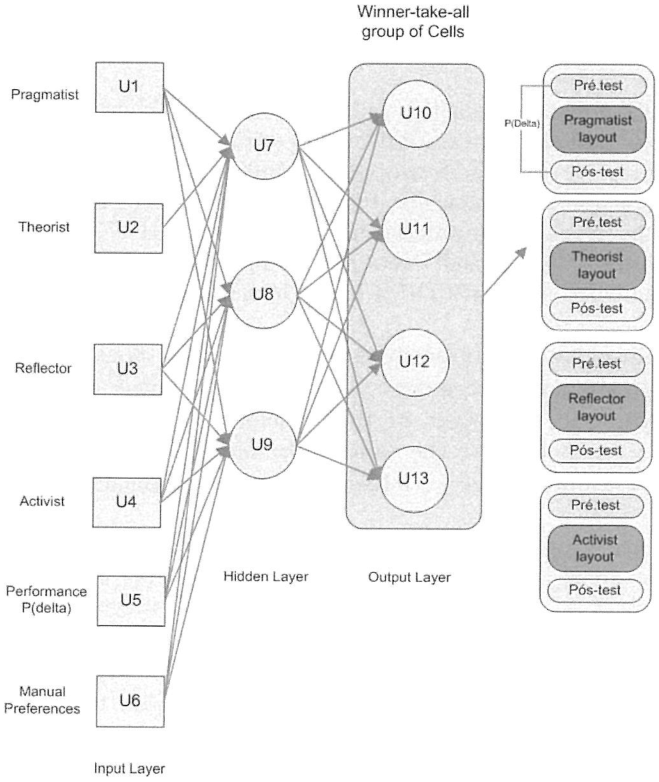


Fig. 8. CeLIP(Cesae eLearning Intelligent Player) Neurocontroller Architecture.

4 Example Workflow

Firstly, CeLIP (Cesae eLearning Intelligent Player) determines and employs a diagnosis in order to create a structure of LO's (Learning Objects) to cover the unit of study. The unit of study represents a portion of the curriculum domain map. This portion is evaluated respecting the knowledge level and the learner's acquired skills in order to decide which learning objects to be delivered.

