



coopmedia 2003

Workshop de Sistemas de Informação Multimédia, Cooperativos e Distribuídos

Actas

ACTAS



ISEP, Porto 8 de Outubro 2003

Organização:



Centro de Computação Gráfica

Co-organização:



Universidade do Minho/DSI



Grupo Português de Computação Gráfica

Patrocinadores:



Fundação para a Ciência e a Tecnologia

MINISTÉRIO DA CIÊNCIA E DO ENSINO SUPERIOR



Apoio do Programa Operacional Ciência,
Tecnologia, Inovação do Quadro Comunitário de Apoio III

Caixa Geral de Depósitos



Apoios:



FACULDADE DE CIÊNCIAS UNIVERSIDADE DE LISBOA



Prefácio

O 2º Workshop de Sistemas de Informação Multimédia, Cooperativos e Distribuídos – Coop-Media 2003, tem lugar este ano no ISEP – Instituto Superior de Engenharia do Porto, no dia 8 de Outubro, integrando as actividades do 12º Encontro Português de Computação Gráfica, que decorre entre 9 e 10 de Outubro.

O Coop-Media 2003 vem no seguimento do primeiro encontro realizado em 30 de Junho de 2000, em Coimbra, e visa fomentar em Portugal o contacto entre investigadores, docentes e profissionais, que realizam trabalho ou utilizam os sistemas de informação multimédia, cooperativos e distribuídos. O objectivo central do Coop-Media 2003 é assim divulgar experiências inovadoras a nível nacional na investigação, desenvolvimento, ensino e aplicação das tecnologias envolvidas.

Os sistemas de informação multimédia, cooperativos e distribuídos representam actualmente uma importante fatia dos sistemas de informação em geral, assumindo um papel central sobretudo na área das aplicações finais onde o grau de interactividade é elevado, não só entre o utilizador e o computador (máquina) mas também entre os utilizadores entre si, através do e com o computador.

A integração da tecnologia multimédia nos sistemas de informação permite promover a inteligibilidade da informação que estes transmitem ou cujo acesso ou manipulação facilitam, explorando a expressividade dos conteúdos informativos através dos sentidos humanos, meio pelo qual o utilizador humano apreende a realidade que o rodeia. A dimensão da cooperação e distribuição alarga, por si, o horizonte dos sistemas de informação no sentido de incluir os outros utilizadores, eventualmente dispersos e distantes geograficamente, na partilha de um espaço informacional e comunicacional comum.

Os trabalhos do Coop-Media 2003 decorrem ao longo de um dia, divididos entre uma palestra proferida por um orador convidado e três sessões técnicas. O orador convidado é João Álvaro Carvalho, Professor Catedrático no Dep. de Sistemas de Informação da Universidade do Minho, fundador da Associação Portuguesa de Sistemas de Informação e um dos pioneiros em Portugal da área científica e tecnológica dos sistemas de informação, que apresenta a palestra com o título “Trabalho, Organização e Cooperação: Da exo-organização à endo-organização”. As sessões técnicas estão organizadas da seguinte forma: Sistemas e Modelos de Cooperação, Multimédia e Interação, Tecnologia e Aplicações.

A comissão de programa avaliou 16 comunicações, das quais foram seleccionadas 10 para apresentação no Workshop. Em face da qualidade das comunicações seleccionadas e do interesse que o evento despertou na comunidade científica e tecnológica destas áreas, pensamos que mais uma vez valeu a pena o esforço de levar por diante uma nova edição do Coop-Media – Workshop de Sistemas de Informação Multimédia, Cooperativos e Distribuídos, sendo possível oferecer aos seus participantes, um programa de grande interesse e qualidade técnico-científica.

O Coop-Media 2003 - 2º Workshop de Sistemas de Informação Multimédia, Cooperativos e Distribuídos só foi possível através do envolvimento generoso e voluntário de uma equipa de trabalho, proveniente de várias instituições, para quem é devido um primeiro agradecimento sincero e reconhecido.

Os Presidentes da Comissão Organizadora e da Comissão de Programa gostariam de agradecer:

Aos membros da Comissão de Programa pelo empenhamento colocado no rigor das avaliações e na selecção final das comunicações submetidas. Aos membros da Comissão Organizadora, em particular à Mourylise Heymer, Nuno Ferreira, Alexandre Oliveira e Leonel Valbom pelo grande empenho no desenho e implementação do site Web, de todo o material de divulgação e pelo apoio logístico. À Marta Meira pelo trabalho árduo do secretariado e da compilação e formatação das actas do workshop. Ao conferencista convidado, João Álvaro Carvalho, a disponibilidade para participar e contribuir para o sucesso do Coop-Media 2003, partilhando a sua experiência, saber e visão. A todos os autores pelo interesse, esforço e empenho colocados na elaboração das suas comunicações. Um agradecimento especial é aqui devido à Fundação para a Ciência e a Tecnologia e aos patrocinadores Caixa Geral de Depósitos e Visiona TI, Lda. pelos apoios concedidos. E finalmente ao Dep. de Sistemas de Informação da Universidade do Minho, co-organizador com o CCG e o GPCG, deste evento, para além do ISEP que cedeu as suas instalações, e pelo acolhimento e apoio dado a esta realização.

Porto, Outubro de 2003

Adérito Marcos
Pedro Antunes
Nuno Guimarães

COMISSÃO ORGANIZADORA

Presidente: Adérito Fernandes Marcos

Co-Presidente: Pedro Antunes

Secretariado: Marta Meira

Publicidade: Mourylise Heymer

Patrocínios: Alexandre Oliveira

Preparação e Impressão de Actas: Marta Meira, Adérito Fernandes Marcos

Site Web e Artes Gráficas: Mourylise Heymer, Nuno Ferreira, João Nunes,

Logística: Leonel Valbom, Tony Lavender

Organização local: António Costa, Miguel Leitão

COMISSÃO CIENTÍFICA

Presidente: Nuno Guimarães – DI/FC/UL

Adérito Fernandes Marcos - CCG, DSI/UM

Adriano Moreira - DSI/UM

Carlos J. Costa - ISCTE

Joaquim Jorge – IST/UTL

Jorge Louça – ISCTE

José Fernandes de Almeida – DSI/UM

Luís Carriço – DI/FC/UL

Luís Paulo Santos – DI/UM

Nuno Correia – DI/FCT/UNL

Pedro Antunes – DI/FC/UL

Programa

Sessão Convidada

<i>Trabalho, Organização e Cooperação: da exo-organização à endo-organização</i>	
João Álvaro Carvalho.....	1

Sessão Técnica 1

Sistemas e Modelos de Cooperação

<i>“Modeling the Information Structures of Meetingware”</i>	
Pedro Antunes, Luis Carriço.....	2
<i>“P – Manager: Um Sistema de Apoio ao Gestor”</i>	
Manuela Aparício, J. Paulo Costa.....	17
<i>“Segurança dos Web Services no Comércio Electrónico Móvel”</i>	
Ricardo Martins, Jorge Rocha, Pedro Henriques.....	28

Sessão Técnica 2

Multimédia e Interacção

<i>“Reconhecimento Automático de Fala Contínua em Português Europeu recorrendo a Streams Audio-Visuais”</i>	
Filipe Sá, Pedro Afonso, Ricardo Ferreira, Vítor Pêra.....	37
<i>“Avaliação de aspectos de sincronização de Livros Falados Digitais”</i>	
Carlos Duarte, Luís Carriço, Hugo Simões, Teresa Chambel, Nuno Guimarães.....	48
<i>“Collaborative Multimodal Authoring of Virtual Worlds”</i>	
Vítor Sá, Filipe Marreiros, Adérito Marcos.....	61
<i>“Supporting Direct User Interventions in Exception Handling in Workflow Management Systems”</i>	
Hernâni Mourão, Pedro Antunes.....	68

Sessão Técnica 3

Tecnologia e Aplicações

<i>“Aplicações Móveis de valor acrescentado – um caso prático”</i>	
Alexandre Oliveira, Pedro M. Figueiredo, Adérito Marcos, Rui José.....	81
<i>“Utilizando uma Base de Dados XML Nativa aplicada ao tratamento de erros num Sistema de logs”</i>	
Giovana Mendes, Nuno Silva, Pedro Henriques.....	91
<i>“Geração Automática de Interfaces Web para Sistemas de Informação Metamorphosis”</i>	
José C. Ramalho, Giovani Libreloto, Pedro Henriques.....	103

SESSÃO CONVIDADA

Trabalho, Organização e Cooperação: Da exo-organização à endo-organização

João Álvaro Carvalho
Universidade do Minho
Departamento de Sistemas de Informação
Campus de Azurém, 4800-058 Guimarães
jac@dsi.uminho.pt

Resumo

A designação de sistemas cooperativos foi criada para referir um determinado tipo de aplicações das tecnologias da informação cuja finalidade é suportar a cooperação entre diversas pessoas num contexto de trabalho em organizações.

No entanto, qualquer trabalho realizado nas organizações implica a cooperação entre diversas pessoas desempenhando as tarefas mais diversas. Nalguns casos- sobretudo quando a cooperação passa pela partilha de informação e pela realização de trabalho conjunto sobre essa informação - essa cooperação é bem explícita enquanto que noutros casos ela pode passar despercebida.

Por outro lado, o modo como o trabalho é organizado tem vindo a mudar. De abordagens em que as tarefas a realizar estão previamente definidas por alguém com responsabilidade de organizar esse trabalho (exo-organização) temos vindo a assistir ao ganhar de importância de abordagens que se baseiam na iniciativa, autonomia e responsabilidade dos indivíduos (endo-organização).

Tendo em conta estes os dois aspectos acima referidos, fará sentido reflectir sobre os diferentes tipos de cooperação e sobre o tipo de suporte que as aplicações das tecnologias da informação poderão dar a essa cooperação.

Breve Biografia

Professor Catedrático do Departamento de Sistemas de Informação da Escola de Engenharia da Universidade do Minho.

Docente e investigador na Universidade do Minho desde 1983 onde tem vindo a leccionar disciplinas relacionadas com os fundamentos dos sistemas de informação e com o desenvolvimento de sistemas de informação, em cursos de licenciatura e de pós-graduação. Orientador de diversos estudantes de mestrado e de doutoramento em trabalhos de investigação que se enquadram nos seus interesses.

Director do Departamento de Sistemas de Informação. Coordenador da linha de Sistemas de Informação do Centro ALGORITMI, centro de investigação e desenvolvimento reconhecido pela Fundação para a Ciência e a Tecnologia.

Autor de diversas publicações na área dos sistemas de informação com especial ênfase em temas relacionados com o desenvolvimento de sistemas de informação, a utilização de sistemas informáticos em contextos organizacionais e os sistemas de informação e o conhecimento organizacional.

Membro da Comissão Instaladora e da Comissão Executiva da APSI (Associação Portuguesa de Sistemas de Informação). Co-editor da revista *Sistemas de Informação* da APSI.

SISTEMAS E MODELOS DE COOPERAÇÃO

Modeling the Information Structures of Meetingware

Pedro Antunes, Luís Carriço

LaSIGE – Laboratório de Sistemas Informáticos de Grande Escala,

Departamento de Informática,

Faculdade de Ciências da Universidade de Lisboa,

Bloco C5 - Piso 1 - Campo Grande, 1700 Lisboa, Portugal

paa@di.fc.ul.pt, lmc@di.fc.ul.pt

Abstract

Meetingware is a technology that has not yet taken a significant role in the organization. There are several reasons contributing to this situation, one of them being the lack of a generic model characterizing meetingware functionality and offering a clear path for organizational integration. This paper proposes a model characterizing in detail the different meetingware elements and corresponding functionality, so that designers and developers be able to easily incorporate them in current and future organizational software environments. The model identifies three fundamental components of meetingware: (1) roles, addressing the diversity of people and activities that we can observe in meetings; (2) resources, considering the meeting logistics as well as the group memory produced and managed by the meeting participants; and (3) process, addressing the organization of the set of activities that meeting participants must execute in order to accomplish the meeting goals.

Resumo

Meetingware é uma tecnologia que ainda não assumiu um papel de relevo nas organizações. Diversas razões contribuem para esta situação, sendo uma delas a falta de um modelo genérico que caracterize em detalhe a funcionalidade associada ao meetingware e ofereça um plano para a sua integração organizacional. Este artigo apresenta um modelo que caracteriza em detalhe os diferentes elementos do meetingware, assim como a funcionalidade correspondente, de modo a que designers e implementadores possam facilmente incorporar estas funcionalidades nos sistemas que utilizam e desenvolvem. O modelo identifica três componentes fundamentais do meetingware: (1) papéis, referenciando a diversidade de pessoas e actividades que se podem observar nas reuniões; (2) recursos, considerando a logística das reuniões, assim como a memória grupal produzida e gerida pelos participantes nas reuniões; e (3) processo, organizando o conjunto de actividades que os participantes numa reunião têm necessariamente que realizar de modo a atingirem os objectivos da reunião.

1. Introduction

Broadly defined, meetingware brings together people, hardware, software and roomware, with the purpose of supporting, managing, guiding and stimulating participation in meetings; improving at the same time the meeting processes and outcomes.

Unfortunately, meetingware is an intricate technology: sometimes distributed in time and space, linking many users with distinct abilities, supporting many different hardware and software configurations, supporting different types of groups, tasks and functionality, many times forcing people to plan in advance the system use, and other times specifically requiring experts to configure, manage and use the technology.

If we want to make significant advances in the diffusion and assimilation of this technology throughout the organization, we have to characterize in detail the different meetingware elements and corresponding functionality, so that designers and developers be able to follow a clear roadmap to easily incorporate them in current and future organizational software environments.

In this paper we analyze the different elements that make up meetingware, highlighting the relevant properties of each element and identifying the relevant relationships between them. The results from this research are organized in a model with three major elements: roles, process and resources.

Our contributions to the state of the art are the following:

- The proposed model results from looking into many meetingware systems. Although not complete, the model is comprehensive, in particular to what concerns the types of information managed in meetings and the types of information exchanged between meetings and the organization.
- We clarify and organize meetingware functionality, addressing the issue of complexity at the conceptual level. One of the endeavors of modeling is to provide simple, yet detailed descriptions of complex behavior, and we believe that the proposed model simplifies the task of designers and developers integrating meetingware functionality with other organizational systems.
- The proposed model facilitates the evaluation of different meetingware technology offered by the research community and software vendors. A definition of the fundamental components that make up meetingware allows making comparisons and permits organizations that wish to acquire this type of technology to confront the technology with their needs.

The paper is organized as follows. First, we present some related work. The following sections are dedicated to characterize the three components mentioned above. Next, we illustrate how the proposed baseline can be used to compare the functionality of different meetingware. Finally, we present some issues for discussion and conclusions about this work.

2. Related Work

Several authors have identified relevant elements in meetingware. Hoffer & Valacich (1991) identify the elements of organizational memory associated to meetingware. Aiken, et al. (1991) identify a collection of fundamental meetingware components including database, model base, interface, network, facilitator and user. Bui & Jarke (1986) characterize the functionality of the communication manager. Jacob & Pirkul (1992) define a framework specifically focusing on group decision making, with three major systems: language, group problem processing and group knowledge. Silver (1991) proposes a taxonomical view with three major different perspectives that are of interest to designers and users: interface, functionality and holistic attributes. Zigurs & Buckland (1998) offer a definition of task and overview different classifications of tasks in a group context. Rao & Jarvanpaa (1991) present a categorization of meetingware technology in three different features: support to improved communication, support to increased participation and computational support for tasks (information processing). Nunamaker, et al. (1991a) characterized four major mechanisms in meetingware: process support, task support, task structure and process structure. DeSantis & Gallupe (1987) categorized meetingware technology in three levels of complexity: level 1 systems facilitate information exchange among participants; level 2 systems provide decision modeling and group decision techniques; and level 3 systems support machine-induced group communication patterns and can include expert advice. Kraemer & King (1988) characterize meetingware as socio-technical packages, comprising hardware, software, organizationware and people.

We have not yet seen a proposal to specify an integrated model with these elements. Yet, these types of integrated models have been developed in other fields. For instance, the Dexter Hypertext Reference Model (Halasz & Schwartz 1994) defines a set of components that capture the relevant abstractions for hypertext systems. The Dexter model serves as a basis for terminology, for comparing different hypertext system designs and even for assessing conformance with the reference model. The goals behind the Dexter model motivated our intentions to develop a meetingware model.

The proposed integrated model also results from the analysis of the following meetingware systems and tools: GROUPSYSTEMS (Dennis, et al. 1988), SODA/DECISION EXPLORER (Eden 1989), MEETINGWORKS (Lewis 1987), EXPERT SYSTEM PLANNER (Bostrom, et al. 1990), IDEA CONSOLIDATOR (Aiken & Carlisle 1992), AUTOMATED FACILITATOR AGENT (Aiken & Vanjani 1998), LOGANWEB (Raikundalia & Rees 1995b; Raikundalia & Rees 1995a; Raikundalia & Rees 1996), CIRE (Romano, et al. 1999), D-PLAN (Antunes & Ho 1999; Costa, et al. 1999), JOBBER (Kazman, et al. 1996).

3. Meetingware and Organizational Systems

Considering an information systems view, we may regard organizations consisting of multiple systems that structure and accomplish work using different forms of communication and coordination mechanisms (Malone & Crowston 1994). These mechanisms include workflows, plans, direct supervision, rules and procedures, training sessions and, inevitably, meetings (Figure 1). All these different mechanisms must interact with each other in order to support organizational goals, but a characterization of such interactions is outside the scope of meetingware since it depends on many organizational factors such as culture, rules, objectives, formal and informal structures, institutionalized practices, market climate, etc (e.g. (Mintzberg 1979)).

Fundamentally, our intention in separating the meetingware and organizational systems is to make a clear separation of what pertains to the meeting from the factors related to the organizational system.

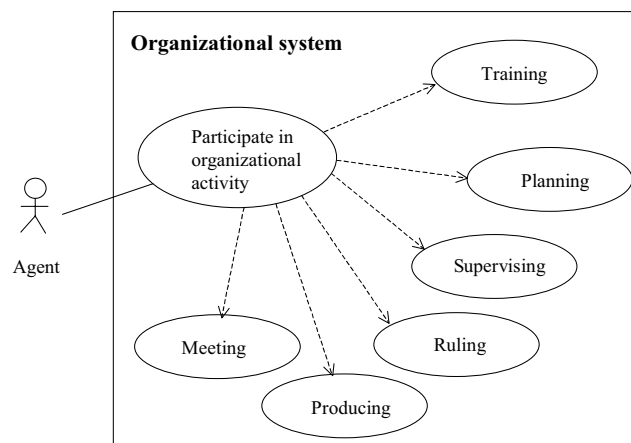


Fig. 1. Meetings as part of the organizational system

4. Meeting roles

Groups are made of individuals who assume interdependent roles according to the organizational expectations, meeting objectives and individual personality (Schwarz 1994).

Considering these different perspectives, it is natural that meeting participants may assume more than one role. Roles may be formally specified by the organization, result from institutionalized practices, negotiated and agreed before the meeting or assumed during meetings.

The following different roles are relevant in meetings: sponsor, facilitator, participant, secretary, observer and organizational agent. Three of these roles are active during the meeting session (facilitator, participants and secretary), while the others are only passively involved in the meeting session. Next, we will describe these roles in more detail.

Participant. The participants intervene in a meeting, producing and sharing various types of information such as ideas and comments. The participants have two attributes relevant within the context of the meeting: qualification and status (others, like personal attitudes are beyond our scope). The number of participants is something that contributes to characterize meetings in different genres, such as task forces, assemblies, commissions or committees, and so should be an attribute to consider when characterizing meetingware.

Facilitator. The facilitator is a neutral role, accepted by the meeting participants, that carries a vast set of facilitative functions (Bostrom, et al. 1993; Clawson, et al. 1993): (1) Promoting a sense of belonging and responsibility; (2) Demonstrating self-awareness/expression; (3) Selecting and preparing the technology; (4) Hearing, clarifying and integrating information; (5) Developing and asking the right questions; (6) Maintaining the group targeted in the results; (7) Creating comfort with technology; (8) Creating an open and positive atmosphere; (9) Building harmony and relationships; (10) Presenting information to the group; (11) Demonstrating flexibility; (12) Planning and developing meetings; (13) Managing conflicts and negative emotions; (14) Understanding technology and its capacities; (15) Encouraging and supporting multiple perspectives; and (16) Directing and managing the meeting.

The manner in which this role delivers its facilitative support to the meeting can be classified in the following categories: user driven (UD), when the facilitative functions are available to all meeting participants and thus there is no person specifically assigned to this role; facilitator-driven (FD), when there is one person designated to assume this role; and chauffeur-driven (CD), when the person designated to assume the facilitation role only manages the technology but not the process (Dickson, et al. 1993).

The role of the facilitator is fundamental to the meeting process, and may contribute in a decisive way to its success (Nunamaker, et al. 1997; Clawson, et al. 1993; Jay 1976). Thus, it must be explicitly addressed by meetingware.

Secretary. The role of the secretary is to take notes and produce a meeting report (The 3M Meeting Management Team 1994). One fundamental aspect to consider in this role is the type of technology available to take notes and produce reports, since this functionality may be centralized (C) or carried out by a group of people (distributed, D). This functionality may also be user-intensive (UI) or automated (A) by the technology (Aiken, et al. 1991; Aiken & Vanjani 1998).

Sponsor. The sponsor has a fundamental, although sometimes neglected, passive role in a meeting. The sponsor is the “owner” of the meeting and, ultimately: (1) Is the repository of the meeting objectives; (2) Defines and clarifies the meeting objectives; (3) Approves the meeting agenda, set up by or in collaboration with the facilitator; (4) Reviews the meeting outcomes; (5) Provides an interface between the organization and the meeting.

Some literature also mentions the role of the meeting leader (The 3M Meeting Management Team 1994). The leading role is a combination (in one person) of the sponsor and facilitator roles and thus will not be considered by us.

Observer. The observer is a passive role dedicated to become aware or infer about actions, interactions and patterns of behavior in meetings (The 3M Meeting Management Team 1994).

Organizational agent. The organizational agent is a role that, although being passive, produces information necessary to the meeting and is affected by the meeting outcomes. The type of information managed by the organizational agent is relevant to define the interface between the organizational and meeting systems.

5. Meeting resources

Under this category we consider two fundamental components of meetingware (Figure 2): logistics and group memory.

Logistics

The meeting logistics includes a description of generic meeting facilities, such as physical rooms, tables, chairs, computers, networks and roomware (Streitz, et al. 1997) (e.g. liveboard (Wagner, et al. 1993)). The different room arrangements can be categorized in: office stations, meeting rooms, laboratories and conference rooms.

One intrinsic characteristic of meetings, which is related to logistics, is the definition of time and place of the meeting. This classification affords defining the following meeting settings: STSP (same time/same place), STDP (same time/different place) and DTDP (different time/different place) (Beise, et al. 1992). One more category has been added to this typology, designated ATAP (any time/any place) to classify situations that cannot clearly be confined in one of the other categories.

Another characteristic of meetings related to logistics is the communication mode. In the absence of technology, it will be Face-to-Face (FtF). However, meetingware supports other communication modes (Fjermestad & Hiltz 1999):

- Group Support System (GSS) – This situation uses software tools that structure communication and assist group decision (such as voting tools);
- Computer Mediated Communication (CMC) – This type of technology primarily provides support to group discussions through messages exchange (such as chat systems), although other types of support may be provided as well;
- Decision Support System (DSS) – A Decision Support System is focused on a person. It comprises single-user software and a single computer shared in a FtF setting. This technology may also be designated chauffeured technology (Clawson, et al. 1993).

The combination of different roomware technologies has been considered influential to assert group productivity. Thus, it is necessary to characterize roomware in detail. Roomware may be classified in (Streitz, et al. 1997): individual workstations in a network (WS); a liveboard configuration, with one single computer (LB); and a combined situation, with individual workstations plus a liveboard connected in a network (WS+LB).

Finally, one should also consider the level of support provided by the technology. This characteristic was defined by DeSantis & Gallupe (1987) in three levels summarized in the section 2 of this paper (1 – facilitating information exchange; 2 – decision modeling and

group decision techniques; 3 – machine induced activities). To this classification, Fjermestad & Hiltz (1998) added a level 0, which considers some types of technology that offer very low group support. For instance, chat systems that show a few lines of text exchanged between users are classified in level 0.

Group memory

Group memory concerns the shared information resources that a group uses to accomplish work. The major attributes of group memory items are (Orlikowski & Yates 1998): purpose; contents; media used; who is involved in producing the item; when was the item produced and where should the item be produced or used.

Regarding meetings, we shall take into account the agendas, meeting reports and support documents (Costa, et al. 2001). The agenda is considered a critical element to manage meetings successfully since meetings tend to crystallize their actions around it (Niederman & Volkema 1996). The agenda may have two different types of information: the list of topics or goals that the group must deal with; and the series of steps that the group should execute in order to accomplish their goals. Agendas including a list of steps are very rare and, in fact, meetings are frequently based on no agenda at all (Romano & Nunamaker 2001).

The reports are the visible outcome of meetings. The most common form is the meeting minutes, but other types of documents may be produced as well, such as action plans (Costa, et al. 2002). The meeting reports are characterized by the structure and format of the aggregation of items in a report. These items present a certain content that results from the participants' interactions and is linked to the meeting agenda.

In what concerns technology use, one should consider four different report formats:

- Automatically generated transcripts of information exchanged in meetings (e.g. persistent conversation);
- Automatically generated summaries, such as voting results;
- Meeting data formatted to support visualization by the participants, secretary, sponsor and organizational agents (Nunamaker, et al. 1991b; Raikundalia & Rees 1995b; Raikundalia & Rees 1995a);
- A collection of group memory components typically generated during meetings, such as dictionaries or formal definition lists. These components support complex group memory management. For instance, a browsing tool allows the users to move through the meeting memory, enlarging in a specific area to obtain details, or zooming out to have a high level vision of meeting data (Nunamaker, et al. 1991b).

In what concerns documents used to support the meeting, they can be essentially of three types:

- Base Documents – The documents that will be affected by decisions taken in meetings, for example a management report submitted for approval;
- Support documents – The documents used to directly support the decisions taken in meetings. For example, in an investment decision, a viability study would be a possible support document;
- Context documents – The documents necessary to characterize or explain the meeting process. For instance, some meetings, such as parliamentary meetings, follow a regiment, and that regiment is necessary to explain the meeting process.

6. Meeting Process

The meeting process structures the set of activities that the participants must execute in order to achieve some common goal.

The nature of activities changes as the participants move forward towards the goal. To reflect this, we typify activities accordingly to that progression and consider the following levels of detail:

- The meeting as a whole;
- The partition of the meeting process in several activities;
- The decomposition of the meeting process in several activities and sub-activities;
- The fragmentation of the meeting in an intricate collection of elementary activities, such as individual interventions or comments.

The first level characterizes the meeting process while maintaining the perspective of the whole. One example is given by the genre approach (Orlikowski & Yates 1994). The genre approach regards a meeting as a pattern of recurrent communicative actions including logistics, agenda, the meeting itself and the meeting report. This approach allows typifying and characterizing meetings like briefings, progress report meetings, staff meetings and management meetings in terms of purposes and communication patterns (Costa & Antunes 2001; Antunes, et al. 2001).

The second and third levels focus on the decision structure. The partitioning approach follows a logical view over decision making that is recurrent in literature (Gonçalves & Antunes 2000; Ho & Antunes 1999). According to this view, the goal is divided in several partial goals that can be accomplished in a systematic way. Planning is a good example of such a systematic approach, where the group has to identify what actions should be taken, by whom, when, what resources are needed and how can success be measured.

The decomposition approach regards meeting processes as decomposable in multiple levels of detail, with goals and sub-goals. For instance, Kaner (1996) proposed a generic pattern consisting of divergent, groan, convergent and closure phases. Each phase was then subdivided in several sub-phases (e.g., the divergent phase was subdivided in surveying the territory, searching for alternatives and raising difficult issues). Another example is this nine-step model proposed by Schwarz (1994): define the problem; establish criteria for evaluating solutions; identify root causes; generate alternative solutions; evaluate alternative solutions; select the best solution; develop an action plan; implement the action plan; and evaluate outcomes and the process. Briggs & Vreede (2001) follow a similar approach, identifying several basic patterns like diverge, converge, organize, elaborate, abstract and evaluate.

Finally, the meeting process can also be characterized in a very fragmented way, according to the flows of individual interventions produced by the participants and facilitator. These interventions can be categorized in process and content interventions (Miranda & Bostrom 1999). Defining the agenda, inviting the participants, opening and closing the meeting sessions, or tracking the agenda, are a few examples of process interventions. An example of the characterization of the meeting process as a collection of content interventions is given by the IBIS argumentative structure (Conklin & Begeman 1988).

The other attributes to consider in process or content interventions are:

- Contents – Corresponds to what is really transmitted to the group; Time – An attribute of great importance to characterize an intervention is the moment when it is produced. Based on this attribute, we can identify a set of characteristics of group communication. On the one hand there is the technical aspect of synchronous and

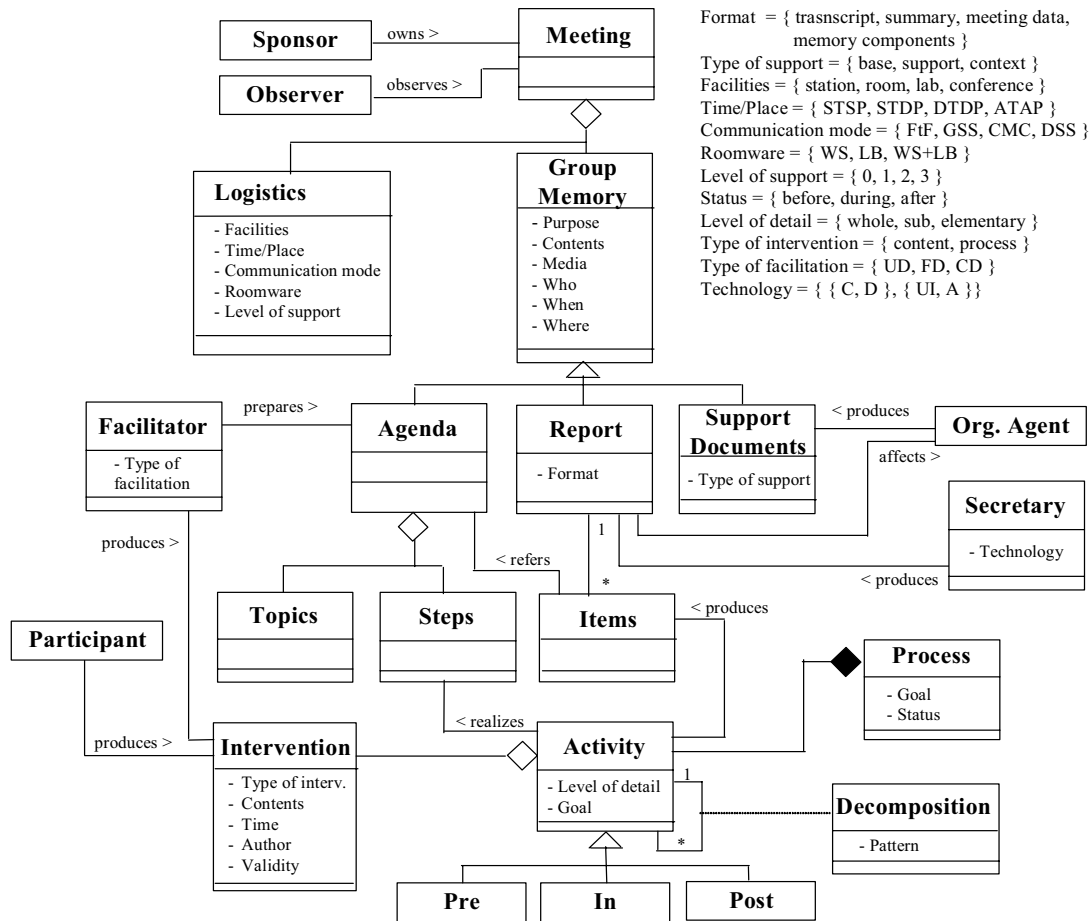


Fig. 2. Meetingware relevant elements

asynchronous communication. On the other hand, associated to the time stamp, there exists the possibility of parallel interventions. This possibility contributes to improve some group activities, like idea generation (Gallupe, et al. 1991).

- **Author** – The person that produces an intervention may be identified or not. This factor can have an important role in the process results. Several researchers have reported the effects of anonymity in the interaction process (e.g. (Connolly, et al. 1990)).
- **Validity** –The validity corresponds to the time during which the intervention can be accessible. The validity has repercussions on the organizational memory.

Taking a different approach, the set of process activities can also be classified according to what is designated as the meeting lifecycle. A detailed analysis of meetings allows verifying that the meeting lifecycle consists of three stages: (1) the pre-meeting stage, considering activities that have to be executed before the meeting; (2) the in-meeting stage, considering activities accomplished during the meeting; and (3) the post-meeting stage, considering activities that may be required afterwards. In the pre-meeting stage we include the meeting proposal, approval, planning (including definition of topics, goals and selection of participants) and invitation (Antunes & Ho 2001). In the in-meeting stage we find content interventions and process interventions.

In the post-meeting stage we should consider meeting assessment, report production and distribution, and progress review.

Note that the proposed baseline is generic and avoids adopting any particular policies to specifying meeting phases, activities or tasks. Ultimately, the meeting process may just consist of several interventions.

7. Examples

In Figures 3-5 we provide some examples of common uses for the model.

The first example considers that decisions are structured according to the rational approach defined by Simon (1997). In this approach there exists a goal and a wish to maximize some utility function, and there is some systematic way to accomplish the task (bounded by considerations of time and cost). Simon (1997) proposed the following three phases for the decision-making process: listing all alternative strategies (intelligence); determination of all consequences that follow upon each strategy (design); and comparative evaluation (choice). As shown in Figure 3, we have decomposed the process activities in the intelligence, design and choice phases, and defined the agenda accordingly. The report is limited to transcribe the adopted decision.

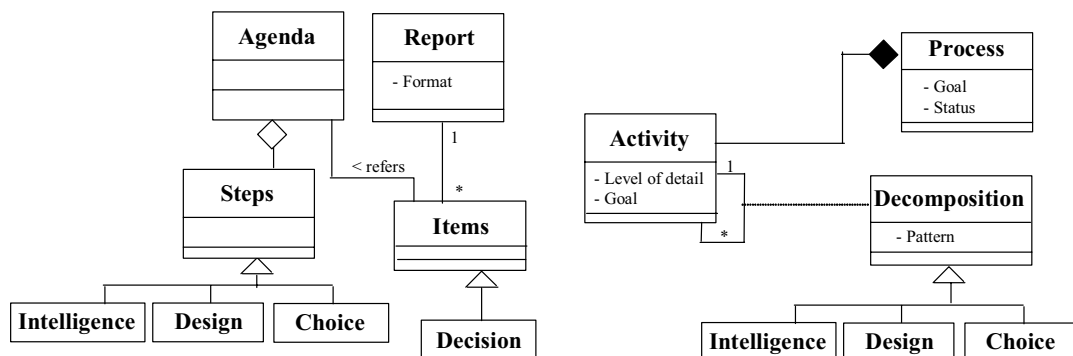


Fig. 3. Rational meeting according to Simon (1997)

In Figure 4 we show a more complex approach to decision-making supported by a tool implemented by Antunes & Ho (2001). The tool uses the Kaner's (1996) approach to structure the decision process, decomposing it in divergent, groan, convergent and closure phases. Note also that the mentioned tool also supports several pre-meeting activities.

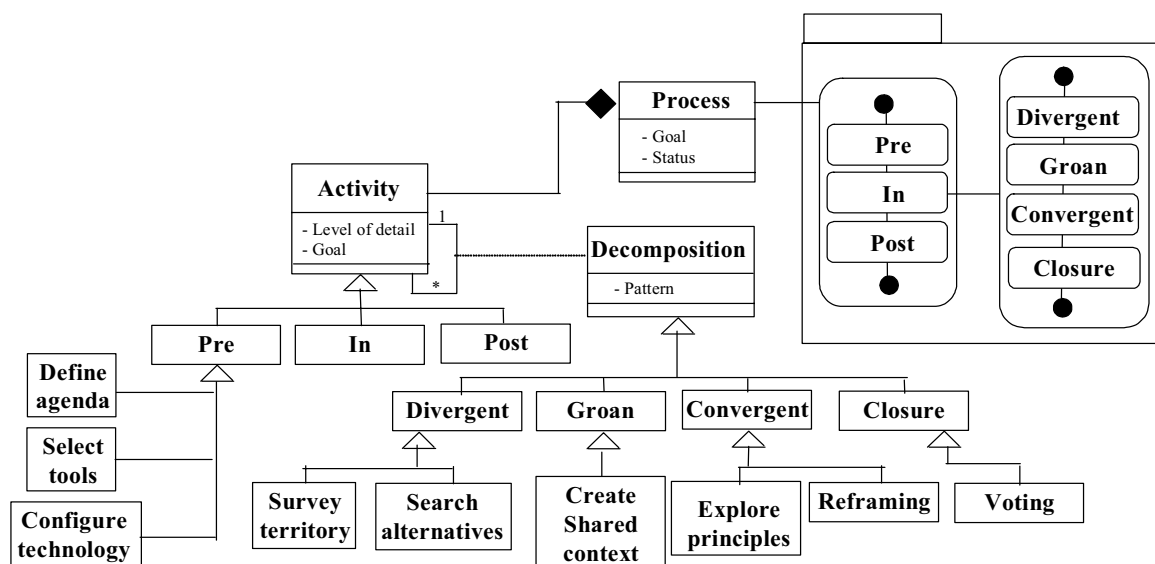


Fig. 4. Rational meeting according to Antunes & Ho (2001)

In Figure 5 we show an example of a meetingware that bases its implementation of the meeting process on the IBIS (Conklin & Begeman 1988) model. IBIS (Issue-Based Information System) was developed to provide a simple yet formal structure for the discussion and exploration of wicked problems. Wicked problems do not yield to the rational approach to problem solving. With a wicked problem, the understanding of the problem is evolving along with the work on a solution. The IBIS model structures the discussion of the

problem in accordance with three major data elements: issues, positions and arguments. These elements are linked together, so that one issue may generate other issues and several positions (against or in favor) supported by arguments.

Therefore the model categorizes the participants' interventions according to these elements. The produced report consists in the generated web of issues, positions, arguments and links, since no definite decision is obtained with this process.

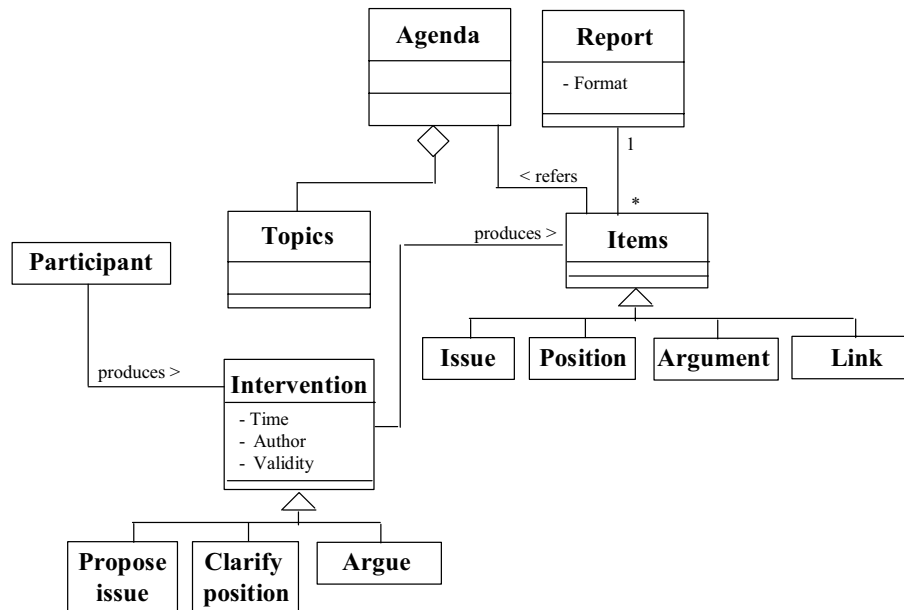


Fig. 5. IBIS (Conklin & Begeman 1988) meeting

8. Discussion

The proposed model identifies three major meetingware components: roles, meeting process and resources. The first component allows specifying the types of roles involved in meetings. The level of detail considered is the necessary one to clarify who interacts with the meetingware and what functionality the meetingware is expected to deliver to users. The model highlights two commonly neglected roles: sponsor and organizational agent. In particular, the organizational agent is a key role, being affected by important meeting resources and acting as an organizational liaison. The lack of organizational integration of meetingware may be attributed to this lack of support to the organizational agent.

The second component is dedicated to characterize meeting resources. On the one hand, this component characterizes the major data elements that get in and out of the meeting (agenda, report and support documents). On the other hand, this component identifies how information is managed by the meetingware (e.g. facilities, meeting configurations and time/space). So, we complement the functional description offered by the previous component with a model of data managed by meetingware at a high level of abstraction. Certainly, the important aspect to emphasize here is the support to resources that are important not only to meetings but extends to the whole organization: the agenda and report. Considering that some degree of formalization is achieved by the model, we expect that meetingware can be more tightly coupled to other organizational tools, such as, for instance, workflow systems.

The third component addresses the meeting process. This component affords decomposing the meeting process in several activities and sub-activities, while describing users' interventions in the system.

It should be noted that the model only describes items that are independent from any specific technology, meeting process, decision process, group interaction process, group facilitation process, etc. The model purposefully avoids items and relationships that would restrict the decisions that necessarily have to be made by the designers about the type of control that the meeting participants can exert on the system. In this respect, the model addresses not only the different levels of detail of the information items produced during meetings (ranging from participants' interventions to complete meeting reports) but also the different structures that the system may impose on the meeting process (ranging from none at all, to a precise list of activities followed by the participants). According to our experience, this approach covers most of the alternatives that designers have to consider when crafting this technology.

The application of the model to evaluate meetingware is flexible enough to accommodate different meeting arrangements, software configurations and information resources, as well as different views over process definitions.

The model has also the potential to define an evaluation grid for meetingware, allowing comparisons between quite different tools and systems, and facilitating the convergence between meetingware functionality and organizational needs.

In the line of the Dexter Hypertext model (Halasz & Schwartz 1994), we can envisage another significant application for the model: the definition of a meetingware standard interchange format. The existence of such a standard would permit the exchange of information between meetingware tools and application. This would allow users to take advantage of different meetingware capabilities and their availability in the organization or across organizations.

Furthermore, a common interchange format could even allow a better integration between meetingware and the remaining components of organizational systems. The existence of a meetingware model clearly facilitates the construction of that format, rendering the model elements into a XML-like syntax. This definition will be addressed in future work.

Recommendations to Designers and Developers

In general, the proposed model may bring benefits to researchers by establishing a common language and affording the comparison of research results across experiments. However, considering its descriptive nature, we believe that it may mostly be beneficial to designers and application developers. The following recommendations were drawn from our experience using the model to design and develop meetingware.

First of all, the problem that meetingware is trying to address is inherently complex, because of the particular combination of people, organization and technology. The technology and the way people use it are both complex as well: supporting multiple configurations, different tasks and functionality, and linking many users with distinct roles. The model offers designers the opportunity to evaluate with stakeholders the organizational needs and map them in the relevant meetingware elements, thus avoiding many mismatches that often occur. Thus, the model should be used early in the design process in order to evaluate several meetingware systems offered in the market and identify a set of preliminary design requirements if design is to proceed. The model can be used to typify the most frequent meetings in the organization, define the agendas and reports used by these meetings and analyze any formalisms or bounds used by the meeting participants in their decisions processes.

This detailed description of meetingware elements and roles is also fundamental to meetingware developers. In particular, it identifies where, how and when the meetingware system can be integrated with other organizational tools and systems. For instance, the meeting logistics may be integrated with calendaring tools now common in organizations; the agenda topics and steps, participants interventions and report items may be integrated with group communication support tools; the meeting support documents may be integrated with hypertext tools; or the agenda and report may be integrated in the organizational document repository.

9. Conclusions

This paper proposes a model for meetingware consisting of three components: roles, meeting process and resources. Each one of these components consists mainly of a collection of class definitions. The model is flexible enough to accommodate different meeting arrangements, software configurations and information resources, as well as different views over process definitions.

The major contribution of the model is the clarification of meetingware functionality, offering a comprehensive description with the potential to be reused in the future, either to evaluate the convergence between meetingware functionality and organizational needs, to support the development of future systems and tools, or to facilitate the integration of meetingware with other organizational systems.

Acknowledgments

This paper was partially supported by the Portuguese Foundation for Science and technology, Project POSI/CHS/33127/99.

References

- M. Aiken and J. Carlisle, "An Automated Idea Consolidation Tool for Computer Supported Cooperative Work," *Information and Management*, vol. 23, pp. 373-382, 1992.
- M. Aiken, O. Sheng, and D. Vogel, "Integrating Expert Systems with Group Decision Support Systems," *ACM Transactions on Information Systems*, vol. 9, no. 1, pp. 75-95, 1991.
- M. Aiken and M. Vanjani, "An Automated GDSS Facilitator." 28th Annual Conference of the Southwest Decision Sciences Institute. Dallas, Texas, 1998.
- P. Antunes, C. Costa, and J. Dias, "Applying Genre Analysis to EMS Design: The Example of a Small Accounting Firm." Seventh International Workshop on Groupware, CRIWG 2001. Darmstadt, Germany: IEEE CS Press, 2001, pp. 74-81. (ISBN: 0-7695-1351-4).
- P. Antunes and T. Ho, "Facilitation Tool - A Tool to Assist Facilitators Managing Group Decision Support Systems." Ninth Workshop on Information Technologies and Systems, WITS '99. Charlotte, North Carolina, December, 1999.
- P. Antunes and T. Ho, "The Design of a GDSS Meeting Preparation Tool," *Group Decision and Negotiation*, vol. 10, no. 1, January, pp. 5-25, 2001. (ISSN: 0926-2644).
- C. Beise, F. Niederman, and P. Beranek, "Facilitating Technology-Supported Group Work: A New Category of IS Personnel." Proceedings of the 1992 ACM SIGCPR Conference on Computer Personnel Research. Cincinnati, Ohio, May, 1992.

- R. Bostrom, M. Aiken, L. Motiwalla, O. Sheng, and J. Nunamaker, "ESP: An Expert System for Pre-Session Group Decision Support Systems Planning," in *Proceedings of the Twenty-Third Hawaii International Conference on Systems Sciences*. Kailua-Kona, Hawaii, 1990, pp. 279-286.
- R. Bostrom, R. Anson, and V. Clawson, "Group Facilitation and Group Support Systems," in *Group Support Systems: New Perspectives*, L. Jessup and J. Valacich, Eds. New York: Macmillan, 1993.
- R. Briggs and G. Vreede, "ThinkLets: Achieving Predictable, Repeatable, Patterns of Group Interaction with Group Support Systems (GSS)." *Proceedings of the 34th Hawaii International Conference on System Sciences*, 2001.
- T. Bui and M. Jarke, "Communications Design for Co-OP: A Group Decision Support System," *ACM Transactions on Office Information Systems*, vol. 4, no. 2, April, 1986.
- V. Clawson, R. Bostrom, and R. Anson, "The Role of the Facilitator in Computer-Supported Meetings," *Small Group Research*, vol. 24, no. 4, pp. 547-565, 1993.
- J. Conklin and M. Begeman, "GIBIS: A Hypertext Tool for Exploratory Policy Discussion," *ACM Transactions on Office Information Systems*, vol. 6, no. 3, pp. 303-331, 1988.
- T. Connolly, L. Jessup, and J. Valacich, "Effects of Anonymity and Evaluative Tone on Idea Generation in Computer-Mediated Groups," *Management Science*, vol. 36, no. 6, pp. 689-703, 1990.
- C. Costa and P. Antunes, "Meetings as Genre Systems: Some Consequences for EMS Design," in *Proceedings of Group Decision & Negotiation 2001*, F. Ackermann and G. Vreede, Eds. La Rochelle, France: Faculty of Technology, Policy and Management, Delft University of Technology, 2001, pp. 261-263. (ISBN: 90-5638-078-8).
- C. Costa, P. Antunes, and J. Dias, "A Model for Organizational Integration of Meeting Outcomes," in *Contemporary Trends in Systems Development*, Maung K. Sein, Bjørn-Erik Munkvold, Tore U. Ørvik, Wita Wojtkowski, W. Gregory Wojtkowski, Joze Zupancic, and Stanislaw Wrycza, Eds. Kluwer Plenum, 2001. (Papers from the Ninth International Conference on Information Systems Development, ISD 2000. ISBN: 0-306-46608-2).
- C. Costa, P. Antunes, and J. Dias, "Integrating Two Organisational Systems Through Communication Genres." *Fifth International Conference on Coordination Models and Languages (Coordination 2002)*. York, UK: Lecture Notes in Computer Science, Springer-Verlag, 2002.
- C. Costa, T. Ho, and P. Antunes, "Facilitating Organisational Activities Using Plans and Audits." *1st International Conference on Enterprise Information Systems, ICEIS '99*. Setubal, Portugal, 1999, pp. 404-411. (ISBN: 972-98050-0-8).
- A. Dennis, J. George, L. Jessup, J. Nunamaker, and D. Vogel, "Information Technology to Support Electronic Meetings," *Management Information Systems Quarterly*, vol. 12, no. 4, pp. 591-624, 1988.
- G. DeSanctis and R. Gallupe, "A Foundation for the Study of Group Decision Support Systems," *Management Science*, vol. 33, no. 5, pp. 589-609, 1987.
- G. Dickson, J. Partridge, and L. Robinson, "Exploring Modes of Facilitative Support for GDSS Technology," *Management Information Systems Quarterly*, June, pp. 173-194, 1993.
- C. Eden, "Strategic Options Development and Analysis (SODA)," in *Rational Analysis in a Problematic World*, J. Rosenhead, Ed. Chichester: Wiley, 1989, pp. 21-42.
- J. Fjermestad and S. Hiltz, "An Analysis of the Effects of Mode of Communication on Group Decision Making." *Thirtieth One Hawaii International Conference on Systems Science (HICSS-31)*. Maui, 1998.

- J. Fjermestad and S. Hiltz, "An Assessment of Group Support Systems Experimental Research: Methodology and Results," *Journal of Management Information Systems*, vol. 15, no. 3, pp. 7-149, 1999.
- R. Gallupe, W. Cooper, M. Grisé, and L. Bastianutti, "Blocking Electronic Brainstorms," *Journal of Applied Psychology*, vol. 9, no. 1, pp. 77-89, 1991.
- N. Gonçalves and P. Antunes, "'Decision Can': A Database of Decision Cases." Second International Conference on Enterprise Information Systems, ICEIS 2000. Stafford, UK, 2000, pp. 139-143. (ISBN: 972-98050-1-6).
- F. Halasz and M. Schwartz, "The Dexter Hypertext Reference Model," *Communications of the ACM*, vol. 37, no. 2, pp. 30-39, 1994.
- T. Ho and P. Antunes, "Developing a Tool to Assist Electronic Facilitation of Decision-Making Groups." Fifth International Workshop on Groupware, CRIWG '99. Cancun, Mexico: IEEE CS Press, 1999, pp. 243-252. (ISBN: 0-7695-0268-7).
- J. Hoffer and J. Valacich, "Group Memory in Group Support Systems: A Foundation for Design," in *Computer Augmented Teamwork: A guided tour*, Bostrom, Watson, and Kinney, Eds. New York: Van Nostrand Reinhold, 1991.
- V. Jacob and H. Pirkul, "A Framework for Supporting Distributed Group Decision-Making," *Decision Support Systems*, vol. 8, pp. 17-28, 1992.
- A. Jay, "How to Run a Meeting," *Harvard Business Review*, vol. 54, no. 2, March-April, 1976.
- S. Kaner, *Facilitator's guide to participatory decision-making*. Philadelphia, PA: New Society Publishers, 1996.
- R. Kazman, R. Al-Halimi, W. Hunt, and M. Mantei, "From Paradigms for Indexing Videoconference," *IEEE Multimedia Magazine*, Spring, pp. 63-73, 1996.
- K. Kraemer and J. King, "Computer-Based Systems for Cooperative Work and Group Decision Making," *ACM Computing Surveys*, vol. 20, no. 2, pp. 115-146, 1988.
- L. Lewis, "A Decision Support System for Face-to-Face Groups," *Journal of Information Science*, vol. 13, pp. 211-219, 1987.
- T. Malone and K. Crowston, "The Interdisciplinary Study of Coordination," *ACM Computing Surveys*, vol. 26, no. 1, March, pp. 87-119, 1994.
- H. Mintzberg, *The Structuring of Organizations: A synthesis of the research*. Englewood Cliffs, NJ: Prentice-Hall, 1979.
- S. Miranda and R. Bostrom, "Meeting Facilitation: Process Versus Content Interventions," *Journal of Management Information Systems*, vol. 15, no. 4, pp. 89-114, 1999.
- F. Niederman and R. Volkema, "Influence of Agenda Creation and Use on Meeting Activities and Outcomes: Report on Initial Results," in *Proceedings of the 1996 Conference on ACM SIGCPR/SIGMIS Conference*. Denver, Colorado, 1996, pp. 192-205.
- J. Nunamaker, R. Briggs, D. Mittleman, D. Vogel, and P. Balthazard, "Lessons from a Dozen Years of Group Support Systems Research: A Discussion of Lab and Field Findings," *Journal of Management Information Systems*, vol. 13, no. 3, pp. 163-207, 1997.
- J. Nunamaker, A. Dennis, J. George, Martz Jr. W., J. Valacich, and D. Vogel, "GroupSystems," in *Computer Augmented Teamwork: A guided tour*, R. Bostrom, Watson, and Kinney, Eds. New York: Van Nostrand Reinhold, 1991b.
- J. Nunamaker, A. Dennis, J. Valacich, D. Vogel, and J. George, "Electronic Meeting Systems to Support Group Work: Theory and Practice at Arizona," *Communications of the ACM*, vol. 34, no. 7, pp. 40-61, 1991a.
- W. Orlikowski and J. Yates, "Genre Repertoire: The Structuring of Communicative Practice in Organizations," *Administrative Science Quarterly*, vol. 39, pp. 547-574, 1994.
- W. Orlikowski and J. Yates, *Genre systems: Structuring interaction through communicative norms*, CCS WP 205. Sloan MIT WP 4030, 1998.

- G. Raikundalia and M. Rees, "Exploiting the World-Wide Web for Electronic Meeting Document Analysis and Management." Proceedings of the Asia-Pacific World-Wide Web '95 Conference. Sydney, September, 1995a.
- G. Raikundalia and M. Rees, "Scenario of Web User Interface Tools for Electronic Meeting Document Generation and Presentation." QCHI95 Symposium. Bond University, August, 1995b.
- G. Raikundalia and M. Rees, "Enhancing Collaboration in Formal, Synchronous Electronic Meetings with LoganWeb." Proceedings of 1996 Australian National Symposium on Computer-Supported Cooperative Work (OzCSCW96). University of Queensland, 30 August, 1996.
- V. Rao and S. Jarvanpaa, "Computer Support of Groups: Theory-Based Models for GDSS Research," *Management Science*, vol. 37, no. 10, pp. 1347-1362, 1991.
- N. Romano and J. Nunamaker, "Meeting Analysis: Findings from Research and Practice." Proceeding of the 34th Hawaii International Conference on Systems Science. Hawaii, 2001.
- N. Romano, J. Nunamaker, D. Roussinov, and H. Chen, "Collaborative Information Retrieval Environment: Integration of Information Retrieval with Group Support Systems." Proceedings of the 32nd Hawaii International Conference on System Sciences. Maui, Hawaii: IEEE Computer Society Press, 5-8 January, 1999.
- R. Schwarz, *The skilled facilitator*. San Francisco, CA: Jossey-Bass Publishers, 1994.
- M. Silver, *Systems that support decision makers: Description and analysis*. New York: John Wiley, 1991.
- H. Simon, *Administrative behavior: A study of decision-making processes in administrative organizations (4th edition)*. New York: Free Press, 1997.
- N. Streitz, P. Rexroth, and T. Holmer, "Does Roomware Matter? Investigating the Role of Personal and Public Information Devices and Their Combination in Meeting Room Collaboration." European Conference on CSCW (ECSCW '97), 1997.
- The 3M Meeting Management Team, *Mastering Meetings*. New York: McGraw-Hill, Inc., 1994.
- G. Wagner, B. Wynne, and B. Mennecke, "Group Support Systems Facilities and Software," in *Group Support Systems: New Perspectives*, L. Jessup and J. Valacich, Eds. New York: Macmillan, 1993.
- I. Zigurs and B. Buckland, "A Theory of Task/Technology Fit and Group Support Systems Effectiveness," *Management Information Systems Quarterly*, September, pp. 313-334, 1998.

P- Manager: Um Sistema de Apoio ao Gestor

Manuela Aparício

LusoCrédito, Lda. Lisboa.

J. Paulo Costa

LusoCrédito, Lda. Lisboa.

Carlos J. Costa

Departamento de Ciências e Tecnologias de Informação, ISCTE, Lisboa.

Resumo

A utilização de ferramentas de apoio ao trabalho de gestão é algo de grande importância. Consequentemente, o trabalho aqui reportado tem por objectivo o desenvolvimento de uma ferramenta de apoio. É nesse contexto que se propõe um sistema de suporte ao planeamento, organização direcção e controlo de operações. Pretende-se ainda potenciar a sua utilização através do uso de computadores de mão (*handhelds*) que possam comunicar entre si ou com outros computadores de mesa.

1 Introdução

A computação móvel, ou *mobile computing* [Kalakota, 2001], tem por base tecnológica PDA (*Personal Digital Assistant*) como ainda os telefones sem fios (telemóveis) e *notebooks* com *wireless*. Por outro lado, o uso do conceito de computação ubíqua (*ubiquitous computing*) não parece o conceito mais adequado, uma vez que pressupõe a omnipresença de sistemas de computação [Weiser, 1993]. Por conseguinte, considera-se mais adequado designar por P-Computing (P de PDA, Palm ou Pocket).

Através dos computadores de mão, os seus utilizadores têm acesso a quantidade significativa de informação longe do seu escritório. Por outro lado, estes dispositivos permitem ainda a realização de tarefas relativamente complexas que há alguns anos eram possíveis só com computadores de grande porte. De simples agendas electrónicas, às vezes confundidos com brinquedos e máquinas de jogo, estes equipamentos passaram a ter um estatuto fundamental [Schwartz, 2000], no apoio à enfermagem e medicina ([Jenkins, 2002], [Garvin et al. 2000]), no âmbito do ensino da medicina [Helwig & Flynn, 1998] ou apoio a decisão de médicos cardiologistas [Heydtmann, 1999], sem contar as importantes potencialidades ao nível dos sistemas de apoio a grupos ([Davis et al. 1999], [Myers, et al. 2000]). Porém, a sua utilização no âmbito do apoio à coordenação de tarefas executadas pelos gestores é algo que não tem sido objecto de atenção suficiente.

De modo a identificar as principais funcionalidades do sistema, estudou-se a forma como os gestores trabalham. Em vez de partir de uma necessidade e de um caso particular, algo que tipicamente se realiza quando se desenvolve uma aplicação comercial, foi empreendida uma abordagem que poderá ser designada de “académica”, ou seja, foi realizada uma revisão da literatura. Verificou-se então que este assunto tem preocupado os investigadores, sendo particularmente apelativo um artigo de [Minzberg, 1975], em que este autor identifica os papéis do gestor. Segundo este autor, o gestor desempenha dez tipos de papéis, que podem por sua vez ser agrupados em três: papéis interpessoais, papéis informacionais e papéis decisoriais. A autoridade formal leva à realização dos três papéis interpessoais, que por sua vez originam os três papéis informacionais. Estes dois conjuntos de papéis permitem ao gestor desempenhar os quatro

papéis decisoriais. Os três papéis interpessoais desempenhados pelo gestor resultam directamente da sua autoridade formal, envolvendo relações entre as pessoas, quer de uma unidade, quer externa a esta. São eles os seguintes: figura de proa (*figure head role*), líder (*leader role*), pessoa que estabelece ligação (*liaison role*). Por virtude dos contactos interpessoais os gestores como centros neurálgicos da unidade organizacional vão desempenhar três papéis que descrevem o aspecto informacional do trabalho gestor. Neste âmbito, três papéis descrevem o aspecto informacional do trabalho do gestor: pessoa que recolhe informação (monitor), disseminador (*disseminator*) e porta-voz para pessoas fora da organização (*spokesperson*). A informação não é um fim em si própria, é apenas um *input* para a tomada de decisão. Com efeito, o gestor desempenha o papel principal no sistema de tomada de decisão da unidade. Neste âmbito, segundo Mintzberg existem quatro papéis que descrevem o gestor como decisor: empreendedor (*entrepreneur*), gestor de conflitos (*disturbance handler*), pessoa que faz a afectação de recursos (*resource allocator*) e negociador (*negociator*).

Embora se trate de uma abordagem particularmente interessante tem uma utilidade bastante limitada. Uma análise mais atenta da literatura da gestão veio permitir identificar outra abordagem que dá ênfase ao processo de gestão. Esta abordagem foi proposta por um engenheiro francês no início do século XX [Fayol, 1926], segundo o qual o processo de administração (ou gestão) consiste de cinco funções: prever (visualizar o futuro e traçar o programa de acção), organizar (munir a empresa de todos os recursos que ela carece), comandar (dirigir e orientar o pessoal), coordenar (harmonizar os actos e o esforço colectivo) e controlar (verificar que tudo se realiza de acordo com as regras estabelecidas). Autores posteriores, sugeriram as seguintes funções: planejar, organizar, dirigir e controlar [Dale, 1973]; planejar, organizar, liderar e controlar [Newman et al., 1967]; ou então planejar, organizar, dirigir, coordenar e controlar [Miner, 1973].

2 Estudo de Funcionalidades

2.1 Estados de uma operação

Uma tarefa pode ser decomposta, ao nível mais elementar, em operações. Com vista a fazer a sua gestão e partindo do conceito de funções de gestão foram identificados diversos estados. Para além do estado planeado, pode ser identificado o estado aceite, que em função da estrutura organizacional (e de poder) pode revestir de características específicas podendo ser designado de autorizado ou confirmado. Finalmente a operação será realizada.

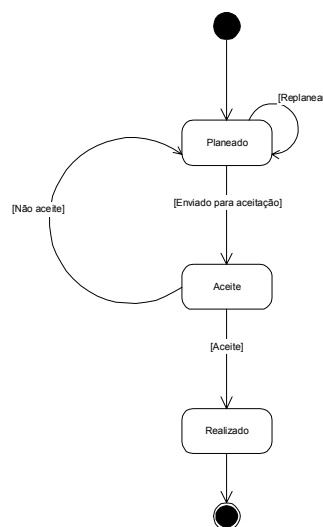


Figura 1- Estados de uma tarefa na perspectiva da sua gestão

2.2 Três cenários

Partindo dos estados identificados atrás podem ser identificadas as actividades correspondentes (planear, confirmar e realizar). Partindo destas actividades e ainda do pressuposto que existem vários agentes que realizam cada uma dessas actividades podem ser identificados três cenários: operacional/operacional, chefe/operacional e director/operacional.

No cenário operacional/operacional, cada um dos operacionais é responsável pelas três actividades (planear, confirmar e realizar).

Existe neste cenário a possibilidade de cruzamento de informação entre os operacionais podendo-se assim criar um ambiente que se proporciona favorável ao trabalho em equipa ([Follet, 1924], [Kanter, 1983]).

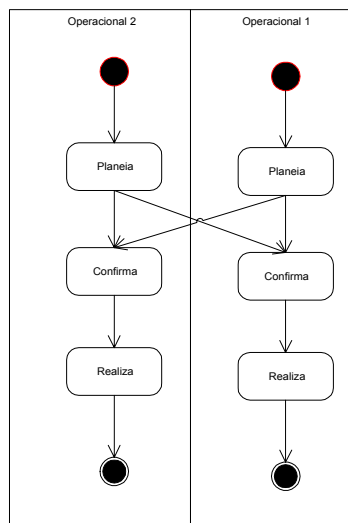


Figura 2- Cenário operacional/operacional

No cenário chefe/operacional, o planeamento é feito pelo superior que em seguida dá instruções para que as operações sejam realizadas pelo operacional. O operacional em seguida aceita (ou eventualmente não) as instruções dadas pelo superior. Existe ainda a possibilidade de o operacional eventualmente não aceitar as instruções dadas pelo chefe.

Este cenário corresponde aproximadamente ao da supervisão directa descrito por [Mintzberg, 1979].

Há ainda que considerar que este cenário, apesar de geralmente estar associado a um nível de centralização elevado poderá ainda acomodar elevadas descentralização. Com efeito, o chefe tanto poderá dar instruções muito específicas (centralização elevada), como transmitir objectivos mais genéricos (o que corresponde a grande descentralização).

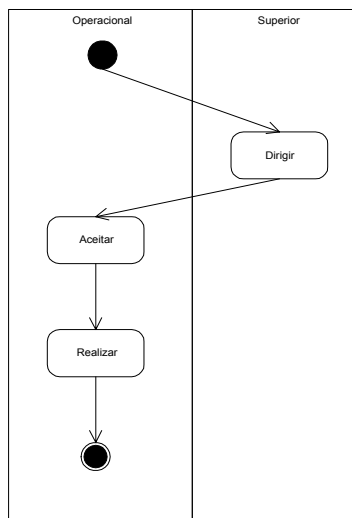


Figura 3- Cenário chefe/operacional

No cenário director/operacional existe alguma autonomia do operacional mas não dependência. Este faz o planeamento mas em seguida pede autorização. O director vai então autorizar ou não o plano, sendo em seguida este plano concretizado pelo subordinado.

Este cenário é de algum modo alusivo a uma gestão por objectivos, preconizada por Peter Drucker [Druker, 1964]. As fases deste tipo de gestão compreenderem também a planificação ou a fixação de objectivos e tarefas e sua posterior aceitação quer por subordinados quer por superiores.

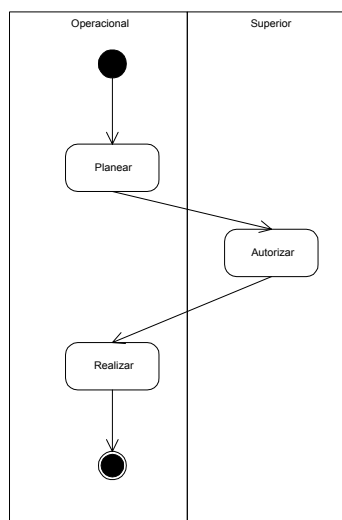


Figura 4- Cenário director/operacional

2.3 Mecanicista vs. Orgânico

Há ainda que ter em linha de conta que a organização pode ter uma estrutura mecanicista ou orgânica [Burns & Stalker, 1994]. No caso de se tratar de uma estrutura mecanicista existe uma ligação mais clara (e rígida) entre os agentes e as tarefas. No caso de se ter uma estrutura orgânica os agentes podem desempenhar um conjunto de tarefas mais vasto. Este aspecto é

particularmente importante na concepção de um sistema uma vez que vai permitir identificar as políticas que podem (e/ou devem) ser incorporadas.

3 Proposta de um sistema

3.1. Arquitectura do Sistema

Na figura que se segue é possível ver a arquitectura do sistema, para um caso particular com dois sistemas clientes de PDA, com ligações aos respectivos computadores pessoais (PC). A componente de servidor é composta por um servidor de Bases de Dados e um Servidor de Web. Estes permitem comunicação com PDAs quer através de sincronização (Sync) quer através de Internet.

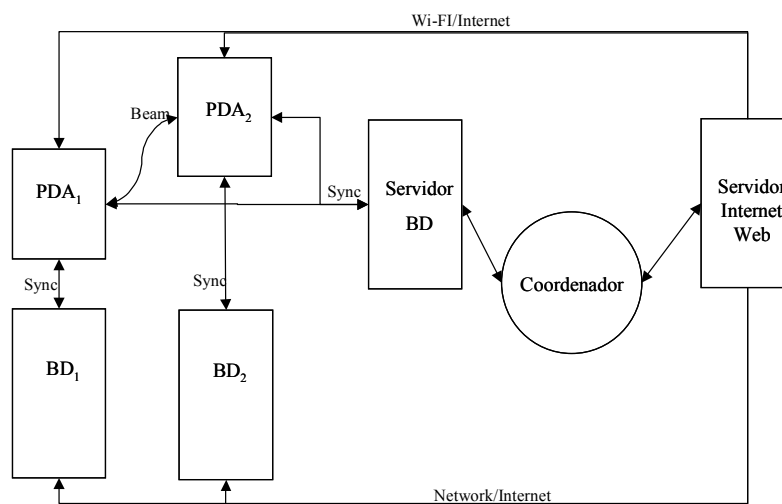


Figura 5- Arquitectura do Sistema

3.2 Estrutura de dados

O sistema deve armazenar informação relativa a cada pessoa. Nomeadamente o seu nome, função, departamento e contacto (ex.: e-mail). Cada pessoa lidera uma ou várias tarefas. As tarefas devem ter identificação da sua designação, objectivo, data de início e fim, bem como hora de início e fim. Cada tarefa é constituída por uma série variável de operações. A operação deve indicar a respectiva designação e prioridade. Cada pessoa planeia, aceita ou realiza uma ou várias operações. É importante identificar quando a operação foi planeada, aceite e realizada. Quer no caso da aceitação quer da realização é ainda importante que o responsável rubrique essas operações. Na Figura 6 pode ver-se o modelo conceptual correspondente.