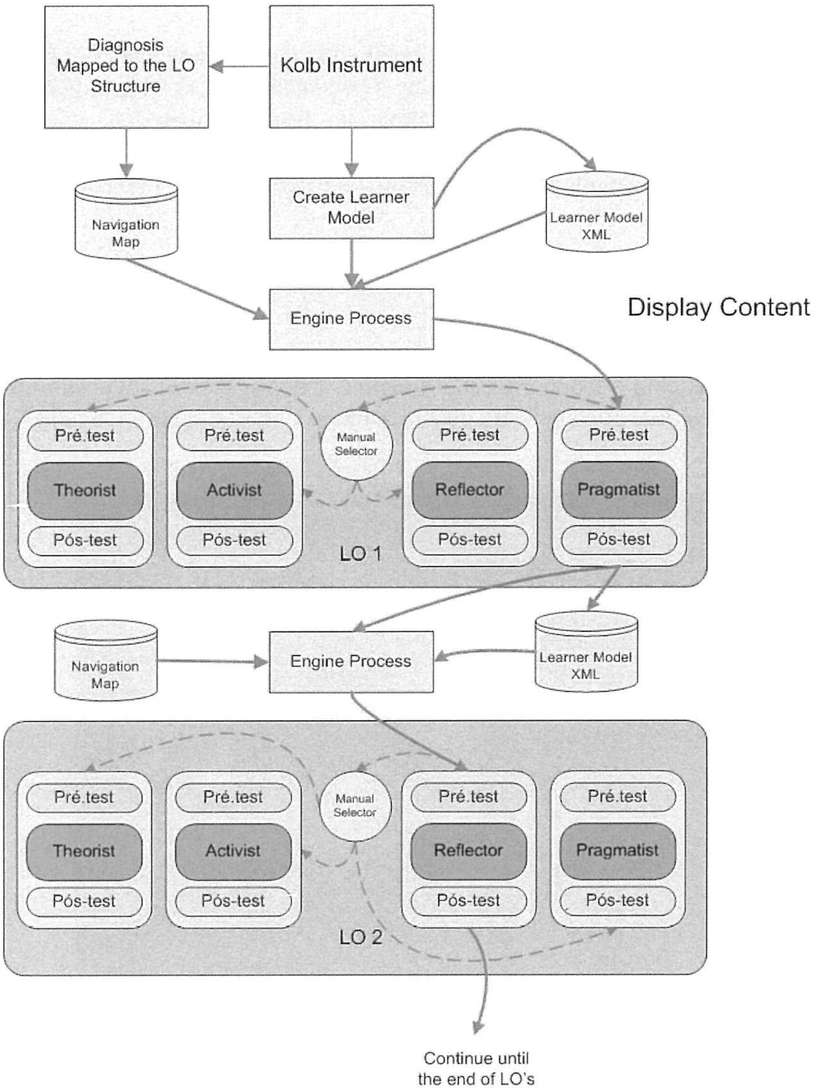


Fig. 9. CeLIP(Cesae eLearning Intelligent Player) Map Navigation Strategy.

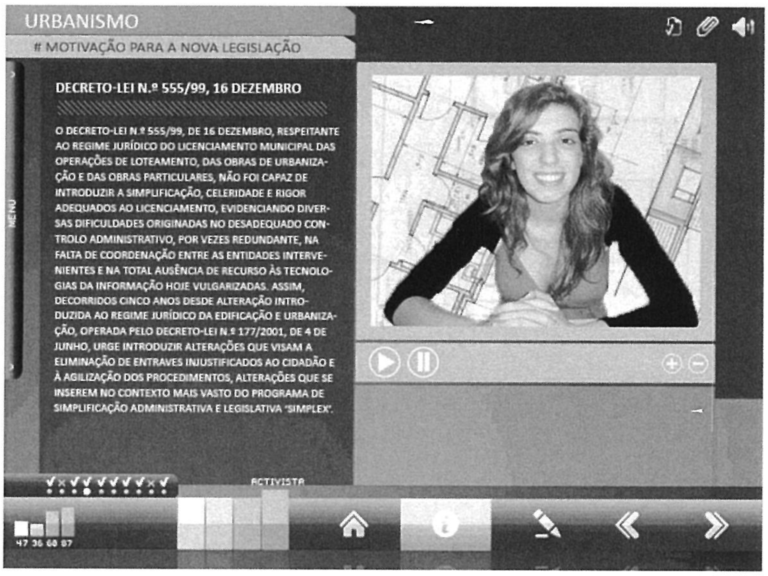
The type of activities and presentation that harbor this chain of objects is determined by using the pedagogical and contextual parameters (learning styles, performance and user preferences).

For each step advance in the navigation structure, CeLIP searches and finds learning objects that best suit learning style of the learner, their preference and performance. Notice that, primarily the objects will have to suit the corresponding portion of the domain as well as a set of concepts and skills.

The workflow presented in (Figure 8) highlights the personalization process performed by CeLIP. The key stages in creating a personalized eLearning experience are modeling the learner, choosing an appropriate learning approach, selecting appropriate content with customized learning objects. The selection of LOs is dependent on the domain and the learner's existing knowledge on that domain.

5 Results and Future Work

We had implemented a first prototype of CeLIP (Cesae eLearning Intelligent Player) and we are now producing a course for central region of Portugal local authorities that become the first eLearning content using this technologies in Portuguese Language.