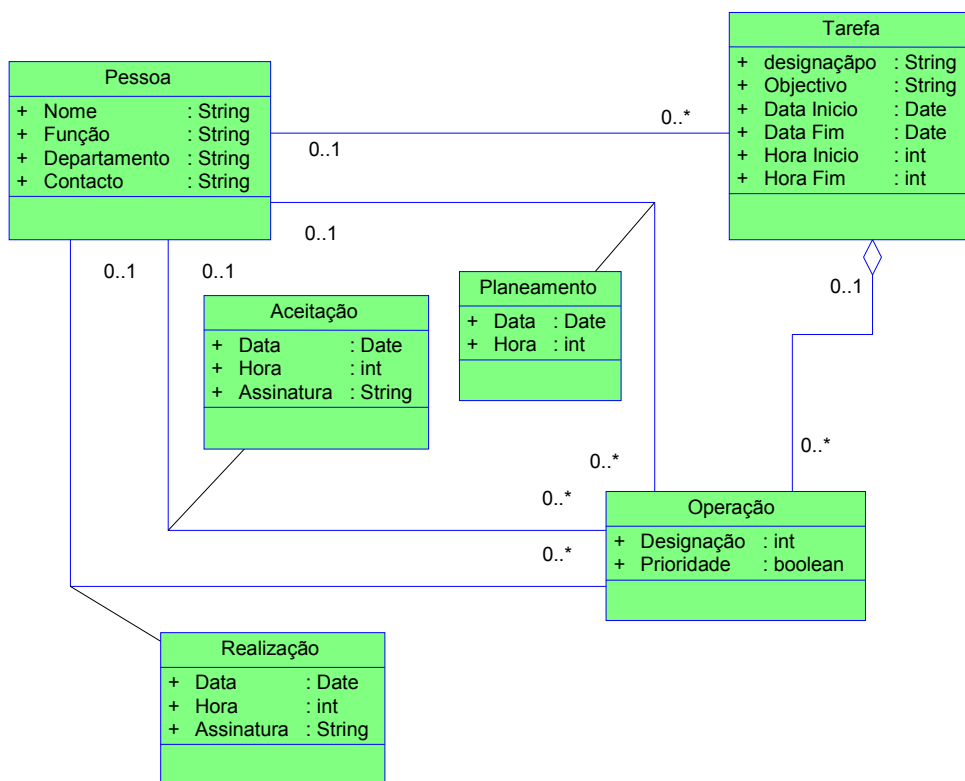


Figura 6- Modelo conceptual

3.3 Interface PDA

O utilizar define um conjunto de tarefas. Cada tarefa é composta por várias operações. De modo a ilustrar, considere-se uma tarefa designada por “Rede *Wireless*”, composta pelas seguintes operações: “Análise de Viabilidade”, “Aquisição”, “Implementação” e “Teste de aceitação”. Por sua vez, a cada uma destas operações está associado uma ou várias pessoas. Cada pessoa pode liderar (L) um tarefa, planear (P), aceitar (A) ou realizar (R) uma operação.

Pessoas	Papeis	CM	JJ	MM
Tarefas/Operações				
1. Rede <i>Wireless</i>	L		X	
1.1. Análise de Viabilidade	P	X		
	A		X	
	R	X		
1.2. Aquisição	P	X		
	A		X	
	R	X		
1.3. Implementação	P	X		
	A		X	
	R	X		
1.4. Teste de aceitação	P	X		
	A			X
	R			X

Figura 7- Exemplo

O menu do sistema mostra todas as funcionalidades disponíveis. De referir que a disponibilidade das funcionalidades depende das políticas definidas, nomeadamente da identificação de estrutura mais ou menos orgânica. De referir que cada tarefa é liderada por uma pessoa, sendo por sua vez decomposta em operações.

The figure shows three screenshots of a PDA interface for the 'Rede de Wireless' task.

- Menu:** A screen with a title bar 'Menu' and several buttons: 'Planear', 'Tarefa', 'Aceitar', 'Realizar', 'Listar', and 'Sair'.
- Tarefas:** A screen with a title bar 'Tarefas' showing details for 'Rede Wireless'. It includes fields for 'Designação', 'Objectivo' (Implementar rede Wi-Fi), 'Lider' (JJ), 'Inicio' (16/3/03 11:25), and 'Fim' (27/10/03 15:25). There is a 'Proc.' button and a set of navigation buttons at the bottom: 'Anterior', 'Seguinte', 'novo', 'Apagar', 'Beam', and 'Menu Principal'.
- Lista:** A screen with a title bar 'Lista' showing a list of operations with columns 'Data' and 'Conc'. The list includes:

Data	Conc
17/3/03	Análise de Viabilidade
12/8/03	Aquisição
12/8/03	Implementação
12/8/03	Teste de aceitação

 At the bottom, there are buttons for 'Aquisição', 'Plano', 'Aceit', 'Real', 'Eliminar', 'Menu', 'Anterior', and 'Seguinte'.

Figura 8- Menu e lista de operações da tarefa “Rede de Wireless”

As operações serão sujeitas a planeamento, aceitação (ou autorização) e realização. Na Figura 9, são mostrados os ecrãs que podem ser vistos no PDA.

The figure shows three screenshots of a PDA interface for the 'Análise de Viabilidade' operation.

- Plano:** A screen with a title bar 'Plano' showing details for 'Análise de Viabilidade'. It includes fields for 'Descrição', 'Data' (17/3/03), 'Hora' (14:26), 'Tarefa' (Rede Wireless), 'Realiza' (CM), 'Prioridade' (1), and 'Data' (3/6/03). There is a 'Procurar' button and a set of navigation buttons at the bottom: 'Anterior', 'Seguinte', 'novo', 'Listar', 'Apagar', 'Beam', and 'Menu Principal'.
- Aceitar:** A screen with a title bar 'Aceitar' showing details for 'Análise de Viabilidade'. It includes fields for 'Descrição', 'Data' (17/3/03), 'Hora' (14:26), 'Tarefa' (Rede Wireless), 'Realiza' (CM), and 'Aceita' (JJ). There is a 'Procurar' button and a set of navigation buttons at the bottom: 'Anterior', 'Seguinte', 'Menu Principal', and 'Beam'.
- Realizar:** A screen with a title bar 'Realizar' showing details for 'Análise de Viabilidade'. It includes fields for 'Descrição', 'Data' (17/3/03), 'Hora' (14:26), 'Tarefa' (Rede Wireless), 'Realiza' (CM), and 'Data Conc' (28/3/03). There is a 'Procurar' button and a set of navigation buttons at the bottom: 'Anterior', 'Seguinte', 'Menu Principal', and 'Beam'.

Figura 9- Planejar, aceitar e realizar da operação “Análise de Viabilidade”

Os dados referidos atrás traduzem-se nas seguintes variáveis e ecrãs (*forms*) implementados no computador de mão. No quadro da Figura 10, podem ser vistas as variáveis associadas a cada um dos ecrãs. Os ecrãs apresentados (plano, aceitar e realizar) são os apresentados na Figura 9. Em cada um dos ecrãs, os utilizadores podem criar (C) ou simplesmente consultar (V) dos dados contidos nas diversas variáveis.

Ecrã Variáveis	Plano	Aceitar	Realizar	Descrição
DscA	C	V	V	Nome da actividade (planeado)
Dta1	C	V	V	Data da actividade (planeada)
Hra1	C	V	V	Hora da actividade (planeada)
Tarf	C	V	V	Nome da tarefa
Rla1	C	V		Quem realiza actividade (planeado)
Prio	C			Prioridade da actividade
Dtpl	C			Data de planeamento da actividade
Acpl	C			Quem planeia actividade
Aca1		C		Quem aceita actividade
Dtac		C		Data de aceitação
Hrac		C		Hora de aceitação
Aact		C		Assinatura de aceitante
Rla2			C	Quem realiza actividade
Dtrl			C	Data de realização
Hrrl			C	Hora de realização
Arlz			C	Assinatura de realização

Figura 10- Ecrãs e variáveis

3.4. Componente Computador de Mesa

Neste sistema, para além de se fazer a implementação de uma componente no PDA também foi implementado uma componente no computador de mesa. No que diz respeito à componente correspondente ao computador de mesa, esta incorpora a base de dados correspondente ao modelo conceptual apresentado atrás. Entretanto algumas dificuldades se levantaram, nomeadamente:

- Como ultrapassar questões de confidencialidade?
- Que políticas de perfis e permissões devem ser implementadas?
- Como fazer sincronização quando existe conflito entre informação existente nos vários PDA? Que informação deverá prevalecer: a do PDA ou a do computador de mesa?
- Como fazer transição entre este sistema e outros sistemas, como por exemplo as agendas?

De imediato, uma das vantagens deste módulo é a possibilidade de eliminação de tarefas redundantes e ainda a possibilidade de uma mais adequada sincronização da informação existente nos diversos sistemas.

4. Discussão

O sistema brevemente apresentado foi implementado num ambiente de desenvolvimento comum a PalmOS e PocketPC. Foi ainda implementado um módulo para o computador de mesa que permite fazer a gestão do sistema, nomeadamente a eliminação de operações redundantes.

De referir que nos testes realizados se tem privilegiado a interacção entre os computadores de mão utilizando o infravermelho. É ainda de referir que tem sido utilizados diversos equipamentos com níveis de sofisticação bastante variado, desde um Palm M100 a um TugstenC, tendo apresentado sempre um elevado nível de viabilidade técnica em todo os sistemas.

O sistema foi testado apenas quanto às suas funcionalidades e de forma parcial. É necessário agora realizar um teste mais geral e verificar em que medida este sistema permite mais adequado apoio à gestão.

O problema de gestão de tempo e coordenação de tarefas está a ser resolvido recorrendo a diversas ferramentas [Malone & Crowston, 1994]. Os sistemas de agenda pessoais podem ser utilizados para gerir o tempo (e.g. www.palmone.com). Especificamente para resolver o problema de optimização no âmbito da gestão de projecto são utilizadas técnicas como o PERT [PERT, 1958] e CPM [Wiest & Levy, 1977] ou são utilizadas técnicas mais genéricas de escalonamento e calendarização de actividades [Schroeder, 1985].

A forma de governação, tal como a estrutura organizacional também tem sido considerada por diversos investigadores [Malone, et al., 1987 e Malone, 1987].

Mas, geralmente, esses sistemas ignoram as pessoas que devem ser responsáveis, às quais devem ser dada autoridade, pessoas que ainda devem aceitar essa autoridade e responsabilidade (Andersen et al., 1984).

No que diz respeito a software desenvolvido para PDA não existem muitos sistemas. De facto, existem algumas ferramentas de calendarização e gestão de projectos, nomeadamente as fornecidas pelos próprios vendedores de PDA. No que diz respeito a coordenação entre utilizadores de PDA, foram desenvolvidos diversos sistemas, especialmente coordenação utilizando sistemas de apoio a reuniões ([Davis et al., 1999], [Wiberg, 2001], [Myers et al., 2000] e [Costa et al., 2001]).

5. Conclusão

Neste artigo foi proposta o conceito de P-Computing, em contraposição com a computação móvel e computação ubíqua. Foi ainda analisado em que consistia o trabalho do gestor, o que foi possível recorrendo a conceitos provenientes da gestão e ao estudo do processo de gestão proposto pelos teóricos. Porém, esse processo foi ajustado de modo a se tornar computacionalmente mais ajustado. Foi deste modo possível identificar funcionalidades de um sistema que foi proposto e desenvolvido – o P-Manager.

Referências

- Andersen, E., Grude, K., Haug, T. & Turner J. *Goal Directed Project Management*. Kogan. Page/Coopers & Lybrand. 1984
- B. Myers, R. Miller, B. Bostwick & C. Evankovich "Extending the Windows Desktop Interface With Connected Handheld Computers". *4th USENIX Windows Systems Symposium*. Seattle,, 2000, 79-88.
- Burn T. & Stalker M. *The Management of Innovation*, Oxford University Press NY.1994.
- Costa, C., Antunes, P. & Dias, J. "EMS/PDA: Connecting Meetings with People in Organisations," in *Proceedings of the 24th Information Systems Research Seminar in Scandinavia, IRIS 24*. Ulvik in Hardanger, Norway, 2001.
- Dale, E., *Management: Theory and Practice*, McGraw-Hill, NY, 1973.
- Davis, R., Landay, J. & Chen V., "NotePals: Lightweight note sharing by the group, for the group". *Proceedings of the CHI 99 Conference on Human Factors in Computing Systems*. ACM Press, Pittsburg, 1999, 338-345.
- Davis, R., Landay, J. & Chen V. "NotePals: Lightweight note sharing by the group, for the group" in. *Proceedings of the CHI 99 Conference on Human Factors in Computing Systems*. ACM Press, Pittsburg. 1999. 338-345.
- Drucker, P. *The Practice of Management*, London, Heinemann. 1964.
- Fayol, H. *Administration Industrielle et Générale*; Paris, 1926.
- Follet, M. *Creative Experience*, Longman London, 1924
- Garvin R., Otto F. & McRae D. Using handheld computers to document family practice resident procedure experience. *Family Medicine*. Feb;32(2) (2000), 115-8.
- Helwig, A. & Flynn, C. "Using palm-top computers to improve students' evidence-based decision making". *Academic Medicine*, May, 73-5, (1998) 603-4.
- Heydtmann, M. "Cardiovascular risk calculation on a pocket sized computer". *Journal of the Royal College of Physicians of London*. May-Jun; 33-3, (1999), 287
- Jenkins, D. Personal digital assistants: a world of information in the palm of your hand. *Clinical Nurse Specialist*. Jan;16 -1, (2002) 38-9.
- Kalakota, R. *mBusiness: The race for Mobility*, McGraw-Hill, 2001.
- Kanter, R. *The Change Masters*, Simon & Schuster, NY, 1983.
- M. Wiberg. RoamWare: An integrated architecture for seamless interaction in between mobile meetings. In *Proceedings of the 2001 International ACM SIGGROUP Conference on Supporting Group Work*. ACM Press, Boulder, Colorado, 2001, 288-297.
- Malone, T. & Crowston, K.. "The Interdisciplinary Study of Coordination". *ACM Computing Surveys*. 26. 1. March. (1994).
- Malone, T. "Modeling Coordination in Organizations and Market". *Management Science*. Vol. 33; No. 10. (1987)

- Malone, T., Yates, J. & Benjamin, R. "Electronic markets and electronic hierarchies". *Communications of the ACM*. 30. (1987) , 484-497..
- Miner, J. *The Management Process, Theory, Research and Practice*, MacMillan, NY, 1973.
- Mintzberg, H. "The Manager's Job: Folk and Facts", *Harvard Business Review*, July - August. (1975)
- Mintzberg, H. *The Structuring of Organizations: A Synthesis of Research*, Prentice-Hall, Englewood Cliffs, 1979.
- Myers, B., Lie, K. & Yang, B "Two-Handed Input Using a PDA and a Mouse", *Proceedings CHI'2000: Human Factors in Computing Systems*. April 1-6, 2000. The Hague, The Netherlands, 2000, pp. 41-48.
- Newman, W; Summer, C. & Warren, K. *The Process of Management. Concepts, Behavior and Practice*, Prentice-Hall, Englewood Cliffs, 1967.
- PERT, *Program Evaluation Research Task. Phase 1 Summary Report. Special Projects Office. Beaureau of Ordinance*. Department of the Navy. Washington. 646-669. July. 1958
- Schroeder, R. *Operation Management – Decision making in the Operations Function*. 3rd Edition. McGraw-Hill. 1985.
- Schwartz, K. "It's a tool, not a toy". *Mobile Computing & Communication*. 11 - 5, (2000). 77-84.
- Weiser, M "Some Computers Issue in Ubiquitous computing"; *Communications of the ACM*; 36 – 7; (1993), 78-84.
- Wiest, J. & Levy, F. *A management Guide to PERT/CPM*. 2nd Ed. Englewood Cliffs. NJ. Prentice Hall. 1977.

Segurança dos Web Services no Comércio Eletrônico Móvel

Ricardo Martins, Jorge Rocha, Pedro Henriques

Universidade do Minho Departamento de Informática – Campus de Gualtar - Braga
{ram,jgr,prh}@di.uminho.pt

RESUMO

A cada dia, a quantidade de serviços disponíveis para dispositivos móveis aumenta de acordo com o surgimento de novas tecnologias. A tecnologia dos Web Services é considerada como sendo uma importante forma de integração de dados entre diferentes plataformas e programas. Ao olhar para os Web Services dentro do comércio eletrônico, é possível destacar o horizonte de possibilidades que surgem. Entretanto, por se tratar de uma tecnologia recente, não é possível avaliarmos os riscos de segurança que podem advir. Dessa forma, não é possível garantir que o uso dos Web Services seja seguro para aplicações voltadas ao comércio eletrônico móvel. O objetivo deste artigo é comparar as tecnologias de segurança mais vulgarmente usadas no comércio eletrônico para verificar quais as que podem ser utilizadas com os Web Services.

KEYWORDS

M-Commerce, segurança, Web Services, XML.

1. INTRODUÇÃO

Os Web Services constituem um novo modelo de partilha de dados que permite publicação de rotinas e métodos acessíveis pela Internet. Com uma interface transparente para o Cliente e facilidade de integração de diferentes aplicações, os Web Services são uma poderosa ferramenta para promover a integração dos dados em diferentes plataformas, incluindo os dispositivos móveis.

A cada dia, o número de Clientes de dispositivos móveis aumenta, devido a novos avanços tecnológicos e serviços, juntamente com uma maior interoperabilidade entre os sistemas. De acordo com [Cahners 2000], o número de dispositivos móveis (Personal Data Assistants – PDAs – e smart-phones) que estarão equipados com dispositivos de acesso a dados é estimado em torno de 1.5 bilhões no final de 2004 em todo o mundo. O número estimado de assinantes de serviços de dados wireless é, sem dúvida, inferior a esta margem, mas excederá o número de Clientes de Internet fixa em 2003 [OVUM]. Esse é um grande mercado que está surgindo, e que as operadoras telefônicas provavelmente não deixarão passar essa oportunidade.

Pensando nestes números, é possível reparar que o comércio eletrônico através de dispositivos móveis (*M-Commerce*) é um vasto mercado ainda inexplorado. De acordo com [Gartner], o valor estimado de transações através do *M-Commerce* estará por volta de \$1.8 trilhões de dólares em 2005 em todo o mundo.

Entretanto, quando o conceito é aplicado à prática nem sempre é tão simples como na teoria. Antes de realizar uma compra, é necessário garantir que a transação é segura e que não há risco de roubo de informação confidencial, tanto do lado do cliente quanto do lado do servidor; os processo dever funcionar sob limitações de largura de banda e hardware.

Devido a esses problemas, o uso dos Web Services pode ser uma excelente solução, gerenciando informações de vários tipos, originadas nos mais variados dispositivos. Mas devido ao fato de que os Web Services são uma tecnologia recente, é necessário provar que seu uso não implica nenhum risco de segurança nas transações eletrônicas, tanto para o Cliente quanto para o vendedor. Neste artigo, são discutidas algumas tecnologias existentes que possibilitam a segurança em transações eletrônicas em dispositivos móveis, e analisada a sua capacidade de interagir com o uso dos Web Services, de forma a garantir que é possível migrar com garantia de segurança de um sistema de comércio eletrônico tradicional para um que utilize Web Services, possibilitando assim a sua utilização por dispositivos móveis.

Na seção 2 será explicado o que é o comércio eletrônico móvel e quais são os seus desafios. Nas seções 3 e 4 serão apresentadas a arquitetura de um sistema de comércio eletrônico móvel, e discutidos os diferentes processos envolvidos em cada etapa de uma transação eletrônica, bem como; as formas de pagamento existentes no atual estágio tecnológico dos sistemas sem o uso dos Web Services; As tecnologias de segurança para comércio eletrônico são apresentadas na seção 5, onde se faz a análise da utilização no contexto da comércio eletrônico móvel utilizando Web Services. O artigo termina na seção 6, com a habitual síntese e trabalho futuro.

2. M-COMMERCE E SEUS DESAFIOS

Existem várias definições sobre o termo *M-Commerce*. Conforme [Profit 2001], *M-Commerce* significa "pagamento de bens ou serviços utilizando um dispositivo móvel", enquanto que para [Muller-Veerse 1999], "*M-Commerce* refere-se a qualquer transação com valor monetário que é conduzida através de uma rede de telecomunicações suportada em equipamentos móveis".

Olhando para essas definições, é possível reparar que o *M-Commerce* representa um subconjunto de todas as transações eletrônicas, tanto na área de *business-to-consumer* quanto *business-to-business*.

O comércio eletrônico móvel envolve o uso de dispositivos de computação móvel na concretização de diferentes tipos de transações eletrônicas, possibilitando-as de ocorrer em qualquer lugar ou hora [Lyytinen 2001].

Quando os Web Services são aplicados para uso no *M-Commerce*, é fácil perceber que seu papel é junto ao servidor, uma vez que os dispositivos móveis ainda não possuem poder de processamento suficiente para comportar a implantação de tal tecnologia. Fica assim claro que toda a informação relativa ao Cliente do negócio a realizar será enviada do dispositivo móvel para ser processada no servidor, e assim que os resultados forem conhecidos, a informação respectiva será transmitida novamente para o dispositivo móvel.

Ao desenvolver-se aplicações para *M-Commerce*, deve-se estar atento às seguintes limitações:

- Hardware
 - Menor poder de processamento das CPUs;
 - Baixa disponibilidade de memória (ROM e RAM);

- Telas pequenas;
- Diferentes dispositivos (telemóveis, PDAs, etc.);
- Redes de telefonia móvel
 - Menor largura de banda;
 - Maior latência;
 - Baixa estabilidade da conexão;
 - Baixa disponibilidade prevista.

3. ARQUITETURA DE UM SISTEMA DE M-COMMERCE

Quando se fala de *M-Commerce*, deve-se pensar em um sistema constituído pelos seguintes componentes:

- **Dispositivos móveis** – o equipamento portátil cliente em que corre um browser ou uma aplicação de *M-Commerce* personalizada (como por exemplo telemóveis e PDAs);
- **Centro de dados** - um WASP (Wireless Application Service Provider) executando o servidor de um sistema de *M-Commerce* em nome dos vendedores. O centro de dados é o coração dessa arquitetura. Ele orquestra as transações de *M-Commerce* que são originadas nos dispositivos móveis, emite pedidos de remessa aos vendedores, cobranças de cartões de crédito, etc. O centro de dados consiste tipicamente em um cluster de servidores mantendo as informações de transações em um banco de dados;
- **Autoridade de segurança** - fornece certificados de segurança e assina transações, etc;
- **Autoridade de verificação** – permite o centro de dados verificar as informações dos cartões de crédito, contas bancárias, etc., entregar aos Clientes suas respectivas compras e transferir fundos para os vendedores;

Por exemplo, quando o comprador acessa o endereço <http://www.lojavirtual.com>, o dispositivo móvel envia os pedidos para o centro de dados onde estão armazenadas as páginas da loja virtual. Ao receber os pedidos, o centro de dados analisa os requisitos e as informações necessárias e envia para a autoridade de verificação a identificação desse comprador. Por fim, o comprador se identifica para a autoridade de verificação e autoriza a transação. Todas as informações trocadas entre o comprador e a autoridade de verificação são realizadas através de um canal seguro. Quem garante a segurança no canal é a autoridade de segurança, que fornece um certificado que serve para confirmar a identificação do titular do certificado (nesse caso o servidor da autoridade de verificação) e o nível de segurança que está sendo utilizado nas páginas que estão a ser acedidas.

4. SISTEMAS DE PAGAMENTO VIRTUAL

Apesar de existirem muitas variedades de pagamento para operações originadas em

equipamentos móveis (conforme a Figura 1), a grande maioria possui as mesmas características. Em todos os casos, o Cliente que deseja realizar um pagamento necessita estar conectado com uma autoridade de verificação (operadora de telefonia móvel, banco ou outro), direta ou indiretamente. De acordo com [FPT 2001], o processo de pagamento virtual inicia-se com o vendedor contactando a autoridade de verificação. O vendedor informa à autoridade de verificação para contactar o Cliente ou encaminha o Cliente para o servidor da autoridade de verificação. A conexão é estabelecida usando canal de voz, SMS (Short Message Service) ou USSD (Unstructured Supplementary Services Data). Uma vez que a conexão é estabelecida, o Cliente autoriza o pagamento (geralmente fornecendo uma senha). Nessa hora, é estabelecido um canal seguro para o tráfego dos dados, de modo a garantir que as informações sigilosas não sejam interceptadas por terceiros. Mais tarde, o Cliente e o vendedor recebem uma confirmação do pagamento. O vendedor geralmente recebe os fundos na sua conta bancária. Para o Cliente, existe uma variedade de maneiras de promover a transferência de fundos. A autoridade de verificação pode debitar na conta bancária, cartão de crédito, conta pré-paga, ou agregam este pagamento junto a outros pagamentos numa conta a ser saldada pelo Cliente no final do mês.

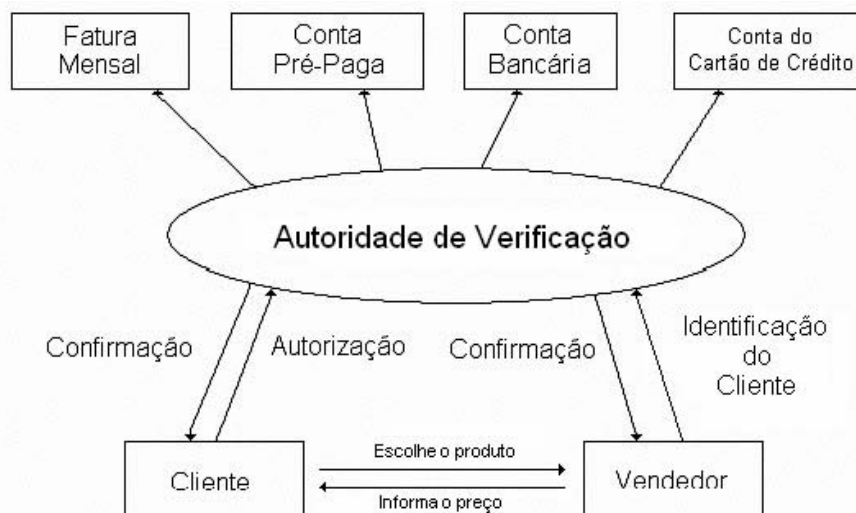


Figura 1 – Funcionamento de um sistema de pagamentos virtuais

5. TECNOLOGIAS DE SEGURANÇA PARA APLICAÇÕES MÓVEIS

Em toda transação realizada pela Internet existe tráfego de informação entre um cliente e um servidor. Se essa informação não é confidencial, a segurança não precisa ser muito rígida, porém, tratando-se de uma transacção de *M-Commerce* podem estar a ser transferidos dados pessoais como números de cartões de crédito e senhas de bancos, e nesse caso a rigidez da segurança é considerada um factor essencial. Ao fornecer os dados necessários para autorizar uma compra através da Internet, o Cliente deve ter garantias de segurança do tráfego dessas informações.

Actualmente, existem diferentes protocolos e tecnologias para garantir segurança e confidencialidade na Internet, e cada um possui maneiras diferentes de proteger a informação, como o uso de chaves públicas, algoritmos de criptografia e autenticação do cliente, entre outros. Ao realizar uma compra em uma loja virtual, o Cliente depara-se com

inúmeras tecnologias para proteger a transmissão dos seus dados. Obrigatoriamente existe a necessidade de proteger toda comunicação que chega e sai da autoridade de verificação, uma vez que ele é o responsável por tratar de informações sigilosas como senhas de contas bancárias e números de cartões de crédito.

Resumem-se nas subsecções seguintes os principais protocolos existentes mais vulgarmente usados para assegurar a confidencialidade.

5.1 WAP BASAEDO EM WTLS

O WTLS (Wireless Transport Layer Security) é um protocolo similar ao protocolo de segurança de transporte da Internet. Ele fornece autenticação, integridade dos dados e privacidade de serviços dentro das limitações dos hardwares utilizados na tecnologia wireless, como processamento, memória e largura de banda [Miller 2001]. Mas devido ao fato de que muitos sistemas de E-Commerce e sites corporativos usam segurança baseadas em SSL, uma transmissão de um site para um telefone WAP deve, primeiramente, atravessar um gateway que converte a encriptação de SSL para WTLS. Durante este processo de conversão entretanto, a mensagem é descodificada e assim está vulnerável a interceptação.

5.2 SSL

O SSL (Secure Socket Layer) [Netscape 1996] quando aplicado a pequenos dispositivos oferece autenticação, integração de dados e privacidade de serviços. Sua utilização nestes dispositivos não é recomendada, pois o protocolo é muito pesado comparando-se às fracas CPUs, baixas larguras de banda e alta latência das redes wireless. Além disso, os requisitos de memória estática e dinâmica são altos e, de acordo com [Evans e Dowloing 2002], o SSL não fornece uma segurança fim-a-fim. Em sistemas onde o servidor que armazena as informações sigilosas é diferente do servidor onde está o web site, a vulnerabilidade é alta.

5.3 XML SIGNATURE

A XML Signature [IETF e W3C 2000] é uma iniciativa conjunta da IETF e do W3C para especificar uma sintaxe XML e regras de processamento para criação e representação digital de assinaturas. As vantagens na utilização da XML Signature, ao invés de outros padrões de assinaturas digitais, estão baseadas na independência de linguagem de programação, fácil interpretação humana e independência de fabricante. Esta tecnologia também permite assinar digitalmente subconjuntos de um documento XML, o que permite um número maior de benefícios para os vários objectivos em *e-business*.

5.4 XML ENCRYPTION

A XML Encryption [IETF e W3C 2002] especifica um processo para encriptação de dados e sua representação em formato XML. Os dados podem ser: dados arbitrários (incluindo um documento XML), elementos XML, ou conteúdos de elementos XML. Um documento XML que utiliza a XML Encryption pode ser visto por qualquer utilizador, mas apenas o proprietário da chave de descodificação conseguirá compreender o conteúdo do que foi criptografado.

5.5 XKMS

O XKMS (XML Key Management Specification) [Verisign, et al. 2001] fornece um protocolo baseado em XML/SOAP para distribuição e registro de chaves públicas. Ele inclui funções para informações sobre chaves, registro, verificação e revogação. O XKMS é destinado a fornecer suporte para gerenciamento de chaves para tecnologias como XML Signature e XML Encryption, mas também pode ser utilizado para suportar outras tecnologias de chaves públicas.

5.6 WS-SECURITY

O WS-Security (Web Services Security) [Verisign, et al. 2002] é uma iniciativa conjunta de empresas como Microsoft, IBM e Verisign destinada ao uso da XML Signature e da XML Encryption para fornecer segurança a mensagens SOAP. Esta iniciativa será proposta ao W3C para padronização em seu devido tempo. O WS-Security é um esforço destinado a fazer com que os Web Services trabalhem melhor em um ambiente global. O WS-Security também inclui alguns importantes componentes como roteamento, confiabilidade e tratamento de transações.

5.7 SAML

O SAML (Security Assertion Markup Language) [OASIS 2001] é um padrão emergente para a troca de informação sobre autenticação e autorização. O SAML soluciona um importante problema para as aplicações da próxima geração, que é a possibilidade de utilizadores transportarem seus direitos entre diferentes Web Services. Isto é importante para aplicações que intencionam integrar um número de Web Services para formar uma aplicação unificada.

5.8 COMPARAÇÃO ENTRE AS DIFERENTES TECNOLOGIAS

Uma vez que todas as tecnologias de segurança, citadas anteriormente, podem ser utilizadas com os Web Services, é necessário que sejam conhecidas suas características, de modo a ajudar na escolha do protocolo mais apropriado à arquitetura de cada sistema. Na tabela comparativa abaixo identificam-se as de cada uma das 7 abordagens referentes, de acordo com os 7 critérios que constam nessas linhas. A tabela permite visualizar a melhor opção para a utilização com os Web Services.

Uma vez que dispositivos móveis possuem baixa largura de banda e processamento limitado, deve-se estar atento quanto às necessidades de segurança e as vantagens que a utilização que cada tecnologia pode trazer.

De uma forma geral, as tecnologias padrões como SSL e WTLS possuem os requisitos necessários para fornecer segurança entre um cliente e um servidor, mas de uma forma limitada, pois ambos fornecem segurança ponto-a-ponto. No caso de uma loja virtual simples, onde todas as informações são tratadas no próprio servidor, isso é mais do que suficiente. Mas a utilização do SSL traz dificuldades adicionais pelo o fato de que é necessário que o programador tenha bons conhecimentos sobre protocolos de comunicação para implementar a segurança, além de que a solução não é independente de plataforma

pois necessita de software específico.

	WAP/WTLS	SSL	XML Signature	XML Encryption	XKMS	WS Security	SAML
Uso de XML	Não	Não	Sim	Sim	Sim	Sim	Sim
Independência de plataforma	Não	Não	Sim	Sim	Sim	Sim	Sim
Suporte a chaves PKI	Sim	Sim	Sim	Sim	Sim	Sim	Sim
Segurança fim-a-fim com Web Services	Não	Não	Sim	Sim	Sim	Sim	Sim
Integridade	Sim	Sim	Sim	Sim	Sim	Sim	Não
Autenticação da mensagem	Sim	Sim	Sim	Não	Sim	Sim	Não
Autenticação do Cliente	Sim	Sim	Sim	Não	Sim	Sim	Sim

Tabela 1 – Características das tecnologias de segurança

Já no caso dos protocolos baseados em XML, a configuração é feita de uma forma simples, uma vez que é simples de compreender o que está descrito no conteúdo dos documentos; além do mais, devido ao XML ser um padrão aberto, é independente de plataforma. O inconveniente dos protocolos baseados em XML é a grande quantidade de informação que deve ser transmitida. Como não é possível realizar um “parsing” das informações antes de as transmitir, pois seria necessário uma configuração prévia e a instalação de software adicional no dispositivo móvel, o Cliente acaba tendo que pagar mais caro pelo tráfego dos dados adicionais contidos no protocolo.

5.9 USO DAS TECNOLOGIAS DE SEGURANÇA COM WEB SERVICES

Conforme a necessidade de cada solução e sua respectiva arquitectura, diferentes tecnologias podem ser utilizadas. Para a primeira geração de comércio electrónico, o uso do SSL era suficiente (e ainda é, em alguns casos), mas se os requisitos das aplicações aumentam, outras tecnologias devem ser aplicadas, para que o nível de segurança aumente.

Ao pensar-se no uso de Web Services destinado a aplicações móveis, deve-se ter em mente que esses serviços estarão localizados dentro de um servidor, e nunca no dispositivo móvel devido às suas limitações. Dessa forma, soluções que não garantam uma segurança fim-a-fim não são uma solução recomendada, uma vez que os Web Services se caracterizam por disponibilizar métodos e rotinas para acesso remoto, esses métodos não necessariamente necessitam estar no mesmo servidor.

Em uma arquitectura onde se utilizam Web Services com um protocolo de segurança que não oferece segurança fim-a-fim, os dados serão decodificados e codificados novamente em todos os intermediários entre o cliente e o servidor. É justamente nesse momento onde existe o risco de segurança, pois é possível que a informação seja armazenada no servidor Web.

Ao analisarmos uma arquitectura onde também se utilizam Web Services, mas com segurança fim-a-fim, verificamos que o problema do tráfego de informação confidencial é contornado, uma vez que só há a descodificação dos dados originais no destino final, e não em seus intermediários. Com isso, as aplicações do *M-Commerce* ganham em segurança e em tempo de resposta.

6. CONCLUSÃO

A utilização dos Web Services para suportar aplicações destinadas a dispositivos móveis possibilitará o aumento da oferta de novos serviços, tornando possível aceder qualquer tipo de dados em qualquer dispositivo e em qualquer lugar.

O *M-Commerce* é designado para plataformas com limitações de hardware. Dentre todas as opções de segurança citadas, que podem ser utilizados com Web Services, o uso de protocolos baseados em XML é a mais recomendada devido a compatibilidade com os Web Services (que utilizam essa tecnologia). Além disso, o XML é um padrão internacional, aberto, para a troca de dados, sendo garantido o funcionamento em diferentes plataformas; é ainda interessante a ideia de transmitir várias informações diferentes de forma segura em um único documento, diminuindo o tempo de resposta e o tráfego. Além disso, técnicas de segurança baseadas em XML fornecem uma proteção dos dados fim-a-fim (diferentemente de técnicas nãoXML) que podem ser utilizadas com Web Services, possibilitando uma comunicação segura entre o cliente e o servidor.

Antes de se criar Web Services para permitir que dispositivos móveis possam realizar compras pela Internet a partir de uma loja virtual, é necessário verificar se a arquitetura do sistema de comércio eletrônico é compatível com a utilização dos Web Services, e conforme a resposta, se existe a necessidade do uso dessa tecnologia. Em conclusão, pode-se afirmar que os Web Services podem ser considerados com técnicas seguras, tornando-os confiáveis para serem utilizados no *M-Commerce*.

Na sequência deste trabalho prévio de pesquisa, serão desenvolvidos estudos experimentais em casos práticos para tirar conclusões quanto à eficácia das várias hipóteses levantadas.

REFERÊNCIAS

[Netscape 1996] Netscape Corporation, 1996. The ssl protocol. Available at <http://wp.netscape.com/eng/ssl3/draft302.txt>.

[Evans e Dowloing 2002] Sarah Evans and Olwyn Dowling, 2002. Is ssl enough security for first-generation web services? Available at <http://www.webservices.org/index.php/article/articleview/529>

[FPT 2001] Institute for Prospective Technological Studies, 2001. The future of m-payments - business options and policy issues. Available at <http://epso.jrc.es/Docs/Backgrnd-2.pdf>.

[OASIS 2001] Organization for the Advancement of Structured Information Standards, 2001. SAML specification. Available at <http://lists.oasis-open.org/archives/securityservices/200106/pdf00002.pdf>.

[Cahners 2000] Cahners InStat Group. abstract of wireless data/internet market gains momentum: Five-year subscriber forecast. Available at <http://www.instant.com/pr/2000/md0003md pr.htm>.

[OVUM] OVUM Group. Ovum forecast numbers. Available at http://www.cellular.co.za/technologies/mobile3d/visa_mobile_3d.htm.

[Gartner] The Gartner Group. The shape of wireless economy. Available at <http://www.dataquest.com/public/static/pressrel/MobileRelease.html>.

[IETF e W3C 2000] IETF/W3C, 2000. XML-Signature syntax and processing (working draft). Available at <http://www.w3.org/TR/2000/WD-xmlsigcore-20001012>.

[IETF e W3C 2002] IETF/W3C, 2002. XML-Encryption syntax and processing (working draft). Available at <http://www.w3.org/TR/2002/CR-xmlenccore-20020304>.

[Lyytinen 2001] Kalle Lyytinen, 2001. M-commerce -mobile commerce: a new frontier for e-business. *Proceedings of the 34th Hawaii International Conference on System Sciences*, Honolulu, United States, Volume 9.

[Miller 2001] S. K. Miller. Facing the challenge of wireless security. *IEEE Computer*, pages 16–18, July 2001.

[Muller-Veerse 1999] F. Muller-Veerse, 1999. Mobile commerce report. Durlacher Corp. Available at <http://www.durlacher.com/downloads/mcomreport.pdf>.

[Profit 2001] The Profit, 2001. Mobile commerce - will it or won't it? Available at <http://www.portal.com/markets/wireless/Profit2001.pdf>.

[Verisign, et al. 2001] webMethods, Verisign, Microsoft, 2001. Xml key management specification (xkms) draft version

1.1. Available at <http://www.verisign.com/resources/gd/xml/xkms/xkmsv1-1.pdf>.

[Verisign, et al. 2002] Microsoft, Verisign e IBM, 2002. Web services security (ws-security). Available at <http://www.verisign.com/wss/wss.pdf>.

MULTIMÉDIA E INTERACÇÃO

Reconhecimento Automático de Fala Contínua em Português Europeu Recorrendo a Streams Audio-Visuais

Filipe Sá, Pedro Afonso, Ricardo Ferreira, Vítor Pêra

Faculdade de Engenharia da Faculdade do Porto,
Laboratório de Processamento da Fala, Electroacústica, Sinais e Instrumentação
sa@fe.up.pt, pedro.afonso@fe.up.pt, ricardo.ferreira@fe.up.pt, vpera@fe.up.pt

ABSTRACTO

Neste artigo é apresentado um reconhecedor de fala contínua, para uma aplicação dependente do falante em português europeu e com vocabulário de alfanuméricos, que utiliza informação audio-visual no processo de descodificação. A principal motivação para introduzir características visuais reside na insensibilidade destas às perturbações que afectam o sinal acústico, robustecendo assim o sistema em ambientes acústicos não controlados. Os resultados que aqui se apresentam demonstram que a inclusão do *stream* visual num formalismo estatístico *multi-stream* síncrono permite a redução da taxa do erro, em aproximadamente 56%_{rel}, numa grande gama de valores da relação sinal-ruído.

1. INTRODUÇÃO

O progresso da tecnologia do reconhecimento automático de fala tem permitido alcançar níveis de desempenho satisfatórios em algumas aplicações interessantes. No entanto, a falta de robustez destes sistemas a perturbações que afectam o sinal acústico continua a ser uma limitação muito grave. Este facto tem conduzido a uma acentuada investigação relacionada com o problema da robustez, tendo surgido abordagens muito diversas. Algumas vertentes desta investigação baseiam-se na extracção de características acústicas mais robustas e no aperfeiçoamento dos modelos acústicos. Outra abordagem, que se tem revelado muito promissora, assenta na utilização de características com um suporte físico de natureza não acústica e imune às perturbações no canal acústico. Esta é a abordagem seguida no estudo que aqui se apresenta, com os sistemas por nós desenvolvidos baseando o processo de reconhecimento numa combinação adequada da informação discriminante associada às características acústicas e visuais extraídas do sinal vídeo da face do indivíduo falante.

Para o desenvolvimento destes sistemas foi necessário criar uma base de dados contendo o registo dos sinais acústico e vídeo correspondentes a frases pronunciadas de maneira contínua no português europeu. A opção de efectuar um estudo experimental nesta língua deve-se em grande medida ao facto de não existirem, tanto quanto é do conhecimento dos autores, estudos realizados anteriormente e publicados sobre a utilização de informação visual em sistemas de reconhecimento de fala para aplicações no português europeu. Na secção 2 é apresentada a estrutura e a informação mais importante relativa a esta base de dados.

Na generalidade dos sistemas de reconhecimento multi-modais, a extracção das características acústicas baseia-se em técnicas comuns aos sistemas convencionais. Essa

foi também a solução seguida nos sistemas referidos neste trabalho, tendo-se optado pelos coeficientes Mel-cepstrais. Em relação à extracção das características visuais, não existem por enquanto métodos *standard*, verificando-se antes uma forte investigação a este respeito. Também nós neste trabalho investigámos diversas técnicas, tendo finalmente optado por basear a representação da informação visual discriminante através da transformada discreta do coseno da imagem da região da boca do falante. Na secção 3 são descritos os aspectos mais importantes sobre a análise acústica e visual efectuada pelos sistemas de reconhecimento desenvolvidos.

Uma questão essencial no reconhecimento baseado em múltiplos *streams* de características consiste na escolha do método para integrar essa informação de maneira a obter uma decisão final óptima. Existem diversas abordagens, desde a simples concatenação dos vectores de características correspondentes aos vários *streams* num único vector até ao processamento completamente independente de cada *stream* e a posterior combinação das respectivas hipóteses de reconhecimento conduzindo a uma decisão final. A abordagem que aqui se apresenta baseia-se no formalismo *multi-stream* [Bourlard, et al. 1996], uma solução intermédia que aproveita as vantagens resultantes da existência de modelos acústicos e visuais independentes e ao mesmo tempo não perde totalmente a informação discriminante devida à correlação entre os *streams*. A secção 4 descreve os aspectos essenciais do módulo de descodificação desenvolvido.

Os resultados que aqui são apresentados demonstram, numa aplicação em fala contínua em português europeu, um grande aumento da robustez ao ruído ambiente combinando simultaneamente as características acústicas e visuais extraídas. Relativamente ao sistema base, utilizando apenas o *stream* acústico e com praticamente ausência de ruído (SNR=40dB), a taxa de erro no reconhecimento de palavras (WER) diminui em aproximadamente 62%_{rel}. Para SNR no intervalo de 5dB a 40dB, essa redução mantém-se aproximadamente no valor 56%_{rel}. A secção 5 apresenta os resultados mais significativos obtidos neste trabalho experimental.

Finalmente, algumas conclusões deste estudo encontram-se destacadas na secção 6.

2. BASE DE DADOS

Contrariamente ao que sucede noutras línguas ([Pigeon97], [MESSER99], [Cheok, et al. 2001]), como são os casos do inglês, francês e alemão, até ao desenvolvimento deste trabalho não estava disponível qualquer base de dados audio-visual para o desenvolvimento de sistemas de reconhecimento automático no português europeu. O interesse da tecnologia multi-modal está já experimentalmente demonstrado em algumas línguas mas não ainda no caso particular do português. Além de ser importante tentar confirmar as expectativas favoráveis para o português, é também muito importante estimar se a eficácia desta abordagem é ou não relevante neste caso. Basta ter presente as diferenças acentuadas, entre as várias línguas, que caracterizam a expressão visual do processo de produção da fala, para compreender que não é legítima a simples extrapolação de resultados de uma língua para outra qualquer. Por exemplo, os conjuntos de *visemas* definidos nas várias línguas apresentam diferenças

significativas. Assim, e uma vez que nenhuma Base de Dados (BD) audio-visual falada em Português Europeu existia na altura do desenvolvimento do projecto, foi necessário criá-la.

Esta BD foi desenhada para uma aplicação em fala contínua e dependente do falante. Cada frase é constituída por um número de telefone, seguido de um nome próprio e de um apelido. O número de telefone é pronunciado como uma sequência de nove dígitos, pertencentes ao conjunto de dígitos em português. O nome e o apelido são soletrados, pertencendo as letras ao alfabeto português actual. O nome e o apelido encontram-se separados pela palavra 'espaço'.

A BD é composta por um conjunto de 419 frases, com uma média de 22 caracteres alfanuméricos em cada frase. Esta BD encontra-se dividida em 2 partes: 300 frases são utilizadas para treino e desenvolvimento; as restantes 119 frases são utilizadas para teste.

Para as frases do conjunto treino e desenvolvimento está disponível a informação da segmentação, definida ao nível da palavra, e a respectiva transcrição ortográfica. Foi gerada e está disponível na BD uma gramática estatística, do tipo Bi-Gram. Não está disponível qualquer léxico baseado em unidades elementares a um nível sub-palavra.

Em relação ao sinal vídeo, foi capturado em formato DV (Digital Vídeo) com dimensões de 720x576 *pixels* a 25 frames por segundo. Este vídeo foi depois comprimido no formato MPEG-4, devido à sua alta taxa de compressão e elevada qualidade. Consegue-se assim reduzir para um rácio de 1/10 do tamanho original, capturado em DV. A BD final obteve uma relação aproximada de 2/1 - Vídeo/Áudio o que é excelente, visto o comum ser o contrário ou mesmo superior.

Na gravação do sinal áudio não foi utilizado qualquer tipo de compressão o que facilita o processo de reconhecimento, para além de ser um padrão usado em inúmeras Bases de Dados. O formato adoptado foi então o 16 bit PCM. A gravação foi efectuada em ambiente laboratorial, tendo sido obtida uma relação de sinal-ruído de aproximadamente 40dB.

3. EXTRACÇÃO DE CARACTERÍSTICAS

3.1 Métodos de localização e extracção da região da boca

São referidos na literatura diversos tipos de abordagens que, em regra geral, podem ser agrupadas em duas classes: orientadas ao modelo, na geometria ou no contorno dos lábios, e orientadas ao pixel. Na primeira abordagem, os contornos dos lábios do falante são extraídos da sequência de imagens através de métodos como contornos activos [Kass, et al. 1998], triangulação com os olhos e o nariz [Stiefelhausen, et al. 1997], *templates* deformáveis [Hennecke, et al. 1994]. Um modelo paramétrico ou estatístico dos contornos dos lábios é assim obtido. Na segunda abordagem, são efectuadas transformações apropriadas dos valores dos *pixels* (exclusão de vermelho [Lewis, et al.

2000], fluxo óptico [Iwano, et al. 2001]) em cada frame. Estas transformações têm como objectivo potenciar eventuais diferenças entre a região labial e as restantes por forma a que a detecção dos lábios se torne um processo mais robusto.

3.2 Métodos de extracção de características visuais

De forma semelhante à efectuada para localizar a boca, a extracção de características visuais pode ser feita usando uma abordagem orientada ao modelo ou orientada ao pixel. Num método orientado ao modelo, os parâmetros do modelo dos lábios obtido anteriormente são usados como características visuais. Este tipo de abordagem tem a vantagem de que o número de características visuais é grandemente reduzido, potencialmente acelerando o processamento subsequente, reduzindo a variabilidade e aumentando a generalização. No entanto, existem diversos aspectos negativos. O processo de encontrar um modelo para cada frame pode ser moroso e, mais importante, o modelo pode excluir informação linguística relevante. Precisamente, uma vantagem de se usar uma abordagem orientada ao pixel é o facto de informação linguística relevante não ser excluída uma vez que toda a região de interesse da boca é considerada. Como ponto negativo salienta-se o facto deste método necessitar de mais poder de processamento do que o anterior o que o torna desaconselhado para sistemas de reconhecimento a operar em tempo-real.

3.3 Módulo de extracção de características visuais

O primeiro passo para a extracção de características visuais consiste na localização da região da boca para cada frame de imagem de vídeo. Após termos pesquisado e analisado os métodos usados por outros investigadores decidimos utilizar uma abordagem própria para a questão da definição de uma Região de Interesse (“Region of Interest – ROI”) em torno dos lábios. Para localizar esta região tentamos aplicar o método de exclusão de vermelho [Lewis, et al. 2000]. Logo verificamos que para a nossa BD este método não era muito eficaz, obrigando-nos a mudar de estratégia. Desse modo optamos por um método similar ao anterior, mas neste caso efectuamos a exclusão do azul. Achamos que este método seria mais eficaz na diferenciação dos lábios, tendo-se obtido, como podemos observar na figura 1, um elevado contraste entre a região de interesse e a face.



Figura 1: Imagem da Região de Interesse (ROI) com exclusão de azul:
(a) boca fechada
(b) boca aberta.

Através de um varrimento desta imagem de exclusão de azul, procuramos os pontos extremos dos lábios: inferior, superior, esquerdo e direito, de modo a poder definir a área de interesse. A figura seguinte (figura 2) mostra os pontos encontrados para localização da caixa sobre a qual irão ser extraídas as características.

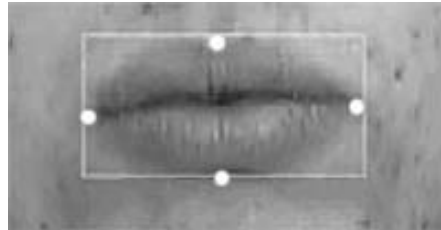


Figura 2: *Determinação dos pontos extremos dos lábios com base na região de interesse.*

A intenção inicial consistia na determinação de características baseadas apenas no contorno dos lábios. Verificou-se no entanto que esse processo excluiria informação discriminativa importante relacionada com o posicionamento da língua e dos dentes. Desse modo optamos por utilizar a transformada discreta do coseno (DCT2) [Potamianos, et al. 2001], a qual seria aplicada a uma janela limitada pelos pontos extremos anteriormente encontrados. Dado que esta janela varia devido ao movimento da boca, optou-se por normalizá-la resultando uma imagem com as dimensões 32x32 *pixels*. Através da aplicação da DCT2 e, em seguida, de uma máscara adequada (ver figura 3), a representação da informação associada a esta imagem é fortemente comprimida resultando um vector de características com dimensão 26. Esta máscara tem como objectivo escolher os coeficientes de maior relevância resultantes da DCT2, correspondendo aos que melhor caracterizam a imagem. Um vector de características é extraído cada 40ms, tendo sido necessário efectuar um processo de interpolação para obter um vector em cada 10ms, obtendo-se assim um *frame-rate* igual a do *stream* áudio.

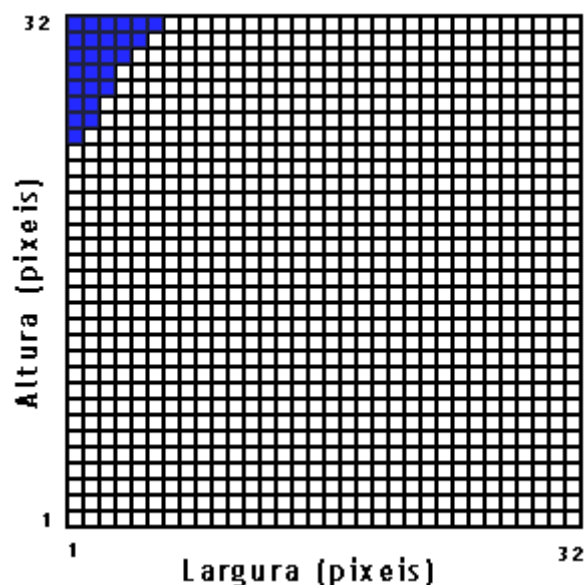


Figura 3: *Máscara aplicada ao resultado da DCT.*

3.4 Módulo de extracção de características acústicas

Ao contrário da extracção das características vídeo, a investigação dos métodos de extracção das características áudio está numa fase muito mais avançada. Podem mesmo apresentar-se diversos métodos *standard* em que se baseiam a generalidade dos reconhecedores automáticos de fala. Para o nosso reconhecedor decidimos usar um desses métodos de representação, os coeficientes Mel Cepstrais (MFCC) [Young, et al. 1993].

Uma vez que o espectro do sinal para frequências superiores a aproximadamente 8KHz não encerra informação discriminativa significativa para efeitos do reconhecimento de fala, o sinal áudio disponível na BD foi submetido a um filtro passa-baixo com este valor de frequência de corte. O sinal resultante foi então sub-amostrado, passando da frequência inicial de 44.1kHz para 16kHz.

Depois de aplicado o filtro de pré-ênfase, o sinal foi segmentado em janelas Hamming com duração 25ms e sobreposição 15ms. Assim, em cada 10ms é gerado um vector de características. Este vector tem dimensão 26, sendo composto por 12 MFCCs, um termo com a energia do sinal na respectiva *frame*, e os elementos correspondentes às primeiras derivadas.

4. MÓDULO DE DESCODIFICAÇÃO

Neste trabalho experimental foram desenvolvidos dois tipos de sistemas de reconhecimento: os sistemas *single-stream*, usando apenas o *stream* acústico ou o *stream* visual, e os sistemas *multi-stream*, usando ambos os *streams* em simultâneo.

O módulo de decodificação nos sistemas *single-stream* assenta numa abordagem *standard* utilizando o algoritmo Viterbi implementado através de *tokens*, o qual aproxima de maneira eficaz o paradigma de pesquisa óptimo. Na modelação acústica, ou visual, são usados modelos elementares sem contexto definidos ao nível da palavra. São utilizados modelos escondidos de Markov (HMM) [Rabiner, 1989], do tipo semi-contínuo (SC-HMM) para modelar cada palavra. Em cada estado a verosimilhança é calculada combinando linearmente as 128 funções de Gauss definidas no *codebook*. O número de estados é dependente da palavra, sendo naturalmente maior ou menor em função da complexidade acústica, ou visual, da palavra. Na modelação linguística é usada a gramática Bi-Gram disponibilizada pela BD.

Em relação ao módulo de decodificação nos sistemas *multi-stream*, foram equacionadas diversas hipóteses alternativas. Uma solução frequentemente seguida, devido principalmente à sua simplicidade, consiste em concatenar os vectores de características acústicas e visuais resultando um único *stream* de vectores, que é então processado como tal. Uma solução no extremo oposto, no que respeita ao momento onde as informações associadas aos dois *streams* são recombinadas, consistiria em processar os *streams* de maneira completamente independente obtendo assim hipóteses

de reconhecimento também independentes. Estas hipóteses, ou listas de hipóteses, seriam então processadas para se obter a decisão final.

Apesar dos resultados interessantes que têm sido obtidos por qualquer um destes métodos, optou-se por uma abordagem diferente, a qual apresenta características promissoras relativamente aos acima referidos. Resumidamente, este método apresenta as vantagens de permitir a modelação independente dos *streams* acústico e visual e ao mesmo tempo não exclui completamente informação discriminante importante devido à correlação destes *streams*. Este método baseia-se no formalismo *multi-stream* tal como é apresentado em [Bourlard, et al. 1996]. Os resultados que aqui se apresentam referem-se apenas à implementação síncrona deste método, estando actualmente em curso a realização de experiências conducentes à obtenção da versão assíncrona que permitirá, se as expectativas se confirmarem, obter resultados mais interessantes ainda.

Assim, o módulo de decodificação desenvolvido para os sistemas *multi-stream* assenta numa abordagem semelhante à utilizada nos sistemas *single-stream*, existindo apenas uma diferença essencial que reside na maneira como as verosimilhanças são calculadas em cada estado. Os valores da verosimilhança relativos a cada *stream* são calculados separadamente, sendo então combinados linearmente. A influência local de cada *stream* pode ser definida ajustando valor do respectivo coeficiente, mantendo-o no entanto constante para todas as palavras. Os resultados que aqui se apresentam são obtidos optimizando em função da relação sinal-ruído o valor destes coeficientes, utilizando para isso parte do conjunto de treino.

5. RESULTADOS

Esta secção está dividida em 2 partes: Resultados *Single-Stream* e Resultados *Multi-Stream* Síncrono.

5.1 Sistemas *Single-Stream*

Na tabela 1 mostramos os resultados obtidos para o reconhecimento áudio e vídeo obtido para a relação sinal-ruído original (SNR=40dB), em duas tarefas de reconhecimento diferentes. Numa das tarefas pretende-se reconhecer uma sequência de 9 dígitos que compõe o número de telefone. A outra tarefa consiste no reconhecimento da frase completa, constituída pelo número de telefone seguido do nome próprio e do apelido (ver secção 3.1).

Tabela 1: Resultados *Single-Stream*.
Taxa de Erro (Word Error Rate - WER) e Frases Certas (Sentence Accuracy - SA).

Tipo Reconhecimento /Tipo Frase	WER (%)	SA (%)
Áudio / 9 Dígitos	0.19	99.16
Áudio / Frase Completa	3.25	52.10
Vídeo / 9 Dígitos	15.59	36.97
Vídeo / Frase Completa	31.96	0.00

Na tabela 2 encontram-se representados os valores da taxa de erro apenas para o decodificador áudio para diferentes relações sinal-ruído na tarefa de reconhecimento da frase completa.

Tabela 2: *Resultados Single-Stream áudio para diversas relações sinal-ruído.*

SNR (dB)	WER (%)
5	63.30
10	53.04
15	45.09
20	31.77
25	14.53
30	12.69
40 (original)	3.25

5.2 Sistemas *Multi-Stream*

No reconhecedor *multi-stream*, a complementaridade das informações associadas aos *streams* áudio e vídeo permite a redução da taxa do erro quando o valor de SNR é elevado, correspondendo ao ambiente acústico verificado durante a gravação da BD, e também quando o canal acústico é afectado com ruído aditivo do tipo gaussiano. Os resultados que a seguir se apresentam referem-se à tarefa do reconhecimento da frase completa.

Na tabela 3 podemos ver os resultados *multi-stream* obtidos para a relação sinal-ruído original. Podemos observar aqui a melhoria da taxa de reconhecimento quando os dois streams são combinados com um determinado valor na relação áudio/vídeo. O peso atribuído aos *streams* correspondente aos valores 0.93/0.07 foi afinado usando 100 frases pertencentes ao conjunto de treino e desenvolvimento. As taxas apresentadas aqui foram obtidas com as frases de teste, nunca usadas até então.

Tabela 3: *Resultados Multi-Stream para diferentes pesos relativos Áudio-Vídeo.*

Peso Áudio	Peso Vídeo	WER (%)
1.00	0.00	3.25
0.93	0.07	1.25
0.00	1.00	31.96

De seguida apresentamos resultados do nosso reconhecedor, adicionando previamente ruído branco às faixas de áudio da base de dados, para simular uma situação real em que exista ruído ambiente. Tal como no caso anterior, a afinação dos pesos Áudio/Vídeo foi obtida usando 100 frases pertencentes ao conjunto de treino e desenvolvimento, deixando as frases de teste apenas para a avaliação final. O valor da relação sinal-ruído original das frases tem o valor aproximado de 40dB.

Tabela 4: Resultados Multi-Stream para diferentes valores de SNR.

Peso Áudio	Peso Vídeo	SNR (dB)	WER %
0.93	0.070	40 (original)	1.25
0.925	0.075	30	4.11
0.93	0.070	25	6.00
0.75	0.250	20	15.88
0.45	0.550	15	23.92
0.1	0.900	10	27.77
0.05	0.950	5	28.80

O gráfico seguinte apresenta as curvas de desempenho, obtidas pelos reconhecedores construídos, em função do valor da relação sinal-ruído.

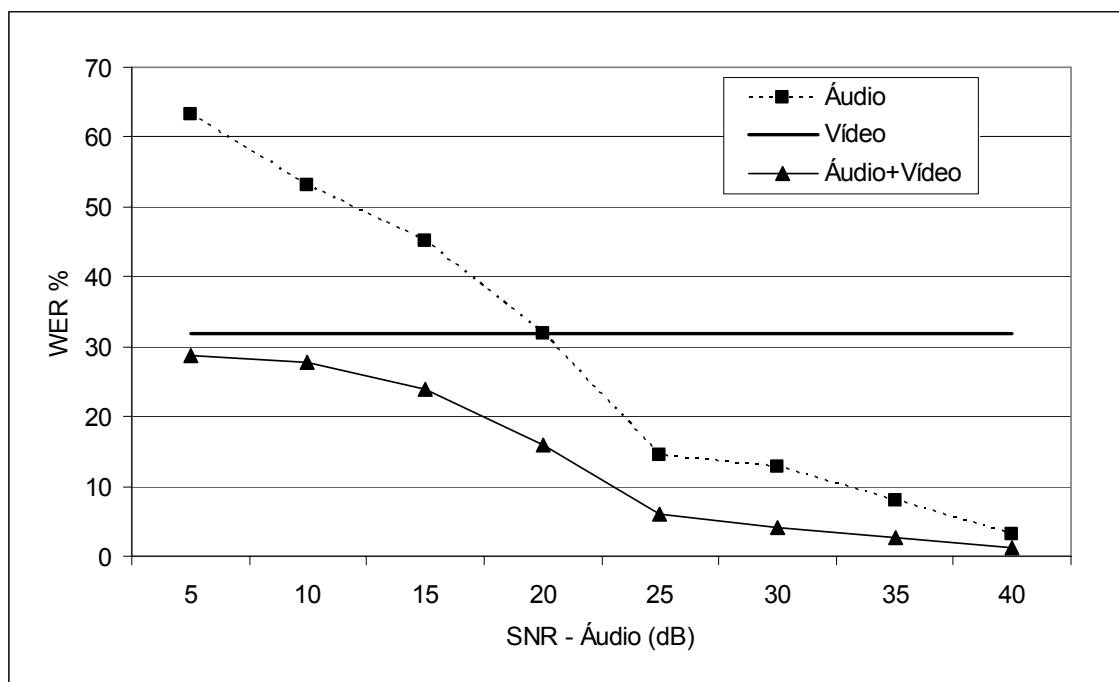


Figura 4: Evolução da taxa do erro para diferentes valores de SNR e reconhecedores: single-stream áudio, single-stream vídeo e multi-stream.

Como a figura 4 mostra, o reconhecedor baseado apenas nas características áudio apresenta uma acentuada degradação do desempenho, sobretudo para valores de SNR inferiores a 25dB. Com a inclusão do *stream* vídeo no processo de reconhecimento, verifica-se uma melhoria substancial na taxa de erro. Esse bom desempenho equivale a um claro ganho na relação SNR: com o decodificador *multi-stream* obtivemos um ganho efectivo nunca inferior a 5dB para taxas de erro iguais quando comparado com o

reconhecedor *single-stream* áudio, chegando a valores acima de 10dB (para um WER de 4%). O uso de características visuais não torna apenas o sistema mais insensível a perturbações externas. Apesar de os reconhecedores baseados exclusivamente em *streams* acústicos apresentarem taxas de erro reduzidas em situações de quase ausência de ruído, a inclusão de *streams* visuais possibilita que essa taxa de erro diminua consideravelmente. Para a relação sinal-ruído original (aproximadamente 40dB) a taxa de erro diminuiu de 3.25% para 1.25% o que significa um ganho relativo de 62%_{rel}.

6. CONCLUSÕES

Os resultados obtidos com este trabalho demonstram experimentalmente que a utilização simultânea de características acústicas e visuais constitui uma abordagem muito eficaz para aplicações de reconhecimento de fala contínua em português europeu. Essa eficácia traduziu-se, nos resultados aqui apresentados, por uma acentuada redução da taxa do erro ao nível da palavra (WER): numa situação de quase inexistência de ruído, esse ganho foi estimado em aproximadamente 62%_{rel}; em situações de ambiente acústico não controlado, com a relação sinal-ruído variando no intervalo de 5dB a 40dB, verificou-se um decréscimo da taxa do erro entre os valores 47%_{rel} (para um SNR de 15dB) e 68%_{rel} (para um SNR de 30dB). Interessa sublinhar a demonstração feita da grande relevância desta tecnologia em aplicações onde o problema da robustez é crítico.

7. REFERÊNCIAS

- [Bourlard, et al. 1996] Bourlard H., Dupont S., and Ris C., “Multi-Stream Speech Recognition”, *IDIAP-RR 96-07*, 1996
- [Cheok, et al. 2001] Cheok, A., National University of Singapore, DSTA Report for project on Multi-Modal Speech Recognition: developing the first Asian audio-video corpus, *World Wide Web*, www.ece.nus.edu.sg/stfpage/eleadc/dsta/report-DSTA-D1-v2a.pdf
- [Hennecke, et al. 1994] Hennecke M., Prasad V., and Stork D., “Using deformable templates to infer visual speech dynamics”, *28th Annual Asimolar Conference on Signals, Systems, and Computer: vol 2 pp. 576-582*, Pacific Grove, CA. IEEE Computer, 1994
- [Iwano, et al. 2001] Iwano K., Tamura S. and Furui S., “Bimodal Speech Recognition using Lip Movement Measured by Optical-Flow Analysis”, 2001
- [Kass, et al. 1998] Kass M., Witkin A., and Terzopoulos D., “Snakes: active contour models”, *International Journal of Computer Vision: vol 1 pp. 321-332*, 1998
- [Lewis, et al. 2000] Lewis T. and Powers D., “Lip Feature Extraction Extraction using Red Exclusion”, *World Wide Web*, www.cs.usyd.edu.au/~vip2000 , 2000
- [MESSER99] Messer K., Matas J., Kittler J., Luetttin J., Maitre G., "XM2VTSDB: The Extended M2VTS Database.", Second International Conference on Audio and Video-based Biometric Person Authentication (AVBPA'99), Washington D.C.,1999
- [Potamianos, et al. 2001] Potamianos G. and Neti C., “Automatic Speechreading of Impaired Speech”, *Proceedings of the International Conference on Auditory-Visual Speech Processing*, pp. 172-182, Aalborg, 2001
- [Pigeon 1997] Pigeon, S. and Vandendorpe, L., “The M2VTS multimodal face database (release 1.00)”, *Proceedings of the First International Conference on Audio- and Video-based Biometric Person Authentication, Crans-Montana, Switzerland, 1997*
- [Stiefelhagen, et al. 1997] Stiefelhagen R., Yang J., and Meier U., “Real-time lip tracking for lip reading”, *Proceedings of Eurospeech 97*, 1997
- [Young, et al. 1993] Young, S.J., Woodland, P.C. and Byrne, W.J. (1993), HTK: Hidden Markov Model Toolkit V1.5, *Cambridge University Engineering Department Speech Group and Entropic Research Laboratories Inc.*
- [Rabiner, 1989] Rabiner L.R., “ A tutorial on Hidden Markov Models and selected applications in Speech Recognition”, *Proceedings of IEEE, VOL.77, No. 2, February, 1989.*

Avaliação de aspectos de sincronização de Livros Falados Digitais

Carlos Duarte*, Luís Carriço*, Hugo Simões, Teresa Chambel*, Nuno Guimarães*

* Departamento de Informática,

LaSIGE,

Faculdade de Ciências da Universidade de Lisboa,
Campo Grande, Edifício C5, 1749-016 Lisboa, Portugal
{cad, lmc, hsimoes, tc, nmg}@di.fc.ul.pt

Resumo

Os livros falados, para além de poderem atingir audiências como os invisuais, possibilitam a "leitura" em situações onde a visão está momentaneamente indisponível. Neste artigo, apresenta-se uma plataforma de desenvolvimento de livros falados. Os livros produzidos permitem interacção multimodal, de forma a tentar alargar ao máximo o espectro de utilizadores. Para além disso, apresentam-se os primeiros passos dados no sentido de enriquecer a apresentação do livro, imergindo o leitor numa experiência multimédia. São também apresentados os resultados de testes de usabilidade, efectuados para validar as opções de implementação tomadas.

1. Introdução

A utilização conjunta da palavra escrita e falada abre novas perspectivas de exploração dos livros. A construção de um livro em formato digital, que permita acompanhar a leitura do texto, com a reprodução duma gravação digital da narração do conteúdo, traz vários benefícios aos leitores. Em relação às cassetes áudio com a gravação analógica dos livros, o formato digital não traz apenas uma melhoria da qualidade de reprodução. Outras possibilidades de interacção com os livros, que dantes seriam extremamente difíceis ou até impossíveis, tornam-se agora acessíveis. Por exemplo, procurar uma palavra ou frase no livro, é agora equivalente a fazer uma procura num documento de texto digital, enquanto antes existia a necessidade de fazer a procura directamente na cassette áudio (eventualmente tendo de escutar a totalidade da obra). Também a criação de apontamentos pessoais é agora possível, através da gravação da voz do leitor, ficando o apontamento sincronizado com o ponto da leitura em que foi feito, permitindo assim a sua reprodução na altura correcta.

Um dos sectores da população que poderá tirar mais vantagens deste formato é o dos invisuais e pessoas com dificuldades de visão. No entanto, outros segmentos da população também podem beneficiar duma plataforma com estas características. A capacidade de interacção multimodal do livro, alarga as possibilidades de leitura, a situações em que o leitor se encontra a realizar tarefas que ocupam a visão, como por exemplo, durante a condução de viaturas, ou em actividades de vigilância.

Esta plataforma digital de reprodução de livros permite também pensar em alargar a experiência de leitura tradicional, e transformá-la numa imersão num ambiente multimédia. Surge desta forma, a possibilidade de composição da apresentação do conteúdo, aproveitando recursos que se encontrem disponíveis, e que possam enriquecer o livro, criando uma nova forma de "contar histórias". Esta evolução passa pela introdução de novos elementos durante a "leitura", duma forma coerente com o conteúdo do livro. Exemplos possíveis de apontar, passam pela introdução de uma

música de fundo, de sons ambientes relacionados com o local da acção, de imagens e vídeos que possam complementar a informação disponibilizada pelo texto original, etc.

Para poder efectuar este enriquecimento é necessário (1) um repositório de “media” a ser integrada nos livros, e (2) uma forma de classificar o conteúdo dos livros para poder integrar a “media” coerentemente na história.

Neste artigo apresenta-se uma plataforma de construção de livros digitais [Duarte, et al. 2003; Carriço, et al. 2003], dando particular relevância aos novos desenvolvimentos efectuados ao nível da apresentação, e aos testes de usabilidade entretanto conduzidos para validar algumas das opções tomadas.

Os livros falados digitais são construídos a partir de uma cópia digital do texto, e de uma gravação da narração do mesmo, ambas fornecidas pela Biblioteca Nacional, um dos parceiros do projecto. Através de um processo automático de alinhamento do texto e do áudio é determinado, para cada palavra do texto, o seu instante na gravação. É assim possível a construção, de uma forma automática, de uma apresentação sincronizada. O texto é processado de forma a ficar em formato XML. Informação adicional, necessária para indexação e sincronização, é acrescentada. Todo o processo decorre de uma forma automática, até se obter uma versão do livro com o texto e o áudio sincronizados. O leitor poderá ler o livro num monitor enquanto ouve uma narração sincronizada, ou poderá optar por usar apenas uma das modalidades. Para interagir com o livro poderá usar comandos de voz, teclado e rato, duma forma coordenada ou independente. Actualmente encontra-se em construção o repositório de “media”, e em desenvolvimento uma forma de auxiliar a classificação do conteúdo do livro, com vista a atingir um processo de realizar essa classificação também automaticamente.