DECRETO-LEI N.º 555/99, 16 DEZEMBRO

O DECRETO-LEI N.º 555/99, DE 16 DEZEMBRO, RESPEITANTE AO REGIME JURÍDICO DO LICENCIAMENTO MUNICIPAL DAS OPERAÇÕES DE LOTEAMENTO, DAS OBRAS DE URBANIZAÇÃO E DAS OBRAS PARTICULARES, NÃO FOI CAPAZ DE INTRODUZIR A SIMPLIFICAÇÃO, CELERIDADE E RIGOR ADEQUADOS AO LICENCIAMENTO, EVIDENCIANDO DIVERSAS DIFICULDADES ORIGINADAS NO DESADEQUADO CONTROLO ADMINISTRATIVO, POR VEZES REDUNDANTE, NA FALTA DE COORDENAÇÃO ENTRE AS ENTIDADES INTERVENIENTES E NA TOTAL AUSÊNCIA DE RECURSO ÀS TECNOLOGIAS DA INFORMAÇÃO HOJE VULGARIZADAS. ASSIM, DECORRIDOS CINCO ANOS DESDE ALTERAÇÃO INTRODUZIDA AO REGIME JURÍDICO DA EDIFICAÇÃO E URBANIZAÇÃO, OPERADA PELO DECRETO-LEI N.º 177/2001, DE 4 DE JUNHO, URGE INTRODUZIR ALTERAÇÕES QUE VISAM A ELIMINAÇÃO DE ENTRAVES INJUSTIFICADOS AO CIDADÃO E À AGILIZAÇÃO DOS PROCEDIMENTOS, ALTERAÇÕES QUE SE INSEREM NO CONTEXTO MAIS VASTO DO PROGRAMA DE SIMPLIFICAÇÃO ADMINISTRATIVA E LEGISLATIVA "SIMPLEX".

Fig. 10. Actual Instance of CeLIP -Cesae eLearning Intelligent Player.

We had found a lot of issues that we must investigate in future work: neuronal network learning parallelism; IMS LIP compatibility (by IMS Global Learning Consortium Inc.) ; Multi model approaches to model learner; time based learning (historical); short and long time learning duality. At end some authors expressed skepticism concerning the viability and validity of using learning style of the learner to adapt or personalize a learning environment to suit the needs of the learner[13].

6 Conclusions

In the present paper we have described the implementation of adaptive methods for content sequencing and adaptive presentation based on learning styles preferences, adaptive hiding result of a diagnosis test and a AI engine using a neuronal network that process the predictions of the best presentation layout for the next LO (learning Object) in the navigation sequence. This architecture is currently in the phase of implementation. We had implemented a user control in the GUI to allow learners to adapt the content presentation. In our first public presentation for "local authorities", we received good feedback from them.

References

1. Kolb, D.A., *Experiential Learning - experience as the source of learning and development.* 1984, New Jersey: Prentice Hall P T R.
2. Philips Dodds, A.P., *SCORM 2004 3rd Edition Sequencing and Navigation (SN) version 1.0,* ADL, Editor. 2006.
3. Philip Dodds, S.E.T., *Sharable Content Object Reference Model (SCORM)® 2004 3rd Edition.* 2006, ADL.
4. Katja Reinhardt, t.A., and Marcus Specht, *Adaptive Course Player for Individual Learning Styles.* Adaptive Hypermedia and Adaptative Web-Based Systems, 2004: p. 442.
5. P. Honey, A.M., ed. *Manual of Learning Styles* 1982, P. Honey: London.
6. P. Parades, P.R., *Considering Sensing-Intuitive Dimension Exposition-Exemplification in Adaptive Sequencing.* Lecture Notes in Computer Science : Proceeding of the Second International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems, 2002: p. 556-559.
7. P. Parades, P.R., *Considering Learning Styles in Adaptive Web-Based Education.* Proceedings of the 6th Word Multiconference on Systemics, Cybernetics and Informatics en Orlando, Florida, 2002: p. 481-485.
8. R. M. Felder, B.A.S. *Index of learning Styles.* 1999 [cited 23-12-2007]; Available from: http://www.ncsu.edu/felder-public/ILSpage.html.
9. Alice Y. Kolb, D.A.K., *The Kolb Learning Style Inventory—Version 3.1 2005 Technical Specifications.* 2005, Haygroup: http://www.haygroup.com/tl/Downloads/LSI_Technical_Manual.pdf.
10. Colan O, V.W., *Evaluating the Multi-model, Metadata-driven Approach to producing Adaptive eLearning Services.* Third International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH2004) Proceedings, Eindhoven, The Netherlands, 2004.
11. Brusilovsky, P., *Methods and techniques of adaptive hypermedia.* User Modeling and User-Adapted Interaction, 1996: p. 6,2-3,87-129.
12. Ali Turker, I.G., Owen Conlan, *The Challenge of Content Creation to facilitate Personalized eLearning Experiences,* in *International Journal on E-Learning (IJEL), Vol.5 January (2006).* 2006.
13. Melis, E.M., Rachada, *They Call It Learning Style But It's So Much More.* 2004.

Competency Management Articulation for Human Resources

Fernando Teodósio

Faculty of Engineering, University of Porto
Rua Dr. Roberto Frias,
4200-465 Porto, Portugal
sociteo@mail.telepac.pt

Abstract. In recent years the approach to competences has gained great popularity due to process and organizational reengineering need. Taking opportunity on some recent work in this area dealing challenges that human resources face to develop planning training, I intend to identify several guidelines to develop a future architecture in a practical implementation. At this article is presented the concept development of competency management.

Keywords: Competencies, Intellectual Capital Management, Knowledge Management, Human Resources, eLearning

1 Introduction

For some time we live in change, driven mainly by new technologies and globalisation. Indeed, competition is increasing, and customers are increasingly educated and with more power, it encourages organizations having to adapt to almost frenetic changes without losing its effectiveness and positioning [1]. It is obvious that the old way of working is inadequate, now the key to organizational success and excellence focuses on people and their management. Companies realized that, in addition to the technologies and processes are the employee's competences and knowledge, being more prepared, they can provide added value to the organization.

It is essential knowing how to manage knowledge and human capital, coaching and emotional intelligence in organisations. Staff is needed, regardless their level, to engage objectives. It is necessary that everything can be controlled by the organization, even when the person is no longer present. It became necessary to measure the contribution (intellectual capital [7]), besides the capital structure, to assess the organization of a real way, accounting on its potential and know-how. Thus, trend goes through more effective measurement systems usage, mainly implementations that make easy exchanging knowledge and produce new approach of understanding business and professionals [2, 3]. This technological society needs to humanize.

The competencies emerged to make possible manage company employees abstract skills, knowledge and capabilities [4]. These will meet employees performance because goes beyond mere knowledge. The traditional training and human resources are guided by the following framed principles: professional dignity, equal opportunities in apposite training access for real training needs. Competences delivers an instrument to control holistic expectations.

There are two different perspectives on approaches to competences: (1) competency management resource-oriented and (2) eLearning-oriented competencies.

Training aims: (1) to promote permanent learning; (2) improve performance; (3) enhance work – "training / action"; (4) creating new employees' competencies dealing with organization mission; (5) predispose all professionals to processes of change strengthening the organizational culture and develop the ability to "learning to learn".

Training is encompassed with the principle of: (1) universality because covers all workers, whatever their role or category; (2) functional in order to meet service needs; (3) decentralised for seeking place diversity; (4) multidisciplinary since support all necessary branches of knowledge to services, taking into account the evolution of knowledge and technological means; (5) complementarily for the reason that it is a natural result of the process education.

There are gaps in human resources and enterprises on computer usage and internal information systems management. Although worldwide invested effort on organizational knowledge management, the fact that this is an inherently people dependent activity makes complex from the beginning reach a satisfactory level of success. This complexity allied to the fact that this is a very wide scope area has hampered standards emergence, each author or group of research, has its own definition, methodology, different annotations and so on.

In this article is explained the training needs gathering importance through diagnose techniques with association between company and worker motivations.

I evaluate practical implementation approaches including technological analysis. The assessment added value is state-of-the-art confirmation and presents a method to handle encountered shortcomings for further research. From here are withdrawn conclusions on the future projects and research to help develop levels of employability, assessment and training, to achieve better conditions for competitiveness.

2 Problem Analysis

An enterprise-level successful informatics engineering solution introduction needs something more than just technology, because needs to think how business process fits on organizational pre-requisites.

Organizations need for an activity that explicitly deals with a knowledge component concerned in business activities has led to knowledge management emergence field. This activity seeks to identify, treat and provide relevant knowledge to the organization business, both by content management, and existing intellectual capital management [7].

To create more effective, quality and relevant training have to follow a systematic methodology integrated by a first phase of organization training needs identification and diagnosis of the needs, followed by training development and implementation, and finally performed training evaluation.