Na próxima secção apresenta-se um resumo do trabalho realizado nesta área, focando livros falados digitais, interfaces não visuais e métodos de análise de conteúdos. De seguida descreve-se a plataforma de construção automática dos livros falados digitais, com particular atenção para as tecnologias adoptadas. Na secção seguinte descreve-se a criação das interfaces para os livros falados digitais. Os resultados dos testes de usabilidade já conduzidos são apresentados de seguida. Finalmente apresentam-se as conclusões e o trabalho futuro a desenvolver.

2. Trabalho Relacionado

2.1. Livros Falados Digitais

Os Livros Falados Digitais (LFDs) foram pensados como um meio de facilitar o acesso da comunidade de invisuais e pessoas com deficiências visuais aos livros. Algumas organizações, em cooperação com membros dessas comunidades, desenvolveram normas respeitantes aos LFDs. Na Europa, o Daisy Consortium, com a colaboração da European Blind Union [European Blind Union 1996], desenvolveu uma norma. Nos Estados Unidos, um trabalho semelhante foi realizado pelo National Information Standards Organization (NISO), em colaboração com a The National Library Service for the Blind and Physically Handicapped. Como resultado do trabalho destas e outras organizações, várias normas foram desenvolvidas e evoluíram independentemente. No entanto, o Daisy Consortium e a NISO decidiram cooperar, tendo daí resultado o desenvolvimento da mais importante especificação de LFDs, a norma ANSI/NISO z39.86 [ANSI/NISO 2002].

O desenvolvimento destas normas possibilita a categorização dos LFDs de acordo com as funcionalidades que disponibilizam, o que também reflecte a complexidade inerente ao LFD. Assim, um LFD pode pertencer a uma das seguintes categorias [Daisy Consortium 2002]: áudio completo e apenas o título; áudio completo e controlo de navegação; áudio completo, controlo de navegação e texto parcial; áudio e texto completos; texto completo e áudio parcial; texto completo, sem áudio.

A plataforma de criação de livros falados apresentada neste artigo, permite o desenvolvimento da classe mais complexa de LFDs, com áudio e texto completos, não deixando no entanto de ser possível, gerar todos os LFDs com as características acima descritas.

De acordo com a lista de características e funcionalidades publicada pela NISO [NISO 1999], um LFD deve garantir capacidades de navegação básicas (avançar um carácter, palavra, linha, frase, parágrafo ou página de cada vez, e navegar para segmentos específicos do LFD), avanço e recuo rápidos, narração a velocidades variáveis, navegação através do índice ou de um Ficheiro de Controlo de Navegação (que deve permitir ao utilizador obter facilmente uma visão geral do conteúdo do livro), leitura de anotações, acesso a referências cruzadas, marcações, procura e outras capacidades.

No entanto, não são apresentadas na norma, quaisquer soluções específicas de implementação. Essas soluções deverão levar em consideração, não só os aspectos relacionados com as especificações propostas, mas também a natureza não visual do ambiente de execução. Conseguir transmitir ao utilizador as diferenças entre anotações, referências cruzadas, navegação estrutural, e sincronização, sem ambiguidade, e mantendo uma interface coerente, levanta problemas de usabilidade [Morley 1998].

2.2. Interfaces de Voz

O trabalho desenvolvido na área das interfaces não visuais pode indicar algumas formas de abordar as questões enfrentadas.

“Browsers” de voz são dispositivos que exibem pelo menos uma das seguintes características: (1) conseguem apresentar páginas web num formato áudio; (2) conseguem reconhecer fala e utilizá-la para controlar a navegação. A interface dos “browsers” de voz partilha com a interface dos LFDs alguns problemas comuns:

- O formato áudio é um meio temporal. Uma página apresentada visualmente pode exibir imagens, tabelas e texto simultaneamente, num formato espacial, que é rapidamente processado pelo sistema de percepção humano. Em contraste, texto falado, só pode apresentar uma palavra de cada vez.
- A emissão de comandos por voz, e o processamento de áudio, são actividades que utilizam as memórias de trabalho e de curto prazo, entrando em conflito com tarefas de planeamento e de resolução de problemas. A informação visual é processada por sistemas cognitivos distintos [Oviatt, et al. 2000].
- Os inevitáveis erros de reconhecimento.

A investigação na área dos sistemas multimodais, já tornou claro que as entradas de voz apresentam vantagens em determinadas situações [Oviatt, et al. 2000]. Alguns estudos [Van Buskirk e LaLomia 1995; Christian, et al. 2000] indicam que “as melhores tarefas para utilizar entradas de voz são aquelas em que o utilizador deve emitir comandos breves usando um vocabulário pequeno”.

As características da interação com um LFD são então vantajosas para a adopção de uma interface por voz, dado que um número relativamente pequeno de comandos pode ser utilizado para implementar as funcionalidades desejadas. No entanto, algumas limitações podem surgir, se, por exemplo, para seguir uma ligação do índice, o leitor tiver de dizer o título do capítulo.

A investigação sobre a eficiência da voz como um modo de entrada ainda não é conclusiva [Haller, et al. 1984; Martin 1989; Visick, et al. 1984], apesar de indicar um aumento do tempo necessário para completar tarefas [Van Buskirk e LaLomia 1995; Christian, et al. 2000]. Algumas das recomendações feitas para a construção de “browsers” de voz, podem ser adoptadas para o projecto de LFDs:

- As ligações devem ser texto facilmente pronunciável.
- As ligações devem ser curtas (poucas palavras).
- Deve evitar-se ligações com sons semelhantes.
- Deve desenvolver-se alternativas às ligações numeradas, dado que estas causam sobrecarga cognitiva.

Formas de transmitir ao utilizador a estrutura dos documentos e auxiliar a navegação, em ambientes não visuais, também já foram alvo de investigação. Em [Goose e Moller 1999] é proposto o uso de áudio 3D. Para transmitir o conteúdo dos documentos, como a presença de ligações e cabeçalhos, foram já estudadas várias técnicas, donde se pode destacar o emprego de ícones auditivos [Gaver 1993; Blattner, et al. 1990] e de combinações de colunas de som e efeitos sonoros [James 1997]. Os resultados obtidos com estas metodologias não são, no entanto, conclusivos, sendo por isso justificada a construção de uma bancada de produção que permita testar diferentes configurações, de forma a explorar a utilização destas e de outras técnicas.

2.3. Análise de Conteúdo

Várias abordagens podem ser empregues no processo de classificação do conteúdo de textos, variando em complexidade, e variando nas aproximações, mesmo dentro da área do processamento de linguagem natural, desde aproximações racionalistas [Chomsky 1986] até aproximações estatísticas [Manning e Schütze 2001]. Um sistema de processamento de linguagem natural deve ser capaz de clarificar o sentido das palavras, a sua categoria, a estrutura sintáctica e o contexto semântico. Recentemente tem sido dado maior ênfase à descoberta de soluções práticas, que possam ser aplicadas a texto não previamente processado, ou seja, texto como existe no “mundo real”. Estes objectivos tendem a favorecer aproximações estatísticas ao processamento de linguagem natural, porque são melhores para aprendizagem automática e para clarificação de significados [Manning e Schütze 2001].

No tocante à língua Portuguesa deve referir-se que há muito pouco trabalho desenvolvido. Por estes motivos, as aproximações à análise de conteúdos apresentadas neste artigo são baseadas na vertente estatística, e ainda de baixa complexidade.

3. Produção de Livros Falados Digitais

Os LFDs gerados pela plataforma automática são construídos a partir de cópias digitais do texto e da narração da obra, ambas fornecidas pela Biblioteca Nacional. O primeiro passo do processo de criação do LFD é determinar os alinhamentos entre as palavras no texto e na narração. A metodologia de determinação automática dos alinhamentos encontra-se descrita em [Serralheiro, et al. 2002]. O resultado deste

processo é um ficheiro com o tempo em que cada palavra é proferida na narração. Para além das palavras, são também identificados no ficheiro os silêncios, isto é, as alturas em que há pausas na leitura. As palavras que constam do ficheiro de alinhamentos nem sempre são idênticas às encontradas no ficheiro com o texto. Por exemplo, quando no ficheiro de texto se encontra “101”, no ficheiro de alinhamentos surge a versão literal do número, ou seja, as três palavras “cento e um”, cada uma com o correspondente tempo de início. Outra situação que pode causar discrepância entre os dois ficheiros prende-se com a existência de erros de reconhecimento quando o ficheiro de texto é gerado, por exemplo, a partir duma digitalização do livro original.

Este processo de alinhamento automático das palavras é da responsabilidade de outro dos parceiros do projecto. Do ponto de vista do trabalho apresentado no artigo identificam-se três entradas para o gerador de LFDs: (1) uma cópia digital do texto, (2) uma narração do texto, também em formato digital, e (3) o ficheiro de alinhamento entre as palavras da cópia e da narração.

A partir deste momento, o processo segue de uma forma automática até se concluir a produção de um LFD pronto para “leitura”. O processo encontra-se dividido em três fases: fase 0 – pré-processamento; fase 1 – produção do LFD; fase 2 – construção da apresentação. Na fase 0 processam-se os ficheiros de entrada, de forma a obter o mesmo conteúdo, mas num formato normalizado. Na fase 1 procede-se à reestruturação de todos os elementos que digam respeito ao conteúdo do LFD, e à sua informação estrutural e de navegação. Na fase 2, utilizando uma especificação de apresentação, constrói-se a versão interactiva do LFD. Na figura 1 pode ver-se um resumo de todo o processo.

A divisão do processo de produção em três fases foi motivada pela necessidade de criar LFDs, numa forma automática, para audiências com características e capacidades muito diferentes. Assim, a especificação de apresentação permite criar várias interfaces para o mesmo livro, sendo inclusivamente possível alterar os modos de entrada e de saída utilizados. O próprio processo de enriquecimento do livro sai facilitado. Deste modo, a separação entre conteúdo e apresentação é total, factor importante para a produção automática de LFDs que possam ser adaptados a cada um dos seus leitores.

Para a implementação do processo de criação automática de LFDs foi utilizada a plataforma Apache Cocoon [Langham e Ziegler 2002]. Esta plataforma utiliza XML e XSLT como tecnologias base, numa arquitectura baseada em “pipelines”, facilitando dessa forma a reutilização de componentes.

De seguida descrevem-se numa forma mais detalhada as fases 0 e 1 de produção de LFDs, sendo a fase de produção da interface do livro focada na próxima secção.

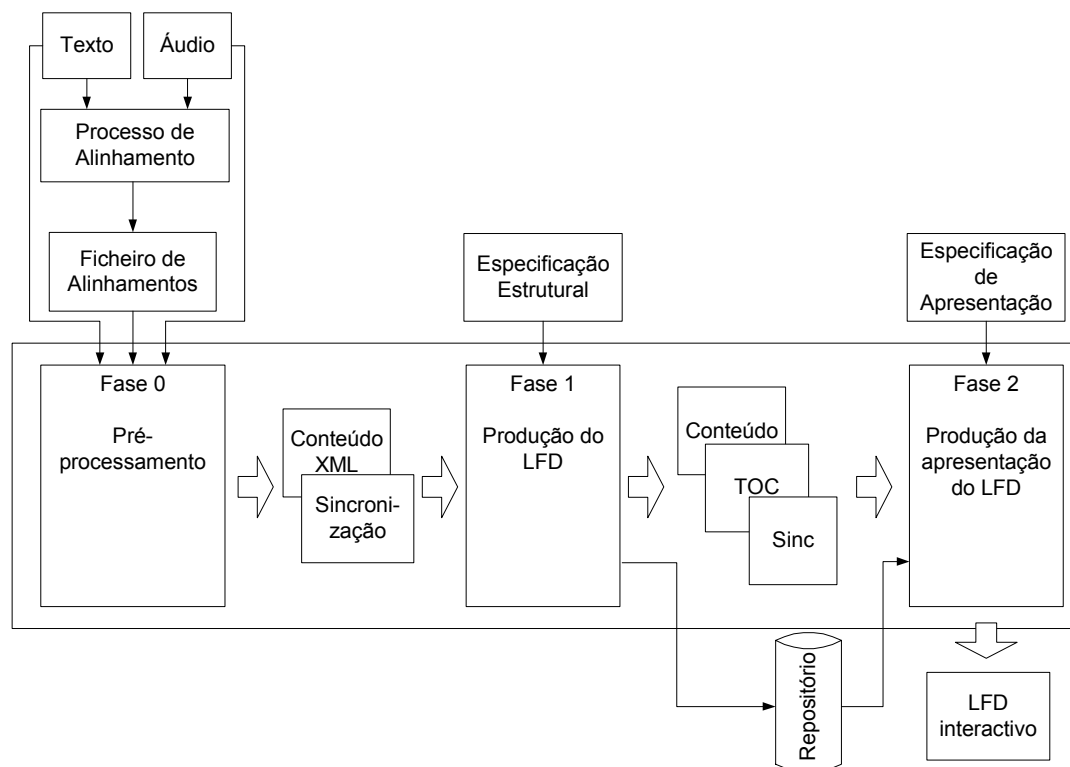


Figura 1- Processo de produção de LFDs

3.1. Pré-processamento

A fase de pré-processamento tem com objectivo restabelecer a informação sintáctica do livro, traduzindo e normalizando os diferentes formatos em que se encontram as entradas. Esta necessidade justifica-se pelo desconhecimento da estrutura usada para representar o conteúdo do livro na cópia digital fornecida. Para além disso é necessário efectuar uma reposição do alinhamento, devido às diferenças de conteúdo quando comparados a cópia do texto e o ficheiro de alinhamentos. O resultado do processo de alinhamento é um conjunto de tempos para cada palavra dita na narração, o que se pode denominar “alinhamento de som”. Aquilo que é necessário para efectuar uma apresentação sincronizada entre o texto e a narração, é uma lista de tempos para cada palavra constante do texto, que pode ser denominado por “alinhamento de texto”. É por isso necessário proceder a uma reposição do alinhamento resultante do processo de extracção de tempos. Assim, para além da situação anteriormente referida, causada pela presença de números em formato numérico, que surgem no ficheiro de alinhamentos sob a forma literal, outras situações produzem inconsistências: o uso de abreviaturas no texto, que depois surgem expandidas no ficheiro de alinhamento; os sinais de pontuação, que não estão presentes no ficheiro de alinhamento; e erros de reconhecimento quando o texto é obtido a partir de um processo de digitalização. Em [Duarte, et al. 2003] encontra-se uma descrição dos métodos utilizados para resolver estes problemas.

A fase 0 produz então um ficheiro normalizado com o conteúdo do livro, um ficheiro com a informação de sincronização para cada palavra existente no texto do livro, e, opcionalmente, um ficheiro com metadados no formato Dublin Core, caso estes tenham sido disponibilizados pelo fornecedor do conteúdo do livro.

3.2. Produção do LFD

A fase de produção do LFD parte destes dados, e recorrendo a uma especificação estrutural, gera os aspectos referentes ao conteúdo do LFD, independentemente da apresentação, que será alvo de atenção na próxima fase. A especificação estrutural cobre os aspectos relativos à estrutura e navegação do LFD, à identificação de unidades de sincronização e ao processo de análise de conteúdo. Informação relevante é fornecida para a criação de índices e tabelas de conteúdos que podem ser utilizados para navegação no livro. Outro dos aspectos relevantes processados nesta fase é referente às capacidades de sincronização. Apesar de a norma ANSI/NISO para LFDs prever apenas a utilização de uma unidade de sincronização (a palavra), esta plataforma de produção de LFDs tem a capacidade de gerar livros com mais do que uma unidade de sincronização. A vantagem de ter várias unidades de sincronização é aumentar as possibilidades de interacção. As unidades de sincronização podem ser usadas com diferentes fins. Por exemplo, durante a apresentação do livro, a frase que está a ser proferida muda de cor. Para conseguir esse efeito é necessária uma sincronização à frase. No mesmo livro, quando o utilizador executa uma procura por uma palavra, o texto passa a ser lido a partir da primeira palavra encontrada, sendo para isso necessário uma sincronização à palavra. Desta forma se pode observar que diferentes funcionalidades ou formas de interacção podem necessitar de diferentes unidades de sincronização.

Assim, a fase 1 gera um ficheiro XML com o conteúdo do livro, organizado hierarquicamente segundo os seguintes elementos: livro, volume, capítulo, secção, subsecção, parágrafo, frase, oração e palavra, eventualmente limitado pela granularidade de sincronização; um ficheiro XML com a árvore de navegação de conteúdo (semelhante a uma tabela de conteúdos); e um ficheiro XML com informação respeitante às várias unidades de sincronização solicitadas.

Também relevante na fase 1 é o processamento realizado para analisar o conteúdo do texto, e a possível extracção de conteúdo para reutilização em apresentações de outros livros. Actualmente encontram-se em desenvolvimento para a plataforma de construção de LFDs as primeiras aproximações à determinação de palavras chave para classificação de conteúdo. Sendo este o trabalho inicial nesta área, as aproximações são ainda de baixa complexidade.

Através de uma análise estatística do conteúdo do livro, realiza-se uma filtragem de forma a obter uma lista de palavras relevantes. A partir da lista de palavras são determinadas quais as palavras chave do texto. O conjunto de palavras chave pode ser determinado por mais do que um método. Uma hipótese é escolher as n primeiras classificadas da lista. Outra hipótese é definir um limiar, e todas as palavras com um número de ocorrências acima do limiar farão parte do conjunto de palavras chave. Esse limiar deverá ser definido em função do número total de palavras do texto, pois a importância de uma palavra que ocorre dez vezes num texto de mil palavras, não é a mesma se o texto tiver cem mil palavras.

Futuramente, pretende-se evoluir esta aproximação determinando a raiz das palavras. Desta forma, palavras que tenham raiz comum serão contadas como uma só, sendo escolhida como palavra chave a raiz desse conjunto de palavras.

4. A Interface do LFD

Na fase de construção da apresentação, são abordados os temas respeitantes à interacção com o LFD. Partindo do conteúdo gerado na fase anterior, podem ser

construídas formas diferentes de apresentar o mesmo livro, e também escolher diferentes modos de interacção.

Os modos de interacção podem variar entre teclado, rato e comandos de voz para as entradas, ou uma combinação deles. Para as saídas, o áudio ou a apresentação visual podem ser empregues, individualmente ou de forma coordenada. Para além dos modos de interacção empregues na interface, a especificação da apresentação pode ainda determinar outros pormenores, tais como a gramática empregue para reconhecer os comandos de voz dos utilizadores, o nível de detalhe de apresentação da tabela de conteúdos (apresentar só os capítulos ou incluir também secções e subsecções), a forma como é apresentada a sincronização entre a narração e o texto escrito, a visualização das anotações feitas pelo utilizador, e a forma de informar o utilizador da presença de uma anotação quando esta é alcançada durante a narração do texto. Também o nível de enriquecimento do livro com elementos do repositório de “media” é detalhado na especificação da apresentação. Este enriquecimento é guiado pelas palavras chaves determinadas na fase anterior. Quando um tema dominante do livro é determinado, existe a opção de adicionar à apresentação os conteúdos do repositório de elementos que estejam classificados com esse tema.

Na figura 2 mostra-se uma apresentação construída para o livro “O Senhor Ventura” de Miguel Torga. Esta apresentação, a ser visualizada no browser Internet Explorer 6, foi construída recorrendo à linguagem HTML+TIME [Microsoft Corporation 2002], a implementação da Microsoft da linguagem SMIL [Bulterman 2001]. Esta linguagem fornece as capacidades de sincronização necessárias para uma apresentação coordenada do texto e da narração. Outras apresentações foram criadas recorrendo à linguagem HTIMEL [Chambel, et al. 2001] para sincronização. Para reconhecimento de voz foi utilizada a plataforma Microsoft Speech.

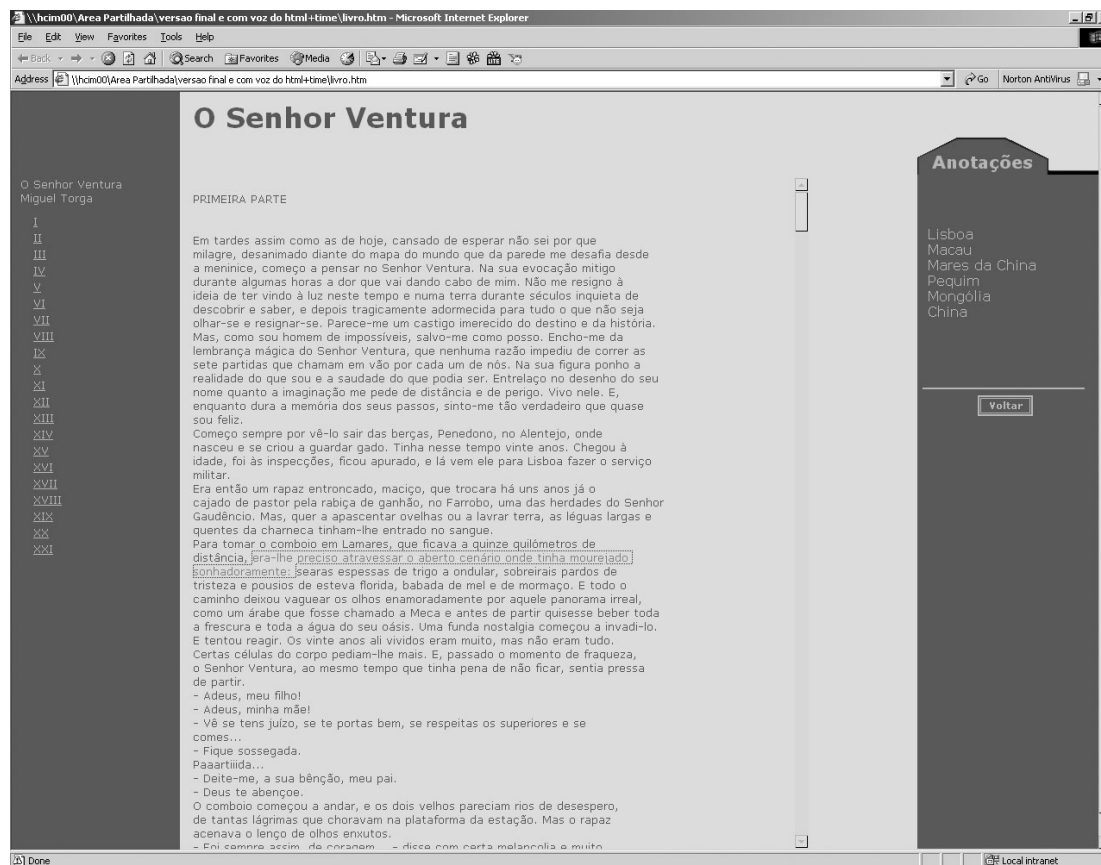


Figura 2 – Uma apresentação do livro “O Senhor Ventura”.

Nesta apresentação o texto é exibido visualmente enquanto é narrado. A sincronização é apresentada mudando a cor da frase que está a ser proferida. O utilizador pode interagir com o livro usando teclado, rato e comandos de voz, aproveitando assim a natureza multimodal da interacção. Na sua maioria, os comandos podem ser emitidos em qualquer dos modos. A tabela de conteúdos apresenta os diferentes capítulos do livro e pode ser usada para navegação, quer seleccionando o capítulo com o rato, quer emitindo vocalmente o comando “ir para” seguido do número do capítulo. Do lado direito do ecrã, o utilizador pode ver as anotações que já foram criadas. Nesta apresentação são exibidas as primeiras palavras de cada anotação, mas a especificação de apresentação pode produzir resultados diferentes. Pode ser exibida a data e hora de criação da apresentação, ou também pode ser exibida a anotação na sua totalidade. Ao seleccionar a anotação com o rato, o seu conteúdo é mostrado, e a narração do livro, bem como o texto exibido, avança para o ponto de leitura em que a anotação foi criada. O mesmo resultado pode ser alcançado dizendo “ler nota” seguido do número da anotação. Um sinal sonoro é reproduzido sempre que a narração chega a um ponto do livro onde foi criada uma anotação. A introdução de anotações pode ser efectuada seleccionando o botão “adicionar” ou dizendo o comando “adicionar”. Actualmente a aplicação suporta anotações de texto e imagens, e futuramente virá a ser adicionado o suporte para anotações áudio e vídeo. O utilizador pode também parar e reiniciar a apresentação a qualquer momento, com os comandos “pause” e “play”.

Para este livro, uma das palavras chaves determinadas pela análise estatística da fase anterior é “mar”. Assim, a apresentação pode ser enriquecida através da adição de conteúdos presentes no repositório de elementos que estejam catalogados como relativos a “mar”, nomeadamente a inclusão de segmentos áudio com o som do mar.

5. Testes de Usabilidade

Para avaliar a interface dos LFDs construídos conduziram-se, até ao momento, duas sessões de avaliação. O principal objectivo foi avaliar aspectos de usabilidade da interface, e o grau de satisfação dos utilizadores. Uma característica particular avaliada prende-se com a forma como é transmitido ao utilizador a sincronização entre o texto apresentado no ecrã e o texto narrado. Para isso foram utilizados duas interfaces para o mesmo livro. A primeira interface é a apresentada na secção 4 (figura 2). Nesta interface a sincronização é transmitida ao utilizador através da mudança de cor das palavras que estão a ser narradas. Mais concretamente é alterada a cor de todas as palavras entre dois silêncios. A sincronização mudando a cor palavra a palavra revelou-se impraticável, excepto em exemplos de dimensão reduzida, pelos recursos computacionais necessários para visualizar os livros construídos dessa forma. A segunda interface (figura 3) apresenta a sincronização através de um indicador da linha que se encontra a ser narrada. Nesta interface não existe alteração da cor das palavras. Todos os outros componentes são idênticos nas duas interfaces.

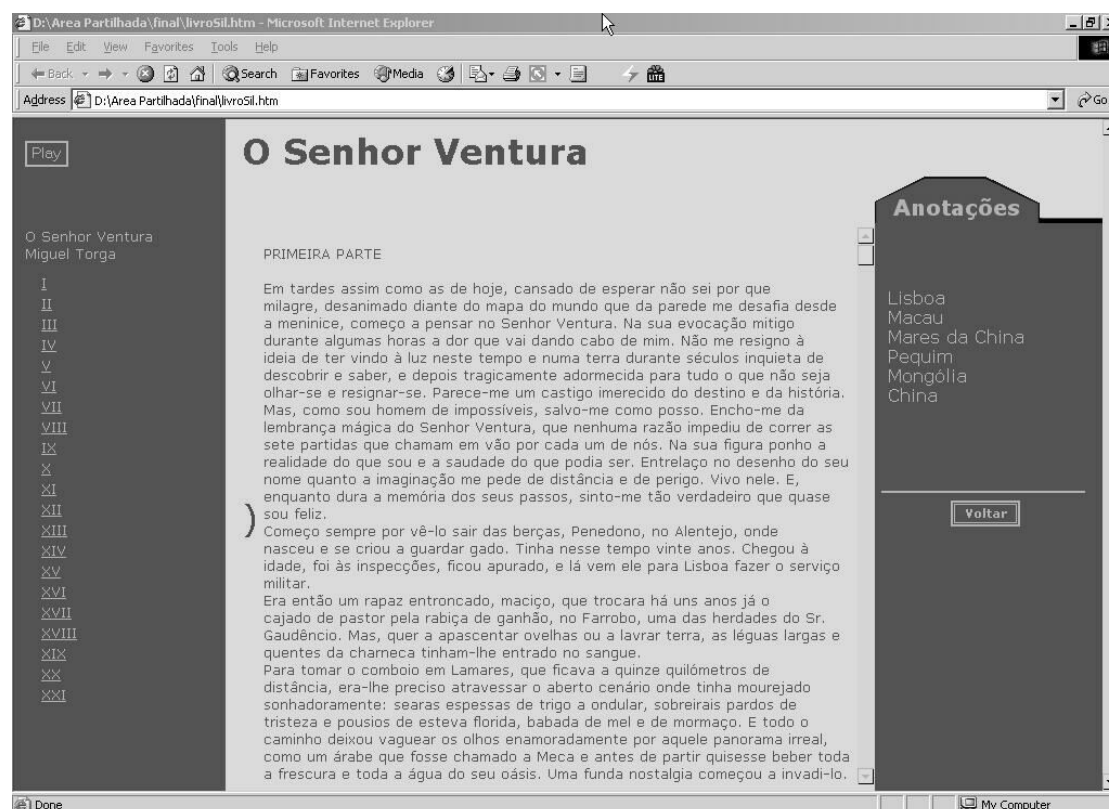


Figura 3 – Segunda interface para o livro “O Senhor Ventura”

Os testes foram realizados com dezasseis pessoas, tendo cada uma das interfaces sido testada por oito utilizadores. Nenhuma das pessoas sofria de problemas visuais ou auditivos. A interface foi explicada a cada um dos utilizadores, sendo-lhes disponibilizado de seguida o tempo que necessitassem para se familiarizar com ela. Quando assinalavam estar preparados, era-lhes apresentada uma lista de 12 tarefas. Estas tarefas exigiam navegação no livro para responder a questões sobre o conteúdo e para criação de anotações. Os utilizadores eram livres de escolher o modo como emitiam os comandos. No final das tarefas os utilizadores responderam a um questionário.

Em termos de facilidade de utilização foi considerado pelos utilizadores (tabela 1) que seguir ligações, quer da tabela de conteúdos, quer das anotações, era mais fácil com o rato do que com comandos de voz. No entanto, essa diferença esbate-se quando se considera as acções de criação de anotações e de controlo da reprodução.

Tabela 1 – Facilidade de Utilização dos modos de entrada

	Ligações da Tabela de Conteúdos		Ligações das Anotações		Criação de Anotações		Controlo da reprodução	
	Rato	Voz	Rato	Voz	Rato	Voz	Rato	Voz
Não Usado	-	13%	6%	25%	13%	31%	6%	6%
Muito Difícil	-	6%	-	6%	-	6%	6%	-
Difícil	-	38%	6%	31%	6%	-	-	13%
Fácil	19%	38%	31%	31%	50%	31%	38%	31%
Muito Fácil	81%	6%	56%	6%	31%	31%	50%	50%

Quanto à utilidade das funcionalidades disponibilizadas, a maioria dos utilizadores considerou a existência de ligações da tabela de conteúdos para o texto, e o controlo da reprodução indispensáveis. As anotações foram consideradas muito úteis. A possibilidade de interagir multimodalmente foi considerada pouco útil por 19% dos utilizadores, muito útil por 56% e indispensável por 25%.

A nível de satisfação, enquanto que todos os utilizadores se mostram satisfeitos com a interacção por rato e teclado, houve dois utilizadores que se revelaram pouco satisfeitos com os comandos de voz.

Os comentários dos utilizadores e a observação dos testes permitiram identificar alguns problemas de usabilidade das interfaces testadas. Uma das questões mais vezes levantada refere-se à ausência de identificação do capítulo que está a ser narrado. Dada a falta de confiança no reconhecedor de voz, esta questão causa insegurança aos utilizadores quando seguem ligações da tabela de conteúdos através de comandos de voz. Problema semelhante ocorre quando seguem ligações das anotações. Uma outra questão, também relevante, é a não existência de números nas anotações, o que forçava os utilizadores a terem de contá-las para usar comandos de voz, quando desejavam seguir a ligação respectiva. Outros problemas mencionados prendem-se com a inexistência de um mecanismo de procura (no entanto o disponibilizado pelo “browser” pode ser utilizado), e com a sensibilidade e ineficácia do reconhecedor de voz.

No que diz respeito à avaliação das apresentações da sincronização, os resultados não podiam ser mais esclarecedores. Enquanto que todos os utilizadores da primeira interface detectaram falhas de sincronização, e referiram o desconforto causado por essas falhas, nenhum utilizador da segunda interface referiu qualquer falha de sincronização. No entanto, um destes utilizadores comentou que seria preferível uma forma de sincronização que acompanhasse a palavra narrada.

6. Conclusões e Trabalho Futuro

Neste artigo apresentou-se uma plataforma de desenvolvimento de livros falados digitais. Esta plataforma, a partir de cópias do texto e de uma narração em formato digital, permite construir apresentações do livro que combinam texto e áudio numa forma sincronizada. O “leitor” pode interagir multimodalmente com o livro, usando o teclado, rato e comandos de voz. A construção do livro é automática, recorrendo à plataforma Apache Cocoon para controlar o processo. O conteúdo do livro, bem como informação de navegação e sincronização são representados usando XML. Para obter apresentações sincronizadas já se utilizaram duas linguagens, HTML+TIME e HTIMEL.

Durante o processo de produção, o conteúdo do livro é separado da sua apresentação. Desta forma é possível reutilizar recursos para construir diferentes apresentações para o mesmo livro. Esta separação também permite efectuar o enriquecimento do livro de uma forma mais expedita.

Um livro enriquecido contém conteúdos acrescentados pela plataforma, no sentido de melhorar o entretenimento ou a produtividade do leitor. Para enriquecer o livro é necessário um repositório de elementos que possam ser utilizados com esse fim, e um processo de classificação de conteúdos dos livros, para determinar quais os elementos escolhidos. Neste artigo descreveu-se o trabalho realizado até agora, no sentido de conseguir automatizar o processo de classificação de conteúdos. Uma aproximação inicial, pouco complexa, foi detalhada. Outras aproximações, mais complexas, terão

de ser desenvolvidas para possibilitar uma análise semântica dos livros. Um próximo desenvolvimento passa pela reunião de palavras com a mesma raiz.

As capacidades disponibilizadas por esta plataforma, sugerem o desenvolvimento futuro de livros falados capazes de se adaptarem em tempo real ao seu leitor. Para isso, será necessário iniciar o desenvolvimento de modelos de utilizador, identificando quais as características destes que podem ser usadas para adaptar o livro. Funcionalidades que permitam ao utilizador indicar o seu grau de satisfação acerca de determinada característica da interface, ou que se apercebam desse grau através da observação das acções do utilizador serão preponderantes para permitir implementar capacidades de adaptação do livro.

Os resultados dos testes de usabilidade conduzidos até ao momento indicam a importância de uma sincronização, entre o texto escrito e o narrado, sem falhas. Quando são perceptíveis pelo utilizador, as falhas de sincronização provocam desconforto e dificuldades de manutenção de concentração nos momentos em que ocorrem. Uma das interfaces testadas, utilizando unidades de sincronização com menor detalhe (linha em vez de palavras) conseguiu eliminar as falhas de sincronização e os problemas a elas associados.

Outro dos problemas de usabilidade identificados diz respeito à falta de informação acerca do capítulo do livro que está a ser lido, o que, conjuntamente com a falta de confiança no reconhecedor de voz, fez com que vários utilizadores preferissem utilizar o rato para seguir as ligações da tabela de conteúdos.

De notar que, apesar do desempenho por vezes pouco eficaz do reconhecedor de voz, apenas 2 utilizadores se mostraram pouco satisfeitos com os comandos de voz, e que 81% dos utilizadores consideram que a possibilidade de interagir multimodalmente é muito útil ou indispensável.

Futuros testes irão permitir avaliar soluções para os problemas de usabilidade identificados nestas interfaces. Testes efectuados usando interfaces que disponibilizem apenas comandos de voz como modo de entrada, irão permitir descobrir novos problemas de usabilidade, bem como identificar os comandos de voz que os utilizadores preferem, quando não se encontram restringidos a uma gramática definida pela aplicação.