Needs diagnosis will enable information obtained from various kinds including: (1) organizational strategic objectives, in short and medium term; (2) employee's performance actual levels; (3) training priority; (4) organizational problems and dysfunctions that can be solved with training. This phase since process beginning enhances all workers participation, whatever their role, function or category hierarchy.

Not disregarding techniques used to make diagnosis such as questionnaires or interview surveys, observation or group dynamics should be used a competency acquiring plan implemented with all workers participation.

Organization business and each department are knowledge and individual personal performance dependent so organizational knowledge management should be based on decision-making reducing time and costs, and increasing product/ service quality.

It is not easy performing training needs diagnosis because company competitiveness requires greater flexibility and jobs rotation. To achieve training success is necessary defining what are objectives being achieved and priorities. The need diagnosis requires much preparation. One way to define training needs is to achievc a simple open questionnaire and/ or informal interviews distributed to chiefs and subordinates. It is essential conducting diagnosis before training developing because desired objective not achieving risks.

The technique of self evaluation bases on activities identification related tasks that are developed in the organization, every competency held and training needed to fill gaps in knowledge or acquiring new skills. It is used a single sheet filled by each worker, after is delivered to their leaders to supplement with the work knowledge professionals do. A copy is sent to training service [8]. Beyond obvious utility for individual training needs diagnosis this instrument allows the leader assume more careful training participant selection and ensures greater access opportunity equality.

From identified needs and priorities established to meet strategic decisions terms and available resources, the training plan is prepared and will incorporate a set of measures to initial and continuing training, defined in specific objectives terms to be achieved and covered workers containing all elements relating to the training activities planning.

For training results evaluation, crucial phase of this integrated process, should be made self evaluation, to perform on job by trainees, three to six months after training, complemented by leaders opinion about the their work behaviour. This information feedback enable identifying gaps in learning or other issues that can be resolved immediately through follow-up sessions or cause further training.

3 Proposed Approach

I want to find how connect strategic processes with operational human resources.

Model framework development is the first phase that allows an organization embrace the universe of principles and technologies level usage associated with the knowledge management.

The data gathering in a real environment will provide a case study for the model validation and is at the same time part of the action research diagnostic phase at the same organization.

I have contacted a company which has already been briefed on the knowledge management principles and methodology.

The model [Fig. 1] can serve as a reference to be implemented in enterprises for sustained feed of the catalog creating and updating process, performance profile and work requirements, training and learning, distributed by the different parties, for this will be necessary, exploit and respond to change.

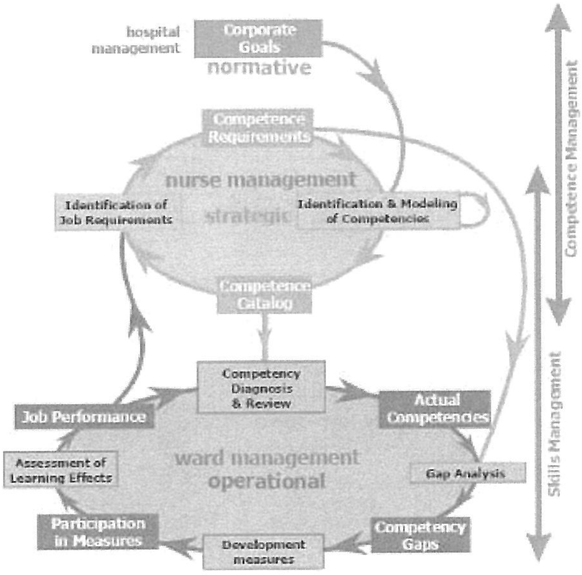


Fig. 1. Competences Management Levels (cp. [6])

The data gathering will be made either through documents provided by the company, either through interviews. These interviews serve also to remaining elements drawing of diagnosis, particularly on the relevance of possible needs for knowledge obtaining and sharing.

The interviews collecting data methodology is not yet fully defined. The first stage will probably be the concept map construction including the activities each individual is involved and their responses to a set of few questions regarding the use and knowledge needs. Two more phases are planned, a method of collection validation and another data collected confirmation, in order to take advantage of the team members evolving sensitivity development [5] to the issues of knowledge management.