7. Referências

- ANSI/NISO, “Specifications for the Digital Talking Book (ANSI/NISO Z39.86-2002)”, 2002, <http://www.niso.org/standards/resources/Z39-86-2002.html>
- Blattner, M. et al, “Earcons and icons: Their Structure and Common Design Principles”, in Ephraim P. Glinert (Ed.), Visual Programming Environments: Applications and Issues, IEEE Computer Society Press, Los Alamitos, CA, 1990, 582-606.
- Bulterman, “Smil 2.0: Overview, concepts, and structure”, IEEE Multimedia, 8, 4, (2001), 82-88.
- Carriço, L. et al, “Spoken Books: Multimodal Interaction and Information Repurposing”, Proceedings of the 10th International Conference on Human-Computer Interaction, Creta, Grécia, 2003.
- Chambel, T. et al, “Hypervideo on the Web: Models and Techniques for Video Integration”, International Journal of Computers & Applications, 23, 2, (2001), 90-98.

- Christian, K. et al, "A Comparison of Voice Controlled and Mouse Controlled Web Browsing", Proceedings of ASSETS 2000, ACM Conference on Assistive Technologies, Arlington, VA, 2000, 72-79.
- Chomsky, N, Knowledge of Language: Its Nature, Origin and Use, Prager, New York, 1986.
- Daisy Consortium, "Daisy Structure Guidelines", 2002,
<http://www.daisy.org/publications/guidelines/sg-daisy3/structguide.htm>
- Duarte et al, "Producing DTB from audio tapes", Proceedings of the Fifth International Conference on Enterprise Information Systems ICEIS2003 (Volume 3), Angers, França, 2003, 582-585.
- European Blind Union, Reaching Forward to the 21st century: User Requirements for the Next Generation of Talking Books, RNIB, London, 1996.
- Gaver, W., "Synthesizing Auditory Icons", Proceedings of the ACM INTERCHI 1993, Amsterdão, Holanda, 1993, 228-235.
- Goose, S. e C. Moller, "A 3D Audio Only Interactive Web Browser: Using Spatialization to Convey Hypermedia Document Structure", Proceedings of the seventh ACM international conference on Multimedia (Part 1), Orlando, Florida, 1999, 363-371.
- Haller, R. et al, "Comparison of Input Devices for Correction of Errors in Office Systems", Proceedings of INTERACT 84, Londres, Reino Unido, 1984, 177-182.
- James, F., "Presenting HTML Structure in Audio: User Satisfaction with Audio Hypertext", Proceeding of the International Conference on Auditory Display ICAD97, Palo Alto, California, 1997, 97-103.
- Langham, M. e C. Ziegler, Cocoon: Building XML Applications, New Riders, 2002.
- Manning, C. e H. Schütze, Foundations of Statistical Natural Language Processing, MIT Press, Cambridge, Massachusetts, 2001.
- Martin, G., "The Utility of Speech Input in User-Computer Interfaces", International Journal of Man-Machine Studies, 30, 4, (1989), 355-375.
- Microsoft Corporation, HTML+TIME 2.0 Reference, 2002,
http://msdn.microsoft.com/workshop/author/behaviors/reference/time2_entry.asp
- Morley, S., "Digital Talking Books on a PC: A Usability Evaluation of the Prototype DAISY Playback Software", Proceedings of ASSETS 1998, ACM Conference on Assistive Technologies, Marina del Rey, California, 1998, 157-164.
- NISO, Document Navigation Features List, 1999,
<http://www.loc.gov/nls/z3986/background/navigation.htm>
- Oviatt, S. et al, "Designing the user interface for multimodal speech and gesture applications: State-of-the-art systems and research directions", Human Computer Interaction, 15, 4, (2000), 263-322.
- Serralheiro, A. et al, "Spoken Book Alignment Using WFSTS", Proceedings of HLT2002, Human Language Technology Conference, San Diego, California, 2002.
- Van Buskirk, R. e M. LaLomia, "A Comparison of Speech and Mouse/Keyboard GUI Navigation", Proceedings of ACM CHI 1995 (Volume 2), 1995, 96.
- Visick, D. et al, "The Use of Simple Speech Recognizers in Industrial Applications", Proceedings of INTERACT 84, Londres, Reino Unido, 1984.

Collaborative Multimodal Authoring of Virtual Worlds

Vítor Sá^{1,2}

vitor.sa@dsi.uminho.pt

¹ University of Minho
Campus de Azurém
P-4800-058 Guimarães

Filipe Marreiros²

fmarrei@igd.fhg.de

² Computer Graphics Center
Fraunhoferstr. 5
D-64283 Darmstadt

Adérito Marcos^{1,3}

aderito.marcos@ccg.pt

³ Computer Graphics Center
R. Teixeira de Pascoais, 596
P-4800-073 Guimarães

Abstract

This work aims the creation of virtual worlds by working teams with resource to advanced interaction techniques, such as speech commands and gestures, as well as the resource to portable devices. The physical characteristics of virtual reality environments lead us to a different kind of user interfaces. However, we found some usefulness on the classic WIMP paradigm, which we also apply by using a PDA, in order to complement the interaction possibilities. In terms of application description we use the Virtual Reality Modeling Language. One of the main goals is the separation/combination of VRML files in order to allow the work to be cared out individually by the members of the team, and later combine the results to produce the global virtual world.

1. Introduction

The present article describes an experimental work in the area of the creation of virtual worlds by working teams with resource to advanced interaction techniques, such as speech commands and gestures, as well as the resource to portable devices. The use of these last ones becomes indispensable due to the physical characteristics of the environment, large projection screens or workbenches, in which it's also useful to have access to WIMP interfaces in order to complement the user interaction possibilities.

The prototype under development is very flexible mainly because: users can use several modalities, in whatever combination they desire (multi-modality); users can change modalities, even during a single interaction - the more convenient for a specific situation (flexi-modality); users may be using a workstation, standing up at large displays or in a mobile situation, and can still continue they work somehow (multi-machine).

In order to exemplify possible scenarios of application, imagine the user creates the virtual world by using speech and direct manipulation in a 3D environment:

- There are things which are difficult to achieve with speech commands, e.g. precise measurements;
- We would like to add the vocabulary grammar with new concepts, e.g. if we are building the geometry of a chair and at the end we want to refer to it as "chair";
- We would like to set the parameters of a color that is not on the grammar, or we would like to refine some other world properties.

This kind of things can be done with our solution in the following ways:

- Separately, in a workstation without the need of a VR system apparatus;
- Collaboratively, by a group of persons that will perform their individual tasks to accomplish a common goal – the creation of the global virtual world;
- Complementally, by using a handheld device without leaving the workbench place, and seeing automatically the consequent results.

To achieve all these goals, several technologies were used and we start by presenting them on the next section. On section 3, we present the application functionality and the encountered solution. We finish with some conclusions and possible future improvements.

2. Involved technologies

2.1. Virtual reality

The virtual reality (VR) environment we used for our experiments is based on a workbench with a display surface of about 1.36 m * 1.2 m, which means a volume of interaction of about 3 m * 3 m * 1.5 m above and in front of the table (width * depth * height). This means that the user can interact with the virtual world within this volume, where his positions and movements have to be precise and efficiently tracked. In order to “run” the virtual table we used the “Avalon” system [Behr 2003], developed at the Computer Graphics Center in Darmstadt.

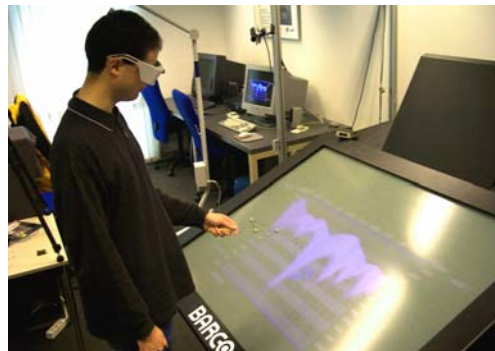


Fig. 1 – Workbench environment

“Avalon” uses the Virtual Reality Modeling Language [VRML] with some extensions as scene description language. The use of VRML has several advantages: the interface is well defined by a non-proprietary, platform and company independent standard (ISO/IEC 14772), the application developer can use a wide range of VRML modeling tools and, very important in our case, it is possible to display the same VRML file on a 2D browser on a desktop PC using 2D input devices, as well as on a VR system with 3D displays using 6D input devices. For this purpose, “Avalon” extends the concepts of the VRML “Viewpoint” and “TouchSensor” nodes: the first one still sets the global viewing direction into the scene, but the view frustum is modified according to the current head position and orientation of the user; the second node reacts on the collision of a 3D cursor with objects in the 3D scene [Sá 2001].

The user controls the 3D cursor with his hand that gets tracked by a tracking system. By this way it is possible to interact with the virtual scene in a very simple and natural way by pointing at objects.

2.2. Gesture recognition

Regarding the interaction mechanisms, for the direct manipulation (gesture modality) of the virtual world we are using the tracking system “EOS“, also developed at the Computer Graphics Center [Schwald 2002]. The system uses a stereoscopic approach allowing natural interaction (with 6DOF) within the virtual world via a pointing gesture. This is combined with a speech recognition component to enhance the independent uni-modal inputs by an integrated multi-modal approach.

We are using video-based tracking with infrared beacons and retro reflective markers, which allows good real-time results even without special light conditions. By this way, the VR system keeps track not only of the user’s head position, to render the images in the correct perspective, but also of the user’s interaction device, the way the user has to perform direct manipulations.

2.3. Mobile device

Our mobile device consisted on a iPAQ Pocket PC with wireless network access. We made our experiments using Virtual Private Network (VPN) technology, inside the *firewall* of our organization. The mobile unit worked as a VPN client connected to a VPN server; the gateway to the other computers behind it on the subnet.

In terms of application development, we adopted the Java technology, taking advantage of its portability, network support and multithreading. We have programmed in conformance with Java 2 Micro Edition (J2ME), and our iPAQ was equipped with Jeode Java Virtual Machine [Jeode].



Fig. 2 – Graphical interface of the mobile unit

The J2ME defines two major categories of components: configurations and profiles [J2ME]. The components we needed were the Connected Device Configuration (the one more suited for high-end PDAs), the Foundation Profile and the Personal Profile. The first component is a vertical set of APIs that provides the base functionality, such as memory footprint and network connectivity. The rest constitute a horizontal set of

high-level APIs providing access to the device capabilities ranging from I/O to Graphical User Interface.

2.4. Speech recognition

We have used the *Java Speech API* (JSAPI) that defines a software interface to state-of-the-art speech technology. The two core speech technologies (recognition and synthesis) are supported by JSAPI. There are several engines on the market with very reasonable accuracy for both technologies, e.g. IBM Via Voice and Microsoft Speech, which are speaker-independent and fully-continuous.

The goal of speech recognition is to transform the user voice (an audio data stream) into a text string. This text string respects specific grammar rules and includes only syntactically correct words. The closer we get to fully unrestricted natural language, the more difficulties we encounter. The use of an artificial language of special commands fulfills our user requirements. Using the Java Speech Grammar Format (JSGF), we built a command-and-control recognizer covering several types of interaction: objects generation, attributes changing, movements and inspections.

2.5. Multimodal interaction

We are following the classical three-level approach, with its lexical, syntactic and semantic layers; a rather straightforward adoption of the LANGUAGE model described by Foley and Van Dam [Foley 1982]. In our context, the lexical layer corresponds to the binding of hardware primitives to software events, in which temporal issues are of main importance; the syntactic layer is where the sequencing of events is performed, that is the combination of data to obtain a complete command, and; the semantic layer is related to the functional combination of commands in order to generate new, more complex ones. So, in a multimodal point of view, each individual modality can be in a stage considered semantic by itself, but without having any meaning in the overall context – this means a correct multi-modal syntax, without any meaning or semantic.

In terms of integration of the different modalities we are using a semantic fusion approach, considering the modalities in a so-called late fusion state. This is appropriate when the modes differ substantially in the time scale characteristics of their features [Wu 1999]. The functional combination of commands is being performed by methods such as state machines or parsers (note in our context that state machines are parsers for regular grammars).

3. Application functionality

The application goal is the creation of a virtual world by a group of persons, using different devices dependent on availability or specific needs. The result is textually described in VRML, which means the need of some separation/combination mechanism of this kind of files.

The interaction possibilities are different depending on the system we are working on, since we have commutations between 3D and 2D environments. In the desktop system the user looks at a 2D projection of the 3D world that is independent of his actual head position and orientation. This limits the possibilities of interaction with

the 3D world. For example, the VRML “TouchSensor” node detects the virtual object the user is pointing at by shooting a ray into the scene, while in 3D this is done by collision, as we mentioned previously.

Opposed to the kind of work that is done at the workstation, the mobile device is used to send information to the VR system, with immediate reaction. It is not possible to do much more with this type of device, but the possibility of having it on the palm of the hand is very useful to complement authoring tasks.

3.1. Authoring

From a implementation point of view, the range of operations that is possible to perform are the accomplishment of a set of VRML nodes, with a root element, to a specific node already existing in the current world, the counterpart remove operation, adding and removing routes, sending time stamps (for the time-dependent nodes) and sending/receiving events.

The dialog manager is based on the parse returned by the speech recognizer. We encode meta-level information about the utterances, using the tag facility of the Java Speech Grammar Format. This information together with the gesturing selections is then used to determine the action that is requested by the user and what objects will be affected.

3.2. Collaboration

In terms of the multi-machine characteristic, the work can be carried out in several devices, thanks to the distributed file system used - Samba Unix. But one of our goals is to allow the combination and separation of the work. Has referred earlier we pretend that a virtual world may be constructed by several users. One immediate question arises: how do we separate the space? One possibility is to attribute portions of the space to the several users. Other possibility would be to assign objects to the users, i.e. each user has its own set of objects. Many questions would arise using any of the approaches taken. To give an example lets consider that a user creates a light source; this light may affect other users, and problems occur when trying to separate and combine the created virtual world. In our prototype we considered 8 users, all with the same amount of space, like is presented in the figure:

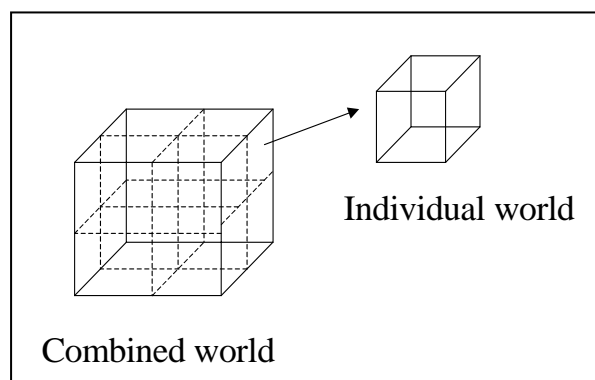


Fig. 3 – Spatial world subdivision

We still didn't consider the cases where objects share several individual worlds. As we referred earlier, further work has to be carried out.

Concerning the combination and separation of VRML files, by parsing them we can find out where an object is located in the space, and this way we can assign the objects to the users. So we can create a VRML file for each user. All the users have to specify the region of space that he/she owns, and the input and output files. So to make a combination and a separation we have the following commands with the format specified:

- combine <nsr> <fiw> <fcw> <ipcw> for combination;
- separate <nsr> <fiw> <fcw> <ipiw> for separation,

where:

- nsr is the identifier of the selected region, in our case we can select 1 of 8 possible regions;
- fiw is the name, including the path, of the file containing the individual world;
- fcw is the name, including the path, of the file containing the combined world;
- ipiw is the IP address of one of the machines that we want the individual world to be saved to;
- ipcw is the IP address of one of the machines that contains the combined world. If the file doesn't exist a new file (world) will be created.

As can be noticed we are using communications that will allow our data to be saved in a desired machine. This is done using a TCP/IP socket connection. To allow the desired actions to be performed, we have running a TCP/IP server on the several machines that take part of the process of creating the world. These servers are waiting for messages from the clients. These messages are sent when the referred commands are used. The clients (commands) use their input parameters to produce the desired message to be sent.

This easy exchange of information is a relevant benefit in terms of collaboration. The users have an easy way of combining their individual work. Furthermore, if the combined world is changed the users can also get that information, provided by the person(s) that are managing the combined virtual world.

4. Conclusion and future work

The work we presented here is currently under development, but the first tests combining the several technologies are encouraging. One problem is still the world separation and combination. In this respect we will experiment other techniques besides the one presented here.

We also plan to expand the type of possible devices, namely for mobility. This means that the user can start his work in his workstation, save it and then continue in a laptop. One problem that may arise is the connections failure. The user can always save his information to a local drive and continue his work disconnected. This is an approach similar to the used by the Coda File System [Coda]. Coda makes a persistent caching

of files, which means that the files recently used by the user exist in the local drive. This allows a disconnected operation for mobile users (clients).

References

Behr, J., Dähne, P.: "AVALON: Ein komponentenorientiertes Rahmensystem für dynamische Mixed-Reality Anwendungen", In: Thema Forschung, 1, pp. 66-73, Germany, 2003.

Coda, Online reference <http://www.coda.cs.cmu.edu/> .

Foley, J. et al.: Fundamentals of interactive computer graphics. Addison-Wesley, Reading, MA, 1982.

Jeode, Online reference http://www.esmertec.com/products/jeode_dell_runtime.shtm

J2ME, Online reference <http://wireless.java.sun.com/configurations/articles/cdc/> .

Schwald, B., Malerczyk, C.: "Controlling Virtual Worlds Using Interaction Spheres", In: Vidal, Creto Augusto (Ed.) u.a.; Brazilian Computer Society (SBC) u.a.: Proceedings of the 5th Symposium on Virtual Reality, pp. 3-14, Fortaleza, CE, Brazil, 2002.

Sá, V.; Dähne, P.: "Accessing Financial Data through Virtual Reality", In: Figueiredo, Antonio Dias de (Ed.) u.a.: Proceedings of 3^a Conferência da Associação Portuguesa de Sistemas de Informação, Coimbra, 2002

VRML Specification on-line, <http://www.vrml.org/>.

Wu L., Oviatt S., Cohen P.: "Multimodal Integration – A Statistical View", IEEE Transactions on Multimedia, 1(4):334-341, 1999.

Supporting Direct User Interventions in Exception Handling in Workflow Management Systems

Hernâni Raul Mourão

Escola Superior de Ciências Empresariais do
Instituto Politécnico de Setúbal /
LASIGE - Laboratório de Sistemas de
Informação de Grande Escala,
Campus do IPS – Estefanilha
2914 – 503 Setúbal
hmourao@esce.ips.pt

Pedro Antunes

Faculdade de Ciências da Universidade de
Lisboa,
LASIGE - Laboratório de Sistemas de
Informação de Grande Escala,
Campo Grande, Edifício C8, Piso 1
1749-016 Lisboa
paa@di.fc.ul.pt

Abstract. We developed a framework to handle exceptions in WfMS. Specially, unexpected exceptions, which are situations not predicted during the design phase, and require human involvement. A good characterization of the exception is needed to help the user in the identification of the solution(s) from an available tool kit: redesigning the flow, ad hoc executing the affected tasks, and manipulating engine status. The proposed characterization results from integrating operational, tactical and strategic perspectives over unexpected exceptions. An open source platform was selected to establish a test base on which the framework will be tested. The framework will be implemented in one company and data from another company will be used for simulation.

Resumo. Desenvolvemos uma plataforma para tratar exceções em sistemas de gestão de fluxos de trabalho. Em particular para as exceções não esperadas, que correspondem a situações não previstas durante a fase de modelação e necessitam do envolvimento do operador. Uma boa caracterização da exceção é importante para auxiliar o operador na escolha da solução(ões) num conjunto de ferramentas disponibilizadas: alterar a definição do fluxo de trabalho; executar de forma ad hoc as tarefas afectadas; ou manipular o estado do motor. A caracterização proposta resulta da integração das perspectivas operacional, tática e estratégica sobre exceções não esperadas. Foi seleccionado um projecto “*open source*” para definir um ambiente onde a plataforma possa ser testada. O projecto será implementado numa empresa e dados recolhidos de uma outra empresa serão utilizados para simulação.

1 Introduction

Workflow exceptions are situations not predicted in the workflow model or when there exists a deviation from the model and the real world [Luo 2001]. Workflow exceptions have been accounted for almost half of the total working time in an office [Saastamoinen 1995].

Up to now workflow exceptions have mostly been addressed in a systems perspective, even though a consensus seems to arise on the necessity of involving the user in the recovery of a specific kind of exceptions. Therefore we developed a framework aimed to fulfill this identified gap. The framework is built around the idea that human intervention must be supported with quality information about the problematic situation, status of the workflow engine and possible recovery solutions.

Considering the work in progress, this paper currently offers two contributions to exception handling in WfMS: (1) an integrated perspective over exceptions and exception handling, considering three different organizational levels (strategic, tactic and operational) affected by exceptions and respective human roles in recovery (design changes, ad hoc execution, and

engine status manipulation); and (2) a set of WfMS components necessary to help and support these roles: event handler, situation awareness, problem characterization and recovery toolkit. These components are currently being validated on an open source platform.

The paper is organized as follows. Section 2 reviews the literature and establishes the grounds for the proposed framework. Section 3 presents the integrated perspective, where different exception handling approaches are mapped to the organizational levels where they occur and are handled. In section 4 we describe the components that constitute our solution. In section 5 we describe the project current status. In section 6 we identify the main current contributions and discuss the expected results.

2 Literature Review

[Eder and Liebhart 1995] characterize failures and exceptions in a single dimension, encompassing two types of failures and two types of exceptions:

- Basic failures – Associated with failures on the systems supporting the WfMS (e.g., operating system, database management system and network failures);
- Application failures – Failures on the applications invoked to execute tasks (e.g., unexpected data input);
- Expected exceptions – Events that can be predicted during the modelling phase but do not correspond to the “normal” behaviour of the process. These exceptions can happen frequently and can cause a considerable amount of work to process (e.g., a flight cannot be booked because it is already booked up);
- Unexpected exceptions – When the semantics of the process is not accurately modelled by the system (e.g., changes in rules, a structural change in the organizational environment, or a change in the order processing of a very important client). This type of exception cannot be predicted at the modelling stage and may require human intervention or even force the system to stop.

[Eder and Liebhart 1996] suggests an escalating concept to transform the failures that cannot be resolved in the level where they occur into exceptions. This way, the classification can be reduced to exceptions.

[Chiu 2000] based on research work from [Eder and Liebhart 1995], combines the above view with another orthogonal characteristic described as “exception source.” The exception source can either be internal or external to the workflow. External sources are consequences of failures in components that participate in the WfMS such as operating systems, database management systems, software applications, machines and equipment, or operations in external organizations. Internal sources are directly related to workflow management issues, e.g., being unable to find a component to execute a task, or a missing a deadline.

A similar classification is adopted by [Worah and Sheth 1997] but without any distinction between expected and unexpected exceptions.

Next, we will analyse the expected and unexpected exceptions in more detail, primarily basing our comments on results produced by the WfMS research community. Later, we will present a broader perspective, oriented towards the organizational semantics. Finally, to complete this review, we describe the exception handling strategies found in the literature.

2.1 Expected and Unexpected Exceptions

Expected exceptions are cases that can be predicted during the modelling stage but do not correspond to the “normal” process behaviour. Some mechanisms should be implemented to handle these situations because they can occur frequently [Eder and Liebhart 1995]. The example used by [Casati, et al. 1999] to explain expected exceptions is a car rental company. When a customer phones to report an accident, the company has to re-schedule all future rentals for that specific car and make the necessary arrangements to repair the damaged car. The “normal” behaviour of the process should have been the returning of the car to the company, as planned, while the accident corresponds to a deviation or an “occasional” behaviour: an expected exception.

[Casati 1998] identifies four classes of expected exceptions, according to the events that generate them:

- Workflow exceptions – On starting or finishing a task;
- Data exceptions – When workflow relevant data is changed;
- Temporal exceptions – On the occurrence of a given timestamp (e.g., car not returned on time, check every week on car repair situation);
- External exceptions – Activated by external events (e.g., the above example of a customer reporting an accident).

Also important is to identify whether the exception is synchronous or asynchronous to the evolution of the process. The only synchronous class of exceptions is the workflow type since it happens after a change in the process state. Asynchronous exceptions are difficult to model with flow diagrams because it is not possible to insert them in the process evolution.

Unexpected exceptions result from inconsistencies between process modelling in the workflow and the actual execution [Casati 1998]; and result from incomplete or design errors, improvements or changes in the business manoeuvre or quality and customer satisfaction issues not known during the modelling stage. This type of exceptions is frequent in highly complex or dynamic environments. Usually, unexpected exceptions force the process to change to a halt state and require human intervention [Heinl 1998].

In situations where this kind of exceptions occurs frequently, one should consider redesigning the workflow model, if it is out of date, or adopting different technologies based on collaborative work or adaptive workflow systems [Casati, et al. 1999].

2.2 Organizational Perspective

[Saastamoinen 1995] proposes a taxonomy based on the organizational semantics associated to exceptions. This work defines a set of base concepts necessary to construct a consistent conceptual framework and fundament the characterization of organizational exceptions. Next we will summarize some of the main concepts defined and later the taxonomy is described.

Roughly we could say that any situation for which the organization has no rule is an exception. This is the line of thought presented in this work. First, the concepts of rule and event are defined and then the rules to handle different types of events which are the base for the organizational procedures. Finally, a concrete definition of exceptions is presented.

Event is defined as a piece of work to be handled by an office that is caused by a detected phenomenon or a state of the system. An event type is a specification of the common features found in certain events. The concept of rule plays an important role in this work as the bases for exception definition. The author defines rule as a formal way of specifying a recommen-

dation, a directive or a strategy, expressed as “IF premise THEN action” or “IF condition THEN action” and then, event handling rule, as an orderly set of rules that precisely and accurately guides an actor in handling certain types of events. An office procedure has a broader scope and is defined as an orderly set of event handling rules aimed at reaching a specified goal in an office by directing an entire handling of an event.

Once the association between rules and event handling rules is established the concepts of normal event, main line, variation, main line event, and variation event constitute the last step towards exception definition. A normal event is an event with the necessary rules for identifying and for handling. A main line is an office procedure for the most predictable events of a certain type and a variation is work that is added to the main line. Note that a variation is an office procedure for less predictable but still known events of a certain type. A main line event is a normal event that can be handled by the main line while a variation event is an event not handled by the main line but handled by another office procedure.

The environment for exception definition is now created. An exceptional event (exception) is an event that neither the main line or the variation procedure can handle.

As discussed by the author, the definition of rule presented is narrower than the variety of rules that exists in an organization, e.g., good business practice, precepts, regulations, conventions, principles, guiding standards, rules of thumb, and even maxims. These kinds of rules are not precise enough to establish a consistent ground to serve as the basis for a framework even though they represent all the knowledge of the organization. Therefore, [Saastamoinen 1995] uses the rule-based knowledge presentation technique for the definition of rules. In a way, procedures and functions are a kind of rules as they guide the actions of a computer.

[Saastamoinen 1995] developed a taxonomy from empirical studies. A special attention was made on the social and financial impacts of exceptions. Six different criteria are used to classify exceptions:

- Exceptionality – Difference between the exceptional and “normal” event;
- Handling delay – Time elapsed between the exception identification and handling;
- Amount of work – Extra work required to handle the exception when compared to the normal event;
- Organizational influence – Number of people involved in the exception;
- Cause – A measure of the importance of the reason for the exception;
- Rule impact – Number of changes in the organization’s rules due to the exception.

Three classes of exceptions were identified according to exceptionality: established exceptions, otherwise exceptions, and true exceptions. Established exceptions occur when rules exist in the organization to handle an event but the right ones cannot be found. Otherwise exceptions occur when the organization has rules to handle the normal event but do not apply completely to the case. Finally, true exceptions occur when the organization has no rules.

According to the organizational influence criteria, exceptions can also be classified at employee, group and organizational level. Employee exceptions are situations that affect only the work of one person, group exceptions, affect a group of people working within the same process, in the same kind of job or in the same process, and organization exceptions, affect the work of persons in more than one department or project in the organization.

A relationship between this approach and the WPMC definitions is established to provide integration with concepts widely used. According to the WPMC, an organizational model represents organizational entities and their relationships; it may also incorporate a variety of attributes associated with the entities, such as skills and roles. The organizational model can

be represented in the system and should be the basis for the characterization of the organizational influence area.

The WPMC definition of a business process is a set of one or more linked procedures, which collectively realize a business objective or policy goal, normally within the context of an organizational structure defining functional roles and relationships. A workflow is the automation of a business process, in whole or part, during which documents, information or tasks are passed from one participant to another for action, according to a set of procedural rules. The definition of office procedures in [Saastamoinen 1995] and the associated definition of exception seem to be coherent with this approach.

On another work oriented towards organizational semantics, [Strong and Miller 1995] defines five alternative perspectives on the nature of exceptions: random event; error; political system; total quality management; and human-computer system. It is important to realize that this classification mixes causes and solutions. We will discuss solutions in more detail in the next session. Note also, that apart from the random event, [Saastamoinen 1995] integrates the other four criteria so we will only consider his work in the future. [Strong and Miller 1995] mentions random event only for completeness, considering that it cannot be predicted nor can efficient procedures be developed to handle them. Since

2.3 Exception Handling

As it is recognized by most literature, predicting any possible cause of failure or exception during design is very difficult or even impossible and makes the system very complex and hard to manage [Eder and Liebhart 1998; Dayal, et al. 1990; Casati 1998; Klein and Dellarcas 2000]. Therefore, being prepared to deal with failures and exceptions at execution time is a critical factor for WfMS success.

A significant effort has been invested so far to deal with failures and exceptions in a similar way of traditional DBMS, i.e., the transaction processing mechanism. However, task semantics in WfMS far exceeds the transaction models. E.g., if a call to a customer fails because s/he is not reachable, nothing else needs to be done at the moment; maybe trying later will be enough. Organizations are complex environments and the traditional approach based only over data integrity and consistency does not constitute a complete ground to base a framework able to resolve these problems [Worah and Sheth 1997].

Nevertheless, in a primitive approach, we could rely on the system that supports the WfMS. In fact, most of the commercially available DBMS on the market implement the necessary mechanisms to react in case of failure, returning the system to a coherent state and enabling forward execution [Casati 1998].

On the other hand, a typical task can span over a long period of time, and in heterogeneous, autonomous and distributed environments there are tasks implemented by equipment/tools that do not run in transactional environments. This is a more complex environment where typical DBMS solutions are not adequate.

The error handling semantics of traditional transaction processing systems is also too rigid for workflow [Worah and Sheth 1997; Luo 2001]. The handling of system errors based on transactional approaches offer, in general, extreme and expensive solutions in terms of lost work and should be avoided. System errors should then be treated as expected exceptions [Casati 1998].

Some suggestions to overcome these problems consider relaxing the ACID properties and incorporating compensation mechanisms for backward recovery and forward execution, incorporating the flexibility required for workflow systems [Luo 2001]. Several researchers are

working in these issues [Eder and Liebhart 1998; Dayal, et al. 1991; Georgakopoulos, et al. 1996; Krishnakumar and Sheth 1995]. A good survey of this area, which became known as Extended Transaction Models (ETM), is presented by [Worah and Sheth 1997].

Some other studies have proposed the adoption of dynamic and adaptive workflow systems to react to exceptions during workflow execution. The operator, on the presence of an exception, could change the system by either creating a new path for the exceptional process or change all the processes running on the system to the newly created path. Some studies have been carried out to keep the system's consistency and correctness during/after the change, e.g., [Ellis, et al. 1995; Aalst 1999; Reichert and Dadam 1998; Myers and Berry 1999].

[Dayal, et al. 1990; Dayal, et al. 1991] recognized the fixed control sequence and the rigid compensation policy of the ETM approach and developed the Event Condition Action (ECA) rules to decouple the detection and handling of exceptions from the system itself and to increase flexibility. [Casati, et al. 1999] describes a system to deal with expected exceptions based on ECA rules. The language Chrimera-Exc is used to specify exceptions and augment the WfMS characteristics to automatically detect and handle expected exceptions.

[Luo, et al. 2002] describes a system using a Case Base Reasoning (CBR) scheme to derive patterns for exception handling. A measure of similarity is computed to identify the nearest pattern in the knowledge base with the current exception and determine the appropriate action to handle the specific case. The system implements a user intervention mode to allow handling totally new cases or defining a new action for a specific case.

[Klein and Dellarocas 2000] developed a knowledge-based approach that incorporates "sentinels" in the system model to automatically identify exceptions. During execution time, when an exception is detected, it is matched to a handbook's of knowledge and an appropriate action is proposed to the operator. The focus of this approach is mainly on coordination processes rather than core processes.

Despite all these efforts to automatically handle exceptions, the majority of authors involved in research recognize the limits of the proposed solutions. They either recognize the importance of interrupting the process and integrating some manual mechanism and then send the process back to workflow control [Eder and Liebhart 1998; Klein and Dellarocas 2000], or explicitly state that in some situations the role of humans is crucial to collect process specific data that is not available to the workflow system.

It currently seems impossible to include task specific semantics within a generalized recovery framework because task behaviour is orthogonal to workflow process [Worah and Sheth 1997]. [Casati 1998] clearly states that unexpected exceptions should be handled by humans, since they have not been predicted during systems modelling. [Luo, et al. 2002] also recognizes the necessity of human involvement when an exception handler cannot be automatically obtained.

According to [Heinl 1998], the WfMS has only the information about the process that was defined during the modelling phase, and has no possible way of identifying the case when the real business process differs from the model. The author even states that only a human with an overall view can identify the discrepancies and find the way to handle the case.

To complete this review, it is important to mention two other approaches introducing a broader perspective over workflow exceptions. [Guimarães, et al. 1997] proposed an integrated architecture of formal coordinated processes with informal cooperative processes. When the system identifies that it is not able to proceed with process execution, it gathers all the information concerning the specific situation and generates a flow interrupt. The interrupt triggers a tool able to select the best suitable cooperative technique to handle the specific case. [Saastamoinen 1995] presents an approach focussed on organizational semantics. Petri

nets, outside the scope of the WfMS, define the reactions to the various types of exceptions, and should be interpreted as a global organizational reaction to exceptions.

3 Integrated perspective

From the above discussion we can assume that accounting for all possible exceptions requires an integrated approach, where different levels of the organizational system affected by exceptions must be involved. Moreover, the different techniques proposed in the literature should be considered if a complete framework is to be developed. These techniques have a specific scope of application with a correspondence in the level of the organizational system where they should be taken into account. We propose the framework shown in Figure 1.

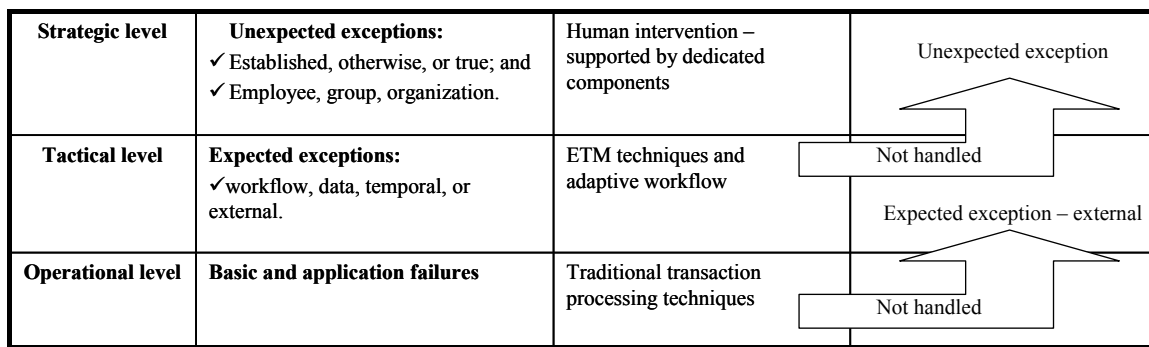


Fig. 1 – Different levels and corresponding exceptions

Depending on the cause and impact of the abnormal situation, a suitable recovery mechanism at the most adequate level should be invoked. These relationships are represented in the second and third columns of figure 1. On the other end, a propagation mechanism is foreseen to enable the exception/failure to be handled in a different level. As described below, the propagation mechanism can be automatically triggered by the system or manually, by an operator, which identifies the necessity to change the level where the exception must be handled. The arrows in figure 1 represent the propagation mechanism.

Another key issue in the integrated perspective is to associate all exceptions to a responsible person. This should prevent the exceptions to get “lost” in the system with no one being responsible for them. The propagation mechanism can also change the responsible person associated with the exception.

Next, we will describe the levels of the organizational system where the different techniques found in the literature to handle exceptions should be invoked. The propagation mechanism is triggered whenever the corresponding layer cannot handle the exception. Then we describe our approach to human involvement in dealing with situations not solved by the previous techniques. At the end of this section we compare some methodologies found in the literature to involve human with our integrated perspective.

The operational level provides an environment for handling basic and application failures only, where the traditional transaction processing techniques can be sufficient to bring the system back to a coherent state and continue execution without human intervention. Whenever these techniques are not able to solve the problem, the event is propagated to the tactical level and the failure is converted into an external exception.

At the tactical level, the WfMS may automatically handle exceptions in many different ways. For instance, using ETM techniques [Eder and Liebhart 1998]. The extensive work done on adaptive workflow, which falls in this level, should increase the system flexibility and augment its adaptation to real world situations in handling, mainly, the expected exceptions. At this level, human contribution is possible but limited to producing the information necessary to have the system applying the exception handling techniques (e.g. compensation, retry, ignore, etc). The works developed by [Casati, et al. 1999; Luo, et al. 2002; Klein and Dellarocas 2000] should also be considered at this level as they contribute to expected exceptions handling.

Eventually, if none of the techniques implemented at this level are able to handle the event, it should be propagated to the strategic level, where the attention of a human operator to an unexpected exception is raised.

When the system has no possible way of identifying the abnormal situation the exception is directly reported by an operator and classified as unexpected exception.

Even though several authors present some ideas on how to incorporate human involvement in exception handling, we believe that the problem has not yet been completely addressed. [Eder and Liebhart 1998] describes the idea of a system that incorporates human intervention in two possible modes: (1) ad hoc extension, where the user can suspend the execution of a task and choose alternative paths or change the execution model; and (2) ad hoc refinement, where the user interrupts the task execution to execute one or more activities and later on proceeds with the interrupted task. Even though the authors integrate this model in their framework, we believe that it should be completed in terms of the tools/methodologies available to help and support the user manipulating the environment in which s/he operates.

In [Luo, et al. 2002], a user-interface is provided to assist human intervention. The exceptions tagged as requiring human intervention come to the attention of an expert that can choose one candidate solution within the system proposals, or rewrite a new solution. In order to react to unexpected exceptions, the system is notified by an external signal and generates an internal exception event. However, it is not clear how this mechanism is implemented neither how the system can deal with any particular type of unexpected exception. Moreover, it is not foreseen any assistance mechanism to provide a general view of the situation nor any support tool to change the model.

Note that although human participation is considered at both the tactical and strategic levels, the strategic level envisages a more dramatic intervention in the WfMS. This will be discussed in more detail in the next section.

4 Our Approach to a Solution

We now propose a system that supports human intervention to handle unexpected exceptions. The interaction between the WfMS and the user is supported by the following components:

Event handler. This component is responsible for launching the recovery process whenever an exception is detected and human intervention is required. It interacts with the user in order to show any upstream and downstream exception propagations related with the exception (e.g., a basic failure previously propagated as an external exception, or a group exception subsequently propagated as an organizational exception).

Situation awareness. This component is responsible for gathering and displaying to the user generic data about the workflow and engine status associated with the exception. In par-

ticular, it lists all tasks defined by the workflow, their status and other details, like task goals. This component also allows the user to identify which tasks are affected by the exception.

Problem characterization. This component uses the criteria defined by [Saastamoinen 1995] to obtain from the user the information necessary to classify an exception as a true, established or otherwise exception. It also allows the user characterizing the organizational influence of the exception (employee, group or organizational level).

Recovery Toolkit. Based on the situation awareness and problem characterization, this component offers a collection of pre-defined actions, which can be combined by the user to manipulate the process design, process execution and engine status. The following collection of pre-defined actions is actually considered:

- Engine status – Start/terminate or suspend/continue several tasks; propagate the exception to other users; manipulate any other information that is necessary, such as process relevant data or workflow participants;
- Process execution – Support ad-hoc extensions and ad-hoc refinements to the process instances affected by the exception; apply design changes according to specification;
- Design – Change one or several process definitions, either affected by this exception or not; define how and when the modifications are applied (one/all instances, immediately or after completion).

Note that the user can execute, in any order, multiple actions in each one of the above three categories. The heuristic used for proposing the best solutions according to the characterization of exceptions is the following: otherwise exceptions are commonly handled by ad-hoc extensions or refinements applied to the affected process instances; established exceptions can be handled by instantiating already available process definitions; and true exceptions require design changes and possibly cascading the exception to other users. If, in the first two cases, the organizational influence of the exception is all the organization, the approach must be coordinated with other areas.

5 Project Status

We selected OpenSymphony ["The OpenSymphony project" 2001] to implement this project. OpenSymphony is an open-source platform based on J2EE technology.

The project consists of Enterprise Java Beans (EJB) for functional components and web-tier components for front end – all components are platform, database and application server independent.

Figure 2 represents the 5 component layers of the OpenSymphony architecture. The Foundation layer comprises a set of core libraries used by the rest of the suite. The Business components implement the business functions, in special the workflow. Our Integrated Perspective is also implemented in this component. The Interaction layer allows web-based applications to interact with system components, whereas the Presentation aid in the final presentation on the web. Finally, the application layer is the ready built solutions for real needs.

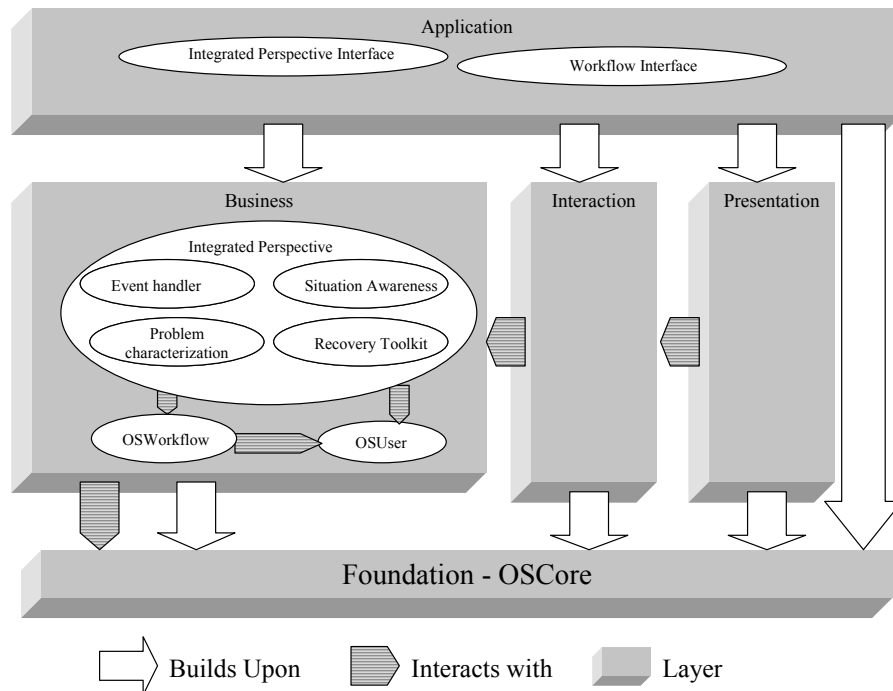


Fig. 2 – OpenSymphony components architecture

The workflow specification uses a XML file rather than a graphical user interface, which gives more flexibility to cope with changes in the process definitions. The workflow engine is based on the concept of a finite state machine where each state is represented by a combination of StepID and status. A transition cannot be fired without an action occurring first. The engine supports Java-based functions (class loader, classes retrieved via JNDI, local EJB, and remote EJB), BeanShell scripts, and Bean Scripting Framework (BSF) scripts. Another type of functions, the Trigger Functions, are triggered by outside sources and run under the system context.

These base components of the framework, together with its capability to coexist with other frameworks oriented our decision to choose this platform.

Figure 3 shows the user interface implementing our exception handling approach. The user interface highlights the four main components necessary to support exception handling.

The user can analyse the list of exception propagations in the Event Handler area, if any. The interface allows analysing upstream and downstream exceptions. Note that downstream exceptions only exist if the user decides to propagate the current exception.

In the Situation Awareness area, the user sees a list of all the tasks running on the system (left side). Selecting the affected ones and pressing the right arrow button, they are transferred to the right side. This way it is possible to identify all the currently running tasks that are affected by the exception. Whenever necessary, the specific task details can be viewed/edited.

The Problem Characterization area enables the selection of the exception characteristics according to our approach. At this stage only the exceptionality and organizational influence are considered. The other four characteristics (handling delay, amount of work, cause, and rule impact) can only be defined at the end of the exception handling, for historical records. In the organizational influence zone there is an area that changes according the selection, allowing to select an employee, group leader or organizational manager, respectively. This way someone is always associated with an exception. There is also the possibility of sending emails to everyone involved.

In the Recovery Toolkit area the user can decide the recovery action(s) to implement on this specific case. As noted before, any combination of the above pre-defined actions can be used for the particular case, although the system suggests some pre-defined actions. On the type, action, and parameters columns the user selects one option from a list.

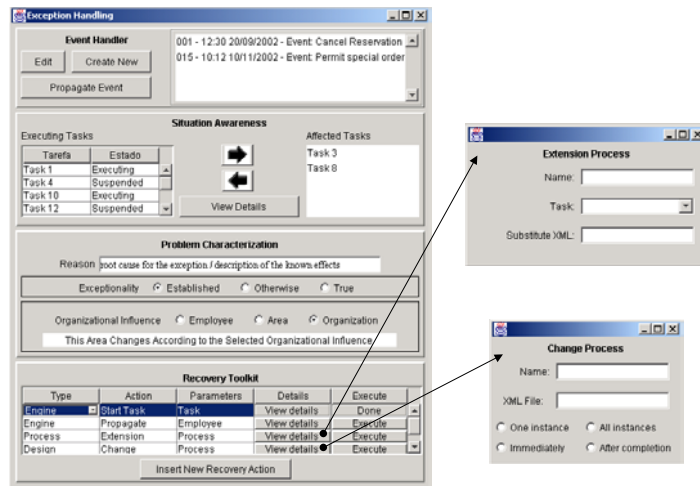


Fig. 3 – Interface for exception handling

A design company was selected to test the prototype in real world situations. The processes were studied and are being modeled. The framework is going to be implemented in the company and the exceptional situations together with the company reactions will be identified and registered. This test will enable the adjustment of the framework to operator response and test it's applicability.

Another company is being selected to collect a set of situations that will be used in a simulation study. The data from this study will complement the data from the implementation augmenting the adjustment of the framework concrete situations.

6 Expected Results

It is expected that the implementation and the simulation will raise issues concerning the ability of the framework in dealing with adjustment of the modeled processes to changing environments and to situations not predicted in the modeling stage. The robustness issues associated with the run time changes to the processes design will be a major concern and the support to the user during this changes will be a critical factor to maintain the coherence of the system.

The taxonomy used to classify exceptions and the level in the organizational system that should deal with the abnormal situation should also be a subject of deeper investigation as they have a critical role in the choice of the best handling mechanism.

With this work we also expect to improve the OpenSymphony platform, allowing the workflow components to cope with unexpected exceptions.

References

- Aalst, W. v. d., "Generic workflow models: how to handle dynamic change and capture management information," Proc. of IFCIS, International Conference on Cooperative Information Systems, CoopIS '99, IEEE International, 1999, 115 -126.
- Casati, F., Models, Semantics, and Formal Methods for the Design of Workflows and Their Exceptions, PhD Thesis, Politecnico di Milano, 1998.
- Casati, F., S. Ceri, S. Paraboschi, and G. Pozzi, "Specification and Implementation of Exceptions in Workflow Management Systems," ACM Transactions on Database Systems, 24, 3 (1999), 405-451.
- Chiu, D. K., Exception Handling in an Object-Oriented Workflow Management System, PhD Thesis, Hong Kong University of Science and Technology, 2000.
- Dayal, U., M. Hsu, and R. Ladin, "Organizing Long-Running Activities with Triggers and Transactions," Int. Conf. on Management of Data (SIGMOD'90), Atlantic City, NJ, USA, 1990.
- Dayal, U., M. Hsu, and R. Ladin, "A Transactional Model for Long-Running Activities," 17th Int. Conf. on Very Large Data Bases (VLDB'91), Barcelona, Spain, 1991.
- Eder, J., and W. Liebhart, "The Workflow Activity Model WAMO," Int. Conf. on Cooperative Information Systems, Vienna, Austria, 1995.
- Eder, J., and W. Liebhart, "Workflow Recovery," 1st IFCIS Intl. Conf. on Cooperative Information Systems (CoopIS'96). Brussels, Belgium: IEEE International, 1996, 124 - 134.
- Eder, J., and W. Liebhart, "Contributions to Exception Handler in Workflow Management," Int. Conf. on Extended Database Technology (EDBT'98), Workshop on Workflow Management Systems, Valencia, Spain, 1998.
- Ellis, C., K. Keddara, and G. Rozenberg, "Dynamic change within workflow systems," Proc. of conf. on Organizational computing systems. Milpitas, CA, USA: ACM Press, 1995, 10-21.
- Georgakopoulos, D., M. Hornick, and F. Manola, "Customizing Transaction Models and Mechanisms in a Programmable Environment Supporting Reliable Workflow Automation," IEEE Transactions on Knowledge and Data Engineering, 8, 4 (1996), 630-649.
- Guimarães, N., P. Antunes, and A. P. Pereira, "The Integration of Workflow Systems and Collaboration Tools," in A. K. Dogaç, Leonid Ozsu (Eds.), Advances in Workflow Management Systems and Interoperability, Instambul, 1997.
- Heinl, P., "Exceptions During Workflow Execution," Int. Conf. on Extended Database Technology (EDBT'98), Workshop on Workflow Management Systems, Valencia, Spain, 1998.
- Klein, M., and C. Dellarocas, "A Knowledge-Based Approach to Handling Exceptions in Workflow Systems," Computer Supported Cooperative Work, 9, 3 (2000), 399-412.
- Krishnakumar, N., and A. P. Sheth, "Managing Heterogeneous Multi-system Tasks to Support Enterprise-wide Operations," Journal on Distributed and Parallel Database Systems, 3, 2 (1995).
- Luo, Z., Knowledge Sharing, Coordinated Exception Handling, and Intelligent Problem Solving for Cross-Organizational Business Processes, PhD Thesis, Dep. of Computer Sciences, University of Georgia, 2001.

- Luo, Z., A. P. Sheth, K. J. Kochut, and I. B. Arpinar, Exception Handling for Conflict Resolution in Cross-Organizational Workflows, LSDIS Lab, Computer Science, Un. of Georgia, 2002.
- Myers, K. L., and P. M. Berry, "At the Boundary of Workflow and AI," proc. of the AAAI-99 Workshop on Agent-Based Systems in The Business Context held as part of AAAI-99, 1999.
- The OpenSymphony project. (2001, 20-04-2001), [Http://www.opensymphony.com](http://www.opensymphony.com).
- Reichert, M., and P. Dadam, "ADEPTflex - Supporting Dynamic Changes of Workflows Without Loosing Control," Journal of Intelligent Information Systems, 10, 2 (1998), 93-129.
- Saastamoinen, H., On the Handling of Exceptions in Information Systems, PhD Thesis, University of Jyväskylä, 1995.
- Strong, D. M., and S. M. Miller, "Exceptions and Exception Handling in Computerized Information Systems," ACM Transactions on Information Systems, 13, 2 (1995).
- Worah, D., and A. P. Sheth, "Transactions in Transactional Workflows," in S. K. Jajodia, Larry (Eds.), Advanced Transaction Models and Architectures, Kluwer Academic Publishers, 1997.

TECNOLOGIA E APLICAÇÕES

Aplicações Móveis de Valor Acrescentado: um caso prático

Alexandre Rocha Oliveira Pedro Miguel Figueiredo Adérito Fernandes Marcos^{1,2}
Centro de Computação Gráfica¹
Rua Teixeira de Pascoais, 596
4800-073 Guimarães
{ Alexandre.Oliveira, Pedro.Figueiredo, Aderito.Marcos } @ccg.pt

Rui João José
Universidade do Minho²
Dep. de Sistemas de Informação
Campus de Azurém
4800-058 Guimarães
{ rui, marcos } @dsi.uminho.pt

Resumo

As tecnologias móveis estão a passar por um processo de massificação, assumindo-se como ferramentas a serem usadas em qualquer situação ou local. A rápida evolução deste tipo de tecnologias, irá permitir atribuir um elevado nível de mobilidade a serviços e conteúdos remotos, fazendo uso de dispositivos móveis como PDA's ou telefones móveis. No presente artigo será apresentado um caso prático de uma aplicação móvel de valor acrescentado desenvolvida no contexto de um projecto de I&D denominado de VADE - Value Added Environments for Dynamic Support to Location-Based Services in UMTS Networks.

1. Introdução

O avanço registado nas tecnologias móveis permite que actualmente seja possível aceder a informação, em tempo real, independentemente do local onde o utilizador se encontre. Existem diversos tipos de dispositivos móveis, tais como PCs portáteis, telemóveis, ou PDAs (Personal Digital Assistant) com capacidades de funcionamento em redes wireless. A evolução deste tipo de dispositivos atribui-lhes hoje uma importância cada vez mais reconhecida quando se fala de mobilidade. Acrescente-se a este facto o desenvolvimento dos conceitos de LBS (Location Based Services) e estão assim criadas as condições base para o desenvolvimento de sistemas de valor acrescentado que permitirão uma melhor exploração de serviços de âmbito local baseados na localização. Estes serviços caracterizam-se pela sua associação a um determinado espaço físico, disponibilizando diferentes serviços e conteúdos à medida que determinado utilizador se desloca no espaço físico associado ao ambiente de valor acrescentado.

2. Estado de Arte

Existem actualmente duas grandes abordagens para suportar serviços baseados na localização. A primeira é a utilizada no âmbito de um operador móvel e baseia-se na capacidade, cada vez mais comum, das redes móveis determinarem a posição dos terminais [ETSI99]. Um *Gateway Mobile Location Centre* (GMLC) gere essa informação e disponibiliza-a às entidades, normalmente geridas pelo mesmo operador, que estejam interessadas. Um serviço baseado na localização pode ser assim criado através da instalação de um servidor com informação geo-referenciada e acesso ao GMLC. Quando chega um pedido de informação, o servidor obtém do GMLC a indicação da posição do terminal que solicitou a informação e gera uma resposta adaptada a essa localização. A principal limitação deste modelo é criar a expectativa de que um único serviço seja capaz de fornecer informação para

qualquer uma das potenciais localizações do utilizador, implicando assim que uma grande quantidade de informação seja consolidada num único ponto centralizado com todas as consequências que isso representa em termos operacionais e de integridade [OGC00]. O projecto VADE explora um modelo de serviços baseado na localização em que a dependência da localização é suportada não apenas ao nível interno de cada serviço mas também ao nível da descoberta de serviços, permitindo assim a selecção de serviços que sejam específicos da actual localização do utilizador.

A outra grande abordagem para suportar serviços baseados na localização consiste em associar o processo de descoberta dos serviços ao âmbito geográfico de uma rede sem fios como IrDA, bluetooth ou WiFi. Neste caso, a dependência da localização resulta da necessidade de se estar no âmbito de cobertura da rede para poder aceder aos serviços que por ela são disponibilizados.

Esta abordagem é frequentemente proposta como um meio de suportar a interacção entre dispositivos fisicamente próximos ou entre estes e o seu ambiente [Caswell00]. A sua principal vantagem é suportar um acesso distribuído à informação dependente da localização, mas tem a desvantagem de funcionar de uma forma essencialmente ad-hoc, faltando-lhe por isso o enquadramento administrativo normalmente disponível no âmbito dos serviços de um operador móvel.

A investigação desenvolvida no projecto VADE visa combinar estas duas abordagens, criando uma plataforma em que o enquadramento dos serviços prestados por um operador, e.g. portal móvel, autenticação, perfil, ou billing é combinado com serviços de âmbito local geridos por uma entidade terceira.

3. Projecto VADE

O projecto VADE explora o conceito de ambiente de valor acrescentado (*Value Added Environment* - VADE) como um novo modelo para suportar a descoberta e utilização de serviços baseados na localização. Um Vade é visto como sendo um domínio físico e administrativo, por exemplo um aeroporto ou um centro comercial, gerido por uma entidade local e em que serviços disponibilizados localmente são combinados com serviços de operadores móveis para dar uma funcionalidade integrada aos utilizadores que se encontrem no seu âmbito geográfico.

O projecto visa modelar e implementar uma arquitectura para suportar o conceito de VADE no âmbito das redes UMTS. Os principais elementos dessa arquitectura são um mecanismo que permita detectar e autenticar a presença de utilizadores no âmbito do VADE e um mecanismo aberto para suportar a descoberta local de serviços baseados na localização.

A grande vantagem desta abordagem é permitir uma utilização complementar de elementos locais disponibilizados pelo VADE com elementos globais disponibilizados por um operador móvel ou por uma qualquer outra entidade externa. Os elementos globais proporcionam serviços mais genéricos e permitem ao utilizador manter definições comuns. Os elementos locais permitem aumentar a verstatilidade dos serviços prestados e fornecer serviços muitos mais específicos de um determinado ambiente do que alguma vez seria possível com um serviço genérico.

A avaliação da arquitectura será feita com base num protótipo que se encontra em desenvolvimento no campus de Azurém da Universidade do Minho e num conjunto de aplicações que tiram proveito das funcionalidades da plataforma, como seja o Guia Móvel de Apoio a Visitantes (ver cap. 4).

O modelo VADE visa suportar dois tipos diferentes de aplicação dependente da localização. Uma aplicação *vade-aware* é uma aplicação que pode ser usada normalmente fora do âmbito de qualquer VADE, mas que quando utilizada no âmbito de um VADE é capaz de tirar proveito dos serviços disponibilizados pelo VADE de forma a melhorar a funcionalidade prestada ao utilizador. O modelo subjacente é o de que, conforme o utilizador se move, a aplicação vai procurando os recursos locais que necessita para poder prestar a sua funcionalidade ao utilizador. Por vezes a aplicação pode não conseguir esses recursos e a sua funcionalidade ficará temporariamente indisponível ao utilizador. Na maior parte dos casos, contudo, o que acontece é que existem recursos para uma dada localização, mas não são muito específicos. Por exemplo, no caso de uma aplicação de mapas será fácil ter informação sobre a zona onde se localiza um determinado centro comercial, mas para as pessoas dentro do centro comercial, o que provavelmente seria mais interessante era ter acesso a uma planta detalhada desse centro. É este tipo de funcionalidade adicional que um aplicação *vade-aware* deveria ser capaz de proporcionar. Para estas aplicações a entrada num VADE representa uma oportunidade para descobrir novos recursos que serão mais específicos daquele local e que permitirão por isso aumentar o valor da funcionalidade suportada pela aplicação.

Ao contrário de uma aplicação global, uma aplicação local do VADE fornece uma funcionalidade que apenas visa ser utilizada no âmbito do próprio VADE.

Para estas aplicação o grande objectivo é conseguir a sua integração no âmbito do ambiente aplicacional do utilizador, tipicamente o portal de um operador móvel. Ou seja, ao entrar num VADE, o utilizador teria à sua disposição um conjunto adicional de funcionalidades específicas desse VADE, através do portal do qual normalmente é utilizador.

A grande vantagem deste modelo é que, dado que as aplicações são específicas de um VADE, podem mais facilmente ser optimizadas para tirar máximo proveito dos recursos aí existentes e por isso prestar um valor acrescentado maior.

A principal dificuldade é conseguir uma integração adequada de funcionalidades num portal móvel, ou seja conseguir que as novas funcionalidades surjam como novas opções dispersas na estrutura do portal e não apenas como uma lista de funcionalidades locais desligadas daquelas com que o utilizador já está familiarizado.

4. Guia Móvel de Apoio a Visitantes

O guia móvel de apoio a visitantes é uma das aplicações que está a ser desenvolvida pelo Centro de Computação Gráfica, que visa demonstrar a aplicabilidade dos conceitos associados ao projecto VADE. Esta é a primeira abordagem aplicacional da arquitectura VADE e tem como finalidade apoiar os visitantes do Campus de Azurém, através de dispositivos móveis que suportem tecnologias como o Wap, Bluetooth, WiFi ou IrDa. Uma vez localizados num ambiente VADE, os visitantes poderão aceder às funcionalidades apresentadas de seguida, que são parte integrante da aplicação 'Guia Móvel de Apoio a Visitantes', disponibilizada no portal do ambiente VADE.

Com o objectivo de definir um cenário concreto de utilização, foram assim identificadas três funcionalidades que se julgam ser particularmente úteis: encontrar pessoa, encontrar evento, e encontrar local.

A funcionalidade encontrar pessoa, terá como base um serviço remoto, enquanto que todas as outras funcionalidades terão como suporte serviços locais do VADE. A descrição destes pode ser encontrada mais à frente.

A integração da aplicação Guia Móvel de Apoio a Visitantes no ambiente VADE é representada pela figura seguinte:

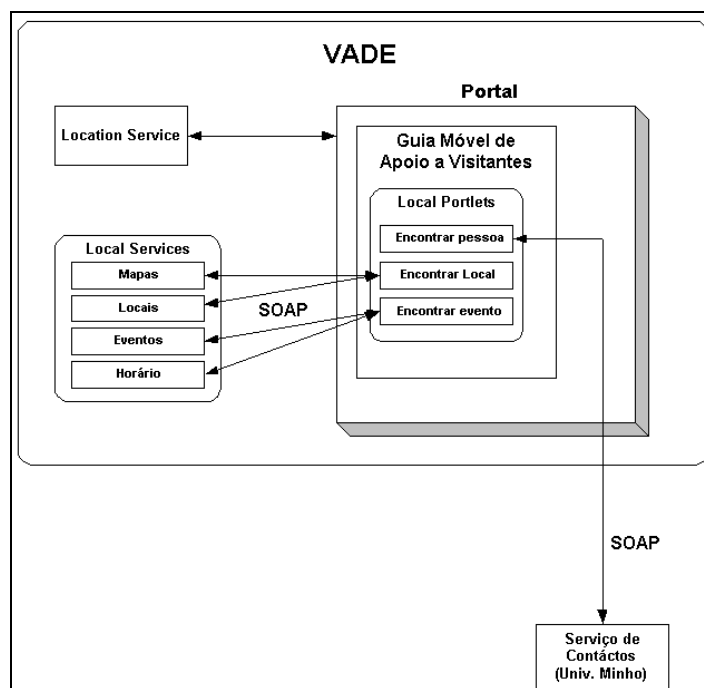


Figura 1 - Modelo Conceptual

Às *local portlets* correspondem as diversas funcionalidades da aplicação e aos *localservices*, os respectivos serviços a usar pelas mesmas. Os serviços estão a ser desenvolvidos de forma genérica por forma a poderem servir de suporte a novas funcionalidades que sejam eventualmente integradas na aplicação aqui descrita.

O *location service* apresentado na Figura 1, é um serviço local do ambiente VADE, não pertencendo assim a nenhuma funcionalidade da aplicação. Este é responsável pela localização dos utilizadores e assume-se como um dos mais importantes serviços da arquitectura, dado que a maioria das aplicações dependem do mesmo. É também da responsabilidade deste serviço, a disponibilização dos serviços associados a uma determinada localização.

No que se refere à questão da comunicação entre funcionalidades e serviços, é usado o protocolo de comunicação SOAP (protocolo baseado em XML que define de forma estruturada o modo como se invocam os serviços remotos).

A forma de interação entre as funcionalidades e os serviços, tem uma abordagem bastante simplista. Depois do utilizador solicitar a informação através do preenchimento de um dos formulários da aplicação, será então invocado o serviço correspondente.

Na construção dos *layouts* são usados *templates*, que são gerados em conformidade com o dispositivo de acesso ao portal. De uma forma geral, no caso de serem usados telemóveis é gerado WML, enquanto que para todos os outros dispositivos será gerado HTML.

4.1 Descrição da Aplicação

Conforme referido anteriormente, a aplicação foi estruturada com base nas três funcionalidades identificadas, que de uma forma isolada ou cooperativa servirão de apoio aos utilizadores durante a sua permanência no ambiente.

De seguida apresentam-se as funcionalidades.

4.1.1. Encontrar Pessoa

Esta funcionalidade tem como objectivo a disponibilização de um conjunto de informações referentes a uma dada pessoa, pertencente ao ambiente VADE.

Esta funcionalidade tem como base um *web service* remoto, disponibilizado na intranet da Universidade do Minho, a sua descrição pode ser encontrada em <http://intranet.uminho.pt/webservices/contactos.asmx>.

A integração desta funcionalidade na aplicação baseia-se em duas abordagens distintas: global, em que os resultados obtidos referem-se a todo o ambiente (não dependem da localização); restrita, em que os resultados obtidos dependem directamente da localização do utilizador.



Figura 2 – Parametrização da funcionalidade

Na primeira abordagem, o utilizador insere apenas o nome da pessoa a procurar (Fig. 2). Como resultado, é obtido o conjunto de todos os nomes que fazem *match* com o nome inserido (Figura 3). A pesquisa será tanto mais refinada quantos os nomes que forem introduzidos. Para visualizar a informação associada a um determinado nome, apresentado como resultado, apenas será necessário fazer a sua selecção. O resultado desta operação pode ser visualizado na Fig. 4.



Figura 3 – Resultado pesquisa;



Figura 4 – Informação disponível sobre o contacto pesquisado

A segunda abordagem para este problema, bastante mais específica que a anterior, passa pelo pedido de contactos referentes ao local onde o utilizador se encontra, por exemplo um departamento, um serviço técnico, entre outros.

Caso o utilizador se encontre num espaço aberto, esta funcionalidade não será disponibilizada. Esta abordagem terá como suporte o mesmo serviço utilizado na abordagem anterior. A diferença fundamental reside na filtragem dos resultados obtidos, tendo como base *o location service*, por forma a disponibilizar apenas os que se relacionam com o local onde se encontra o utilizador. Com base nesta funcionalidade, outras semelhantes poderão ser rapidamente desenvolvidas, como por exemplo uma funcionalidade que permita obter informação sobre um departamento (telefone, fax, morada, etc).

4.1.2. *Encontrar Locais*

Esta funcionalidade irá servir os utilizadores que eventualmente necessitem de orientação no espaço físico associado ao ambiente VADE, nomeadamente aos utilizadores que desejem deslocar-se a um determinado local do ambiente, e não saibam como alcançá-lo.

A solução óptima para um serviço deste tipo é conseguida através da localização absoluta do utilizador no ambiente, que pode ser referenciada num mapa do local onde este se encontra. À medida que o utilizador se movimenta, a sua posição é automaticamente actualizada num mapa apresentado no seu dispositivo. Caso o utilizador pretenda deslocar-se a algum local no ambiente, esse local será igualmente assinalado no mapa. Dado que o seu movimento no espaço é actualizado constantemente, tornar-se-ia fácil desta forma alcançar o local pretendido.

Perante a situação actual do VADE, em que existe uma margem de erro considerável no que se refere à localização do utilizador, não é possível aplicar o tipo de abordagem descrita anteriormente. Logo uma outra abordagem de orientação foi desenvolvida, em que serão apenas disponibilizados mapas estáticos acompanhados por pequenas descrições textuais pré-definidas. O utilizador poderá também aceder a mapas genéricos do local que darão acesso a mapas mais específicos.

Devido a restrições de *display* apresentadas por alguns dos dispositivos móveis, nomeadamente telemóveis, será disponibilizada uma versão textual da funcionalidade, eliminando a opção de visualização de mapas.

A funcionalidade atrás descrita irá recorrer a dois serviços locais distintos: um serviço Mapas, que será responsável pela disponibilização de toda a informação relativa a mapas existentes no ambiente; e um serviço Locais, responsável pela componente textual. Os serviços em causa encontram-se de momento em fase de desenvolvimento.

4.1.3. *Encontrar Eventos*

Toda a informação relativa a eventos a decorrer ou agendados no ambiente, tal como a informação relativa a horários de aulas, será disponibilizada por esta funcionalidade.

A obtenção de informação relativa aos horários, pode ser conseguida através de dois métodos distintos: horário diário de determinado curso e horário semanal de determinada disciplina.

O horário diário de determinado curso disponibiliza a informação horária de determinado curso, dados o dia, ano e semestre, conforme pode ser observado na Figura 5.



Figura 5 – Horário diário de determinado curso

Como resultado são apresentadas as várias disciplinas, juntamente com a informação que lhes está associada, nomeadamente: a sala, o horário, e finalmente o turno, ver Figura 6.



Figura 6 – Resultado de uma pesquisa do horário diário de determinado curso

O horário semanal de determinada disciplina apresenta um comportamento idêntico ao descrito para o método anterior. A diferença essencial reside no tipo de resultado obtido, pois neste caso será apresentada a informação horária semanal de uma determinada disciplina, dados o curso, ano e o semestre, conforme resultado apresentado na Figura 7.



Figura 7 – Resultado de uma pesquisa do horário semanal de determinada disciplina

No caso da pesquisa de eventos, será possível solicitar informação sobre os eventos a realizar ou realizados num dado espaço do ambiente. Contudo, a opção mais natural será a pesquisa de eventos não dependente da localização do utilizador. Esta funcionalidade faz uso de um serviço de horários já desenvolvido, a sua descrição pode ser vista em <http://tomcat.ccg.pt/axis>, e de um serviço de eventos em fase de desenvolvimento.

4.1.4. *Interacção entre funcionalidades*

De forma a maximizar as potencialidades da aplicação, para além de uma utilização isolada, as funcionalidades que a constituem poderão cooperar entre si, como referido anteriormente. Esta cooperação traduz-se na invocação directa de uma funcionalidade a partir de uma outra. Assim sendo, caso um utilizador pretenda encontrar o local de trabalho de determinada pessoa, terá de utilizar a funcionalidade encontrar pessoa. A partir do resultado, poderá ser obtido um mapa onde está especificada a localização de um gabinete, por exemplo.

A mesma situação pode ser criada em relação aos eventos. Depois de obtido o resultado de uma pesquisa sobre eventos, ou horários, será apenas necessário seleccionar o nome do local para invocar a funcionalidade ‘encontrar locais’. Automaticamente será disponibilizado o serviço de orientação anteriormente descrito.

4.2 Tecnologias

A tecnologia usada como base para o desenvolvimento e suporte desta aplicação foi o Jakarta Jetspeed, da Apache, que é uma implementação *open source* de *Enterprise Information Portal*. Esta tecnologia tem por base Java e XML, entre as suas numerosas funcionalidades, destacam-se: implementação do conceito de *portlet*, *template based layout*, customização individual de páginas e *portlets*, gestão de segurança no acesso às *portlets*, os utilizadores podem aceder a um portal via *web browser*, dispositivos WAP ou qualquer outro dispositivo móvel.