The model instantiating the organization will plan future procedures of this action research together. The purpose is to assess knowledge and make training assumptions to prevent gaps and enhance employability.

Human resources competences development through tools allow companies, organizations and employees know their diagnoses and then conducting training operations to fill up gaps found [Fig. 2]. All parties in the process have to be prepared to different perspectives, operating at different levels of abstraction, different responsibilities and tasks. Technologically, the system must be prepared to e.g. follow organizational competencies side, the same way that takes into account the objectives of individual learning.

Introduction of competency management requires a mentality change in order to handle that will be made interactive workshops.

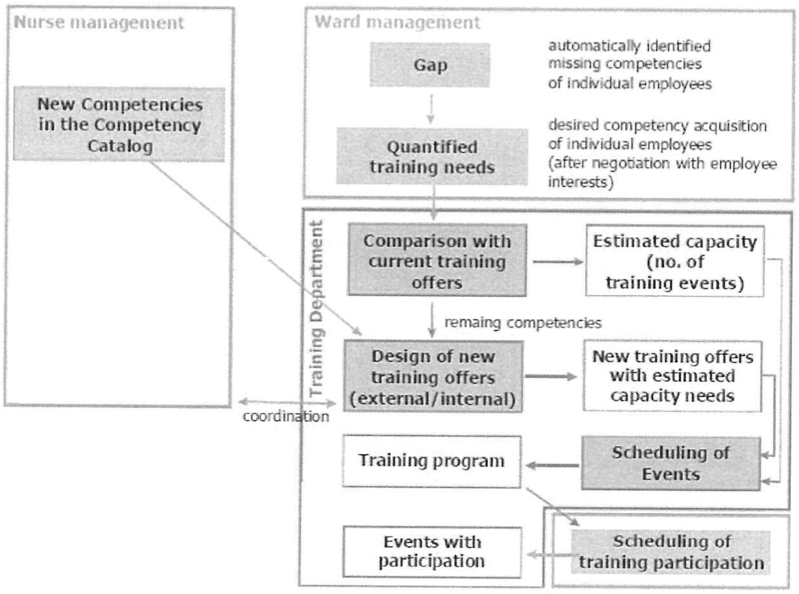

Nurse management
New Competencies in the Competency Catalog
Ward management
Gap
automatically identified missing competencies of individual employees
Quantified training needs
desired competency acquisition of individual employees (after negotiation with employee interests)
Training Department
Comparison with current training offers
Estimated capacity (no. of training events)
remaing competencies
Design of new training offers (external/internal)
New training offers with estimated capacity needs
coordination
Training program
Scheduling of Events
Events with participation
Scheduling of training participation

Fig. 2. Competency-based training needs planning (cp. [6])

4 Implementation Proposal

4.1 Competency Catalog Modelling

The catalogue of competences has to be completely modelled from scratch. The first step will be conducting semi-structured interviews. With this information, to interactively develop the catalogue, are conducted workshops series. The objective is to collect general and specific skills and the most of each department competencies which is difficult.

4.2 Action Research

An action research consists of a cycle that begins with a diagnosis and that follows the stages of action planning, action, evaluation and knowledge acquired description.

Importing the method principles, is set for a first cycle with goals from short term to gain confidence from the organization, with the results and from a second cycle of action research based on the findings of the previous cycle.

4.3 Technical Training Council

It is the creation of a council technical training, consisting of representatives of various professional groups, which will have advisory functions, in support of diagnosis, design and evaluation of the integrated process of training.

4.4 Requirements, Existing Competencies and Training Needs Analysis

The training is not done in a vacuum much less an organization. It involves costs and is naturally expected gains of any nature. When it found a gap, the need to have response training, was triggered by a training needs diagnosis process in order to set up action that should solve. Training should lead to an adjustment of professional performance set out by the organization expected objectives [7].

At this context training needs are normally associated with cases of jobs discrepancy detected between the desired performance and performance verified. The individual profiles can be between 5 and 30 skills.

4.5 Competences self-diagnosis

When professionals are familiarized to think on qualifications and diplomas, it is difficult to think of competences.

By accessing self-diagnoses employees and businesses have the opportunity to test their competences on an autonomous basis with immediate availability of its associated benchmarking and indication of their output profile, the gaps in respect to the different types of technical competences and behavioral, and so on...

There are two self-diagnoses, one for business and one for the employees. In the first case, company evaluates its position in dimensions as processes, people, technology, knowledge management and strategy. Furthermore people can assess their cognitive skills, technological and generic literacy, specific technology skills, social skills, motivation and knowledge [8].

Self-diagnosis will be done by employees who operate in areas as different as services, trade, industry, tourism, agriculture and transport and achieve tasks of management, technical, administrative and operational.

4.6 Analysis process stages

The most appropriate way on thinking about training needs analysis is embedding it in the context of a strategy for quality improvement in an organization or community. It aims not only detect occasional problems, but provide relevant information to act in a strategic way. This process of collection, selection, processing and interpretation of data includes three levels of analysis that coincide with its three basic phases.

4.6.1 Professional Field Analysis

In this phase may take place three types of analyses leading to the training needs identification.

Organizational level analysis

Seek first to examine training relevance, which follows the analysis of when and where training can and should be applied. It involves among other things, available resources analysis, organization specific conditions, technical system, work relationships, and so on.

Functional level analysis

Seeking to determine the most cost effective way of implementing a function or group of functions, the conditions and necessary equipment, as well as the knowledge and competences required. Nowadays has been given increasing importance to the competences and other than technical knowledge that is crucial to give an effective performance of diverse professional functions. In the past they tended to be in second pane or even omitted due to a technical perspective as well as that analysis was performed.

Personal level analysis

Assess professional performance as well as the actions and conditions necessary to achieve on-job required level.

4.6.2 Training Field Analysis

It deals with "translate" training content, routes and methods resulting of the analysis above.

4.6.3 Pedagogical Field Analysis

At this stage when combined with the previous seeks to define the specific conditions of pedagogical activities to develop pedagogical theories, evaluation tools, pedagogical techniques, time distribution and space, material resources and logistics, and so on.

4.7 Techniques

The methods and techniques selection of needs analysis is not merely a mechanical process. It depends on considered real situation type, but also the past experience. In the professional field for the information collection, e. g., use of techniques such as: interviews, brainstorming, surveys and questionnaires, direct observation. In the information collected analysis, resort to techniques such as content analysis, functions analysis, capabilities, competencies and others.

4.8 Training Methods

The analysis can be done with the aim of bridging the need for one of the following methods of training.

4.8.1 Initial Training

Training is qualification need of workers to a given occupation performance, as a result of the new jobs creation, replacement of employment, organisational change or other similar situations.

4.8.2 Continuous or Improvement Training

Training is based on found deficiencies in the activity performance, taking into account the level of results and attitudes expected. The situations that may cause these needs are diverse, such as: productivity under expected results; technological or organizational changes that involve adjustments in the production system and the qualifications of workers.

4.8.3 Conversion Training

Today, this type of training is widespread and comes from the extinction of certain occupations, such as the introduction of new technologies that changes so deep the contents of an occupation; the extinction of certain economic activities which show little profitable in the face of international competition, and so on. These and many other situations cause the continuous vocational retraining of workers to completely different activities.

4.9 Training evaluation

This assessment is seeking various types of results: the differences between the objectives and achievements, the effects of training in jobs; the factors which have influenced no so good results, such as: inadequate means and methods; deficiencies attributable to trainers and to logistical conditions in which the action took place, inappropriate selection of the trainees, and so on. It is important to accept that training evaluation is not only to the detection of differences between the objectives and expected results, but it is above all a reflection process on the training device itself.

4.10 Competencies Evaluation

There is no necessary correspondence between the knowledge acquisition and mobilize or apply capacity in the context of work.

The depression of school certificates in many areas, by an effect of massification, which has resulted in demand by employers, workers with more competencies than school credentials.

The workers mobility also meant the revitalization of professional competencies, demonstrated in terms of curriculum pathways, a factor which has been highlighting the employment acquired competencies, especially in countries where the majority of the population has reached a high level of education. The problem is even more complex. It is not simply to competences assess, but to know how this assessment should also translate in terms of promotion and ranking in the workplace.

5. Technology

In the context of a service oriented architectures is permitted: (1) combine employees requirement and competencies, (2) add requirements, experience, performance and quality management, (3) be present objectives that influence key competences, (4) compare the requirements to perform certain work, to find the gaps of competence, (5) select training measures which take into account the competence gaps, (6) evaluate during/ after attending the training action the learning effects, (7) increase the employee competencies profile when appropriate [4]; enabling better business processes integration and operation with IT services.

The competence models and hierarchies can be contextually integrated with LMS/ TMS, and these applications with the knowledge management [9].

6. Conclusions and Future Work

This paper presented an approach to the competences management through an analysis of some of the work related to this issue.

The formulation presented for the problem, despite being vague serves as a basis for future investigations, it is intended that various entities and workers can test, before, during and after, their competence/ skills through a simulator, and referred to standard courses eLearning to be made. After development and implementation of on-line tool, this will be available on the Internet.

The work plan includes three months for the consolidation of state-of-the-art, three months to complete the first cycle of action research, twelve months for planning and intervention of actions that decide to adopt, and finally, six months of writing for the completion of the final document.

References

1. C. K. Prahalad, G. Hamel: The core competence of the corporation. Harv. Bus. Rev., pp. 79--91 (1990)
2. J. B. Barney: Firm resources and sustained competitive advantage. J. Manage., vol. 17, no. 1, pp. 99--120 (1991)
3. B. Wernerfelt: A resource-based view of the firm. Strat. Manage. J., vol. 5, pp. 171--180 (1984)
4. Andreas Schmidt, Christine Kunzmann: Sustainable Competency-Oriented Human Resource Development with Ontology-Based Competency Catalogs. Proceedings of E-Challenges (2007)
5. G. Berio, M. Harzallah: Knowledge Management for Competence Management. Proceedings of E-Challenges (2007)
6. Christine Kunzmann, Andreas Schmidt: Ontology-based Competence Management for Healthcare Training Planning – A Case Study. 6th International Conference on Knowledge Management (I-KNOW) (2006)
7. Ley T., Ulbrich A.: Achieving benefits through integrating eLearning and Strategic Knowledge Management. Proceedings of the 5th International Workshop for Interactive Computer Aided Learning (ICL) (2002)
8. Berio G., Harzallah M.: Towards an integrating architecture for competence management. Computers in Industry (2007)
9. Harzallah M., F. Vernadat, IT-Based competency modeling and management: from theory to practice in enterprise engineering and operations. Computers in Industry, Vol. 48, No. 2, pp. 157-179 (2002)

2. AUTHORS IN ALPHABETICAL ORDER

Alves, Paulo 112

Azevedo, Alexandre 27

Barbosa, Fernando Sérgio 134

Correia, Filipe Figueiredo 100

Ferreira, Hugo Sereno 100

Garcia, Jorge Esparteiro 123

Lopes, Carla Teixeira 52

Martins, Nuno 15

Miranda, José Carlos 15

Moreira, José Joaquim 76

Mota, Jorge 158

Pereira, António 64

Pereira, Carla Sofia 88

Reis, Luís Paulo 64

Rocha, Válter 39

Teodósio, Fernando 170

Valente, Pedro 64

Verde, Isidro Vila 146

3. PAPERS IN ALPHABETICAL ORDER

A New Grouping Method for XSL 1.0 146

A System for Automatic Construction of Exam Timetable Using Genetic Algorithms 76

An Approach to Improve Speed and Objectivity in Audits 112

Aspect-Oriented Web Development in PHP 123

Calibration Agent for Ecological Simulation: A Metaheuristic approach 64

Collaborative Ontology Specification 88

Comparing Three Aspect Mining Techniques 134

Competency Management Articulation for Human Resources 170

Evaluation and Comparison of Automatic Methods to Identify Health Queries 52

Métodos de Iluminação Global Baseados em Fotões 27

Sensing the World: Challenges on WSNs 39

Trends on Adaptive Object Models Research 100

Using Learning Styles and Neural Networks as an Approach to eLearning Content and Layout Adaptation 158

X3D and Java Fusion in a Medieval Fantasy Game 15