As *portlets* podem ser vistas como componentes, em que cada componente é responsável por aceder e disponibilizar informação, seja esta proveniente de um site, de uma base de dados, de um servidor de e-mail, etc.

Do ponto de vista do utilizador, uma *portlet* é uma janela de um portal que disponibiliza um serviço ou informação específica.

Da perspectiva do desenvolvimento aplicacional, *portlets* são módulos que são desenhados para correr num *portlet container* que disponibiliza um ambiente de execução no qual as *portlets* são instanciadas, utilizadas, e destruídas.

Deste modo, tendo em conta as vantagens oferecidas por esta tecnologia, e uma vez que dispõe de todos os mecanismos e funcionalidades que satisfazem os requisitos definidos na especificação do portal do VADE, decidiu-se pela sua utilização.

5. Conclusões e Trabalho Futuro

O desenvolvimento de serviços móveis de valor acrescentado tem ainda um longo caminho a percorrer, encontrando-se ainda em fase de investigação. No que se refere ao VADE, pretende-se que sejam desenvolvidos ainda mais serviços para além dos acima apresentados. Contudo, o principal objectivo é demonstrar que o conceito é explorável em cenários reais onde a computação móvel pode introduzir mais valias, nomeadamente a nível de acesso a informação contextualizada. Dado que a arquitectura VADE se encontra ainda em fase de implementação, não é possível nesta altura traçar um cenário conclusivo sobre o desenvolvimento de aplicações neste contexto. Porém, este projecto já permite afirmar nesta fase que as tecnologias se encontram estáveis e está para breve a massificação de serviços móveis de valor acrescentado.

A integração destes em portais dos operadores móveis será talvez o cenário mais plausível. Contudo, a evolução de standards como o 802.11b poderá vir a ditar novas regras no mercado das infra-estruturas *wireless*.

6. Referências

Caswell, D., P. Debaty. Creating Web Representations for Places. Handheld and Ubiquitous Computing. P. T. a. H. W. Gellersen, Springer. 1927: 114-126, 2000

ETSI TS 101 724 V8.0.0, Digital cellular telecommunications system (Phase 2+); Location Services (LCS); (Functional description) - Stage 2 (GSM 03.71 version 8.0.0 Release 1999).

OGC 2000, In Support of an Open Location Services (OpenLS) Testbed, Open GIS Consortium.

The W3C consortium: Mobile access web page; <http://www.w3.org/Mobile/>.

José, Rui; Adriano Moreira; Filipe Meneses; Geoff Coulson, An Open Architecture for Developing Mobile Location-Based Applications over the Internet, 6th IEEE Symposium on Computers and Communications, Hammamet, Tunisia, 2001.

VADE home page. <http://get.dsi.uminho.pt/vade>.

Utilizando uma Base de Dados XML Nativa aplicada ao tratamento de erros num Sistema de logs

Giovana Mendes

I2S Informática - Sistemas e Serviços¹
Giovana.mendes@i2s.pt

Nuno Alberto P. Silva

I2S Informática - Sistemas e Serviços
Nuno.silva@i2s.pt

Pedro R. Henriques

Departamento de Informática – Universidade do Minho
Campus de Gualtar - 4710-057- Braga – Portugal
prh@di.uminho.pt

RESUMO

Nos dias atuais a maioria dos sistemas informáticos evoluíram para aplicações com interfaces baseadas na WEB. Com o Standard XML sendo cada vez mais utilizado para representar informações nos mais diferentes patamares de aplicações, é necessário que exista uma maneira eficaz para tratar e armazenar esses documentos. Nesta comunicação, iremos demonstrar a solução adotada para tratar documentos XML gerados por um mecanismo de logs de uma aplicação WEB, falaremos do armazenamento de documentos XML, utilizando o caso de estudo da descrição de logs de erros, de modo a facilitar a apresentação desses erros ao utilizador, novamente na interface WEB.

1. INTRODUÇÃO

Nos dias atuais os sistemas informáticos têm a necessidade de evoluir as suas aplicações, e com isso rever o seu funcionamento para poder responder com eficiência e rapidez às necessidades do mercado, que buscam sistemas que estejam disponíveis, por exemplo, na WEB. Com o standard XML sendo cada vez mais utilizado em aplicações na WEB, é necessário que exista tratamento para esses documentos XML e alternativas para os armazenar.

Existem inúmeros trabalhos sobre alternativas de armazenamento de documentos XML [Florescu99 et al.][SousaA02][VieiraH][Runapongsa02]. Estas alternativas estão em discussão no trabalho de mestrado, em elaboração, na Universidade do Minho[MendesG].

¹ I2S Brasil, atualmente alocada à I2S Portugal para realização do mestrado na Universidade do Minho.

Esta comunicação abrange apenas ao armazenamento de documentos xml numa base de dados XML Nativa, demonstrando um caso de estudo em que a nossa investigação foi aplicada.

O caso de estudo foi no âmbito dos seguros, com um caso prático utilizado na I2S Informática - Sistemas e Serviços, e trata-se de um sistema WEB que gera documentos XML com logs de erros; encontrados na transferência dos dados da interface WEB para o sistema AS400.

Iremos começar por fazer uma introdução às bases de dados XML Nativas, depois descreveremos o caso de estudo, com um caso concreto em que uma base de dado XML Nativa foi utilizada.

2. BASE DE DADOS XML NATIVAS

Segundo [DBXML] podemos caracterizar uma base de dados XML Nativa da seguinte maneira:

- Possui um modelo lógico para documentos XML, segundo o qual se deve armazenar e extrair os documentos. No mínimo esse modelo deve incluir elementos, atributos e PCDATA.
- Um documento XML é a unidade fundamental para o armazenamento (lógico), correspondendo na base de dados relacional tradicional ao conceito de linha numa tabela.
- Não é necessário ter uma camada física particular para o armazenamento. A base de dados pode ser construída como relacional, orientada a objetos, em arquivos compactados, etc.

As bases de dados XML Nativas são base de dados destinadas especificamente para armazenar documentos XML. Como outras bases de dados elas têm suporte para transações, segurança, multiutilizadores, linguagens para consultas (XQL, XML-QL...), API's (*application program interface*) para programação (DOM - *Document Object Model* , SAX - *Simple API for XML*) etc., a única diferença é que seu modelo é baseado em

XML enquanto outras base de dados possuem outro tipo de modelo, como por exemplo, o modelo relacional.

Para a escolha de uma base de dados XML Nativa é necessário que se tenham bem claros os objetivos a serem atingidos, e para isso classificar os documentos XML da seguinte forma:

Sistemas orientados a dados:

Nesse tipo de sistema possuímos uma estrutura regular, um esquema para o modelo, e na maioria das vezes o XML é utilizado somente como meio de transporte de dados.

Sistemas orientados a documentos:

Neste caso temos o documento com uma estrutura irregular, não correspondendo a nenhum modelo relacional; como por exemplo, no âmbito da demografia histórica, os casos de registros dos eventos realizados na paróquia: casamentos, batismos, óbitos.

2.1 Arquitetura

2.1.1 Base de dados XML Nativas baseadas em Texto

Nessa arquitetura os documentos XML são armazenados como texto, quer no próprio sistema de arquivos, quer num campo tipo CLOB(*CHAR LARGE OBJECT*) em base de dados relacional ou noutro sistema de textos proprietário. O seu funcionamento é semelhante a uma base de dados hierárquica, e a grande vantagem desta arquitetura é a forma de indexação, pois, assumindo que os dados estão armazenados em bytes seguidos só é necessário uma única pesquisa no índice.

2.1.2 Base de dados XML Nativas baseadas em Modelo

Nesta arquitetura além de armazenar os documentos como texto é construído um modelo deste documento, que também é armazenado. A forma como esse modelo é armazenado depende da base de dados, que pode ser relacional ou objeto-relacional. Esse tipo de base de dados pode ter problemas de desempenho ao realizar busca e retorno de documentos que não estejam no formato que está armazenado, como por exemplo invertendo a posição dos elementos do documento.

2.2 Xindice

Como exemplo de uma base de dados XML Nativa temos o Xindice que é uma base de dados desenvolvida pela Apache², escrita na linguagem Java e *open source*. A linguagem utilizada para consultas é a XPATH [W3C99] e suporta utilização de *API's* de DOM³ e SAX⁴.

A grande vantagem dessa base de dados é a flexibilidade. Portanto, se for necessário controlo sobre a estrutura do documento, é conveniente escolher outro tipo de base de dados.

O projeto Xindice continua em desenvolvimento e atualmente suporta o armazenamento de documentos XML que estão formatados e não guarda nem utiliza nenhum tipo de esquema que permita fazer restrições (por exemplo, DTD's – Document Type definition, ou XML Schema [W3C02]).

2.2.1 Funcionalidades

Abaixo listamos um resumo das funcionalidades dessa base de dados

- **Coleção de Documentos**

Os documentos são armazenados em coleções. As coleções podem ser criadas com documentos do mesmo tipo ou então para armazenar todos os documentos juntos.

- **Linguagem XPATH**

A linguagem para consulta utilizada é a XPATH que está definida na W3C.

- **XML indexado**

Para aumentar a performance das interrogações é possível criar índices para os elementos e para os atributos.

² <http://www.apache.org>

³ <Http://www.w3.org/DOM/>

⁴ <Http://www.saxproject.org/>

- **Linguagem para atualização dos documentos - Xupdate**

Xupdate é uma linguagem baseada em XML, para que seja possível fazer atualizações nos documentos. Essa atualização pode ser feita tanto para uma coleção de documentos como para um documento específico. Essa linguagem está definida em [MartinL].

2.2.2 Estrutura da Base de Dados

A estrutura do Xindice é destinada para armazenar coleções de documentos, que são ordenadas hierarquicamente como num sistema de arquivos.

No Xindice os dados são armazenados em uma instância de base de dados. Essa instância pode conter várias coleções de filhos. Como norma a base de dados é criada como /db, e o caminho para todas as coleções deve iniciar com /db, mas é possível renomear esta instância se for necessário.

3. CASO DE ESTUDO

O nosso estudo foi realizado no âmbito dos seguros, de forma a tratar logs de erros que são gerados por uma aplicação que está disponível na WEB. Este estudo foi um exemplo de como as bases de dados XML Nativas podem ser utilizadas. Como existem diversas alternativas de armazenamento de documentos XML, iremos demonstrar um caso em que a base de dados XML Nativa foi uma boa solução.

3.1 Arquitetura Atual

Para fazer a integração do sistema em AS400/DB2 com o da WEB, foi definida uma arquitetura para que fosse possível um meio de comunicação simples e *standard* entre os dois sistemas. A arquitetura foi definida conforme a figura 1.

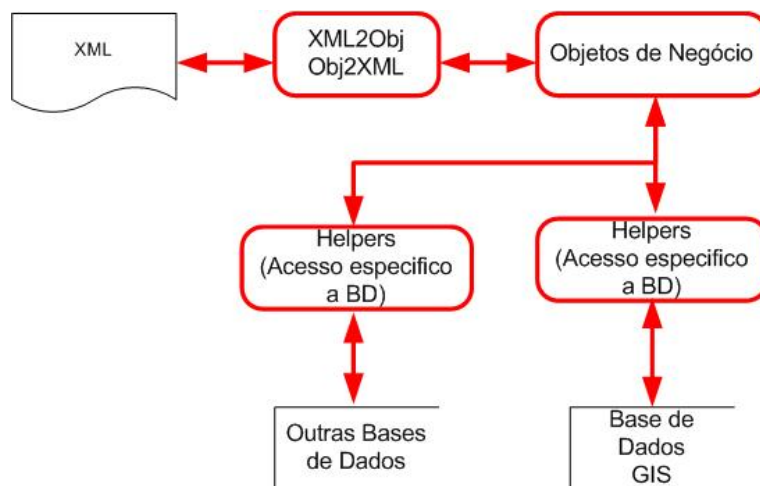


Figura 1 - Arquitetura Atual

Com esta arquitetura carregam-se os objetos de negócios com as informações da base de dados. Já com os objetos carregados as informações são processadas conforme a necessidade.

3.1.1 Conversão de Objetos para XML

Atualmente os objetos de negócio geram o documento XML conforme é mostrado na figura 2.

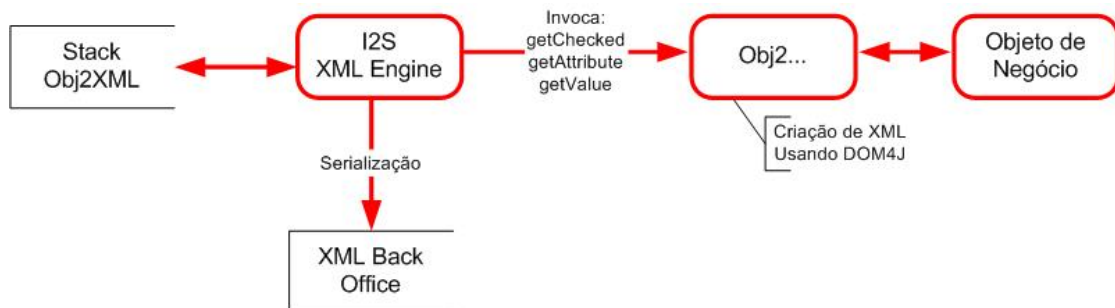


Figura 2 - Conversão de Objetos para XML

A arquitetura funciona da seguinte maneira:

O XML Engine coloca o objeto principal a consultar na pilha de classes percorridas, sendo que para a classe que estiver no topo da pilha, o XML Engine é responsável por invocar os seguintes métodos para cada um dos elementos do XML a criar, acrescentando os respectivos elementos ao XML de consulta:

- *getChecked* – responsável por verificar se um determinado elemento deve ser incluído no XML;
- *getAttribute* – responsável por criar os atributos do elemento no XML;
- *getValue* – responsável por colocar os valores dos elementos no XML a partir dos atributos da classe.

Caso o objeto principal consultado contenha referências a outros objetos, o objeto em causa é colocado no topo da pilha e o processo é iniciado para a classe em causa. Após ter percorrido toda a árvore de objetos a consultar, é efetuada a serialização do XML e enviado o XML resultado da consulta para o sistema na WEB.

3.1.2 Conversão de XML para Objetos

Para conseguir integrar a informação no sistema central, com base no XML em formato reconhecido pelos objetos de BackOffice, é ativado um processo de abastecimento desses objetos, conforme a figura 3.

Para cada objeto de negócio foi criada uma classe responsável pelo seu abastecimento a partir de um XML (por exemplo: à classe Apolice, corresponde uma classe XML2Apolice).

O processo é baseado num componente chamado de “XML Engine” que é responsável por efetuar o *parsing* do XML recorrendo ao DOM4J, e para cada uma das etiquetas e atributos existentes, fazer a respectiva instanciação e atribuição aos objetos XML2Obj. Por cada etiqueta que encontra no XML, o XML Engine procura identificar se essa etiqueta corresponde a uma classe ou não. Caso corresponda, acrescenta a identificação dessa classe à pilha que guarda a pilha de classes que vai sendo percorrida. Se a etiqueta não corresponder a uma classe, significa que é um atributo da classe que está no topo da pilha, sendo que neste caso o XML Engine irá invocar os métodos: ClasseTopoStack.strTag, ClasseTopoStack.setTag e ClasseTopoStack.endTag.

Quando as validações de negócio (ao efetuar o set dos valores nos objetos) provocam erros, é invocada a classe ErrorContainer, sendo guardado o XPath da etiqueta ou atributo em causa, e a respectiva mensagem. Se no final do processo existirem erros, é executado o processXMLLog, que efetua o *merge* do XML original com as mensagens de erro guardadas no ErrorContainer.

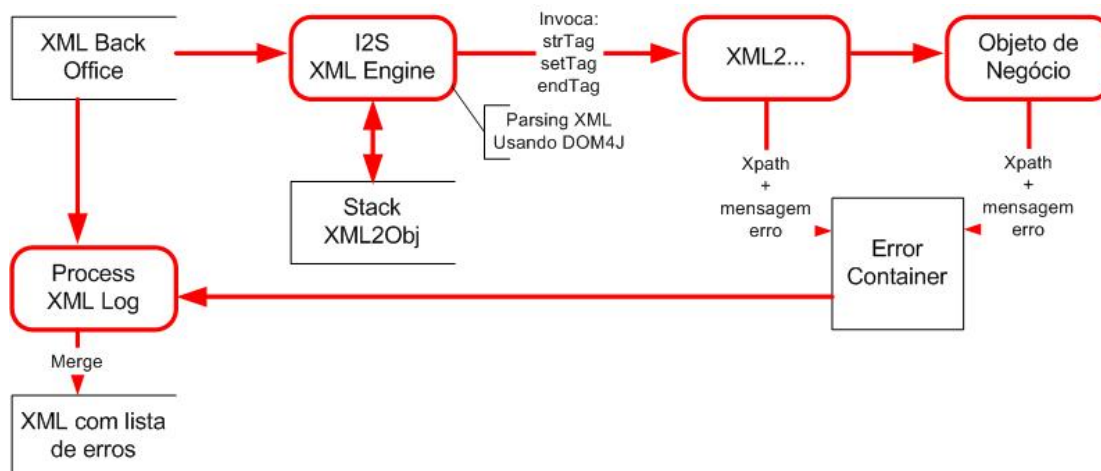


Figura 3 - Conversão de XML para Objetos

Com o documento XML criado a partir dos erros é necessário armazená-lo e possuir uma maneira de realizar consultas, para que os utilizadores possam facilmente identificar os erros que foram gerados, pois atualmente a pesquisa nesses documentos torna-se difícil e complexa ao utilizador.

4. SOLUÇÃO ADOTADA

Como o documento XML de logs de erros possui informações que são semi-estruturadas, ou seja, não possuem um esquema rígido, optamos por utilizar uma base de dados XML nativa. Outro fator importante foi a utilização de uma base de dados XML Nativa que não obrigasse a nenhum tipo de mapeamento para outra base de dados, pois os documentos são armazenados no mesmo formato XML.

A aplicação que desenvolvemos é composta por dois componentes: as aplicações *Java Server Pages*⁵ (JSP), que fornecem a interface; e um conjunto de *servlets*⁶ que respondem aos pedidos invocados pelas páginas JSP; conforme a figura 4

⁵ <http://java.sun.com/products/jsp/>

⁶ <http://java.sun.com/products/servlet/>

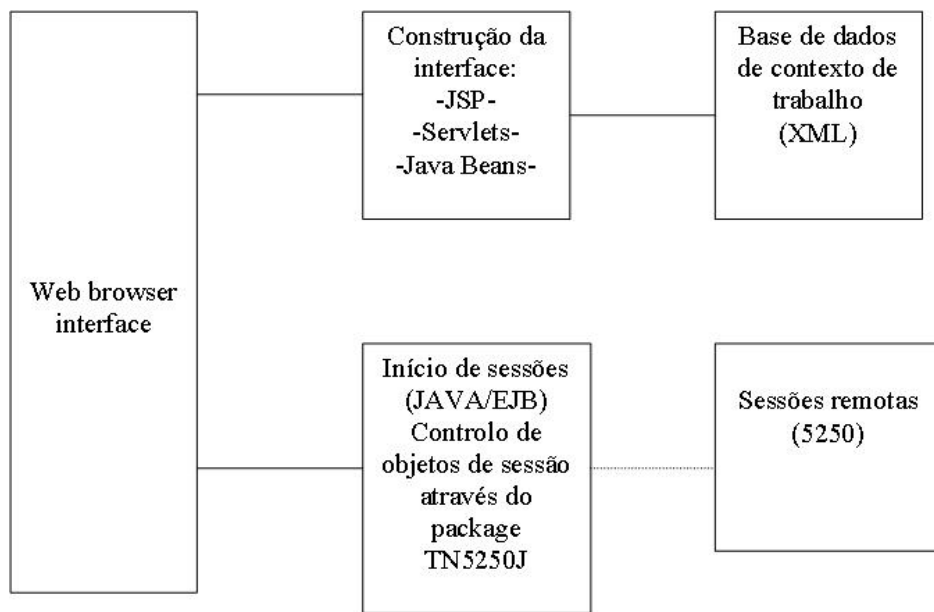


Figura 4 - Esquema do Sistema

4.1 Tecnologias Utilizadas

Xindice

A escolha da base de dados Xindice, foi motivada por:

- Compatibilidade com plataforma Windows e Unix;
- Utilização de API's Java ;
- Base de dados *open source*;
- Consulta aos documentos através da linguagem Xpath;
- Utilização da linguagem Xupdate – Evitando carregar todo o documento para alterar um único registro.
- Utilização de Índices – O Xindice fornece uma boa plataforma para suporte de léxico sob o formato XML, que pode ser indexado para responder a diferentes requisitos de processamento.

Servlets e Java Server Pages

Um *servlet* é uma classe escrita na linguagem Java cujos objetos têm a finalidade de gerar documentos codificados em HTML(*HyperText Markup Language*) . Esta característica dos *servlets* implica que para a construção das páginas seja necessário recorrer à programação em Java.

Em JSP, já não há a necessidade da programação em Java para se construir uma página. Uma página escrita em JSP é uma página escrita em HTML que contém pequenos fragmentos de código Java e/ou tags especiais (definidos na especificação JSP).

A tecnologia JSP permite misturar HTML normal e estático com conteúdo gerado dinamicamente dos *servlets*. O HTML convencional é escrito normalmente, utilizando ferramentas de construção de páginas, para as partes dinâmicas o código é envolvido por tags especiais, que normalmente começam com `<%` e terminam com `%>`.

Servidor Tomcat

No entanto, para se conseguir executar uma página JSP, é necessário existir um servidor que compile o JSP num *servlet*, esse servidor é o Tomcat7. O servidor Tomcat tem a capacidade de converter/compilar automaticamente uma página JSP num *servlet*.

4.2 Resultados obtidos

Nos testes inicialmente realizados, obtivemos bons resultados. Conseguimos facilmente manipular e armazenar os documentos XML e permitir que numa interface WEB se fizessem consultas sobre esses documentos.

Por possuir uma estrutura descendente hierárquica, o Xindice mostra-se particularmente eficiente quando é possível incluir na pesquisa o caminho completo do documento, ou seja, quando o Xpath engloba o caminho completo desde a raiz, em vez de se efectuar pesquisas do tipo `//Elemento`.

No entanto, ainda não foram realizados testes com grandes volumes de informação, sendo necessário continua-los; até porque sabemos de algumas limitações do Xindice ao trabalhar com documentos com mais de 5MB.

⁷ <http://jakarta.apache.org/tomcat/>

CONCLUSÃO

O nosso trabalho teve com o objetivo mostrar uma solução para armazenar e processar documentos XML. Apresentamos as bases de dados XML Nativas e como podem facilitar o processamento de documentos XML em alguns casos específicos.

Uma base de dados XML Nativa só deve ser utilizada quando as informações que tivermos forem documentos que não tenham nenhum esquema rígido; migrar tabelas relacionais para documentos XML e gerir esses documentos em base de dados XML Nativas não é uma boa solução, mas utilizar essa base de dados para armazenar os documentos XML para serem transferidos para base de dados relacionais já se torna uma boa alternativa, assim como trabalhar com documentos XML sem estrutura rígida, como seria o caso dos logs de erro.

Propomos uma solução para consulta de logs de erros gerados por um processo de tratamento de pedidos pendentes (processamento de Protocolo em *batch*), em que a informação dos logs é gerada em formato XML. O tratamento de pedidos, consiste em integrar no AS400 os pedidos de alterações de contratos de seguro oriundos de um interface WEB.

Como trabalho futuro teremos: a continuação da investigação sobre o armazenamento de documentos XML no âmbito da tese de mestrado, como por exemplo o Projeto da IBM, chamado Xperanto [Shanmugasundaram01 et al.] [Shanmugasundaram02 et al] e que atualmente foi renomeado para *IBM XML for Tables* [IBM01], e foi disponibilizada uma versão para *download*. Esse sistema tem como objetivo gerar documentos XML a partir de bases de dados relacionais, com a utilização de consultas numa linguagem XML (que atualmente é a linguagem Xquery [W3C03]).

REFERÊNCIAS

[DBXML] XML: DB Initiative for XML Databases. <http://www.xmldb.org/>.

[Florescu99 et al.] Daniela Florescu and Donald Kossmann. Storing and querying xml data using an rdmb. In IEEE Data Engineering Bulletin, pages 27–34. IEEE Computer Society, 1999.

[**IBM01**] IBM: XML for Tables, <http://www.alphaworks.ibm.com/aw.nsf/techs/xtable>.

[**MartinL**] Lars Martin, *Xupdate – XML Update Language*.
<http://www.xmldb.org/xupdate/xupdate-req.html>

[**MendesG**] Giovana Mendes: Exploração de Soluções de Arquivo para documentos XML anotados na linguagem XML e o seu processamento. Master's thesis, Universidade Do Minho, em andamento.

[**Shanmugasundaram01 et al.**] Jayavel Shanmugasundaram, Jerry Kiernan, Eugene Shekita, Catalina Fan, and John, Funderburk, *Querying xml views of relational data*. In, editor, Proceedings of the 27th VLDB Conference, 2001.

[**Shanmugasundaram02 et al.**] Jayavel Shanmugasundaram, Eugene Shekita, Rimón Barr, Michael Carey, Bruce Lindsay, Hamid Pirahesh, and Berthold Reinwald. Efficiently, *publishing relational data as XML documents*. VLDB Journal: Very Large Data Bases.

[**SousaA02**] Artur Afonso de Sousa. Bases de Dados Web e Xml. FCA, 2002.

[**VieraH**] Humberto José Vieira Junior. Xverter: Armazenamento e consulta de dados xml em sgbd's. Master's thesis, Universidade Federal do Rio de Janeiro, 2002.

[**W3C99**] The World Wide Web Consortium W3C, Novembro 1999, *XML Path Language (XPath) Version 1.0*. <http://www.w3.org/TR/xpath>.

[**W3C02**] The World Wide Web Consortium W3C, Maio 2002, *XML Schema*.
<http://www.w3.org/TR/xmlschema-1/>

[**W3C03**] The World Wide Web Consortium W3C, Maio 2003, *XQuery 1.0: An XML Query Language*

[**Xindice**] <http://xml.apache.org/xindice/>

[**Runapongsa02**] Runapongsa, K., PATEL, J., 2002, "Storing and Querying XML Data in Object-Relational DBMSs*". In: International Conference on Extending Database Technology (EDBT'2002) Workshops XMLDM, pp. 266-285, Prague, Czech Republic.

Geração automática de interfaces Web para Sistemas de Informação Metamorphosis

José C. Ramalho¹, Giovanni R. Librelotto^{1*}, Pedro R. Henriques¹

¹Departamento de Informática – Universidade do Minho
Campus de Gualtar – 4710-057 – Braga – Portugal

{jcr,grl,prh}@di.uminho.pt

Abstract. *Hoje em dia a presença na Web é quase mandatória para todos os intervenientes na sociedade da informação. Assim, não é estranho que muitas empresas e instituições dediquem uma grande fatia do seu tempo ao desenvolvimento e manutenção de websites. Esta tarefa tem sido uma grande consumidora de tempo e recursos.*

O que hoje é novidade é normalmente tido como garantido no dia seguinte pelos utilizadores. Os utilizadores nunca estão satisfeitos querem sempre mais: informação actualizada, acessos personalizados, ...

Para responder a estes requisitos os websites têm de ser dinâmicos e têm de ser capazes de reconfigurar automaticamente a sua estrutura, o seu conteúdo e o seu aspecto visual. Este cenário tem favorecido a criação de ferramentas para geração automática e manutenção de websites.

Neste artigo, não propomos mais uma ferramenta desta espécie mas sim uma nova abordagem ao problema.

Nesta nova abordagem o sistema é dividido em duas grandes camadas. Uma camada física que corresponde ao sistema de informação que se quer expôr na Web, que normalmente é constituído por bases de dados, documentos XML, partes do sistema de ficheiros e toda a panóplia de ficheiros nos mais variados formatos que se possa imaginar, e que foi denominada por camada de recursos. Uma segunda camada de metainformação, designada por ontologia, que fornece uma panorâmica estruturada sobre os recursos de informação.

O ambiente ainda em desenvolvimento e baptizado de Metamorphosis, é composto por vários componentes. Neste artigo iremos abordar o componente TMBuilder que extrai automaticamente a ontologia dos recursos de informação, e outro componente DINavigator que a partir da ontologia gera automaticamente o website. Todo o ambiente é baseado na tecnologia XML, desta maneira garante-se a portabilidade e a independência relativamente a plataformas de hardware e software.

1. Introdução

Todos os dias, milhares de recursos de informação são colocados na Web. Este facto faz com que a Web cresça exponencialmente a cada dia que passa tornando as tarefas de pesquisa de informação cada vez mais complexas e difíceis. Na tentativa de resolver este problema surgiram várias iniciativas e uma nova área de investigação tem emergido: *Semantic Web*.

*Bolsista CNPq - Brasil

Quando nos referimos a *Semantic Web* estamos a referir-nos a uma rede semântica de conceitos. Cada conceito está relacionado com um conjunto específico de recursos de informação e pode ser relacionado com outros conceitos. Esta rede de conceitos pode então ser utilizada para navegar por entre recursos Web ou simplesmente num sistema de informação que tenha uma interface Web.

A ideia principal subjacente ao *Metamorphosis* é a integração da especificação destas redes semânticas ou ontologias com a navegação por entre os recursos relacionados e o seu armazenamento.

O projecto de desenvolvimento do *Metamorphosis* teve como requisito inicial a utilização da tecnologia XML para suportar todos os componentes do sistema com vista a manter a portabilidade e independência de plataformas de todos os componentes.

A ideia de utilizar XML para a especificação de ontologias não é nova e várias linguagens de anotação definidas em XML foram desenvolvidas. Das várias iniciativas houve uma que ganhou o estatuto de norma ISO: Topic Map ISO 13250. No entanto, neste cenário, há linguagens de anotação que têm de ser consideradas, como: RDF [LS99], RDFS [BG00], DAML [DAR01], OIL [OIL02] ou OWL [BvHH⁺02]. A mais genérica e abstracta é a *Topic Map*, ou a sua versão mais recente em XML, *XTM*. Relativamente à dimensão da comunidade de utilizadores rivaliza com o *RDF*. Os utilizadores duma e de outra têm normalmente os mesmos objectivos mas diferem na abordagem que fazem à resolução do problema que pretendem resolver: o *XTM* induz uma abordagem *top-down* enquanto que o *RDF* é utilizado em abordagens *bottom-up*.

Quando se está a desenvolver uma interface Web para um sistema de informação usa-se, normalmente, uma perspectiva *top-down*. Para além deste aspecto, pretendia-se uma linguagem o mais abstracta possível para permitir facilmente modificações e acrescentos. Com este requisitos, o *XTM* surge como a melhor opção para especificação de uma ontologia a partir da qual é gerado um website.

Há muitas ferramentas que usam ontologias para a geração automática de websites ou para navegar num conjunto de recursos. A maioria destas ferramentas estão implementadas numa linguagem tradicional de programação como o *Java*, o *Perl* ou o *Python*. No *Metamorphosis* obtivemos os mesmos resultados recorrendo apenas à tecnologia XML. O sistema é apenas composto por documentos XML, uns com informação, outros com transformações XSL [RH02]. Além disso, o sistema foi enriquecido com novos componentes e terá outros que ainda estão em fase de desenvolvimento que vêm acrescentar funcionalidades muito interessantes a uma ferramenta deste tipo.

Na próxima secção dá-se uma breve descrição dos conceitos técnicos necessários para entender o resto do artigo: Topic Maps, XML, e transformações de documentos XML (XSL). A secção 4. introduz a arquitectura do sistema desenvolvido. A seguir, a secção 5., discute a implementação do componente de navegação encarregue da geração automática de websites. Finalmente, a secção 6. apresenta um dos casos de estudo que foram utilizados para o desenvolvimento do sistema. O artigo termina com algumas conclusões e com a descrição do trabalho em curso.

2. Semantic Web e Ontologias

A *World Wide Web* (Web) é sem dúvida um dos maiores sucessos na história dos empreendimentos humanos, contando com utilizadores de todo o mundo, manipulando e acedendo a uma enorme quantidade de informação.

Apesar da complexidade crescente da Web, isto não é refletido no estado actual das tec-

nologias utilizadas para a sua manipulação. A maior parte das tarefas de interpretação, acesso, extração e manutenção da informação ainda é deixada a cargo dos utilizadores.

Os motores de busca são ineficientes quando se trata de fazer inferências complexas e relacionar assuntos aparentemente disjuntos. A simples anotação de páginas HTML por intermédio das tags <META>, ou mesmo o emprego de padrões de metadados, não é suficiente para incluir a semântica desejada, que possibilitaria a execução de tarefas mais sofisticadas e mais úteis do que as actualmente existentes.

Na abordagem de Tim Berners-Lee [BLHL01], esse tipo de construção leva a limitações e a um tratamento trivial por parte dos atuais browsers, do conteúdo das páginas Web – limita-se a um cabeçalho, links para outras páginas; mas, em geral, as ferramentas não possuem uma forma confiável de processar o conteúdo semântico das informações contidas em uma página.

Com base nessas premissas, surgiu a ideia da *Semantic Web*, na qual o conhecimento da Web é armazenado por meio da utilização de (meta) dados processáveis por ferramentas. Pretende-se que a *Semantic Web* não seja separada da Web, mas uma extensão desta tecnologia. Basicamente, os mecanismos a serem desenvolvidos para o estabelecimento da *Semantic Web* compreendem duas vertentes: a disponibilização de uma coleção de dados estruturados e regras de inferência associadas a essa coleção; e a criação de ferramentas capazes de percorrer a Web realizando tarefas complexas com base nessas estruturas de conhecimento.

Esse conjunto coerente de coleções estruturadas de informação forma uma ontologia. Uma ontologia é uma teoria lógica para descrever o significado pretendido de um vocabulário formal, isto é, seu compromisso com uma conceitualização particular do mundo. Estas incluem estruturas que permitem manipular termos de uma forma muito eficiente e útil para o utilizador e mecanismos de validação para comunicação entre programas. A importância de seu uso é devida à capacidade de representar hierarquias de classes de objetos (taxonomias) e seus relacionamentos.

Dentro de uma mesma área podem ser encontradas diferentes definições e classificações de ontologia. Na área de Inteligência Artificial, Guarino define a ontologia como uma caracterização axiomática do significado do vocabulário lógico [Gua97] e para Sowa, a ontologia define os tipos de coisas que existem no domínio de uma aplicação [Sow00].

Na área de Sistemas de Informação, na qual se encaixa este trabalho, ontologia é definida como um conjunto de conceitos e termos que podem ser usados para descrever alguma área do conhecimento ou construir uma representação para o conhecimento [ST99]. Segundo Chandrasekaran [Cha99], ontologias são teorias de conteúdo sobre os tipos de objetos, propriedades de objetos e relacionamentos entre objetos que são possíveis num domínio de conhecimento específico.

O desenvolvimento de ontologias irá alimentar o mecanismo de construção da parte semântica da *Semantic Web*. O modelo em camadas proposto por Berners-Lee¹ tem sido aceite principalmente como representação para a arquitetura da *Semantic Web*. O desenvolvimento de tais mecanismos depende, obrigatoriamente, de linguagens que expressem a informação de maneira a ser entendida por máquinas. O desafio é proporcionar uma linguagem que manipule igualmente, de maneira eficiente, dados e regras para deduções sobre esses dados e que permita que regras existentes em qualquer sistema de representação de conhecimento possam ser exportadas para a Web.

A fim de prover o primeiro mecanismo necessário à *Semantic Web*, a anotação da informação em XML (*eXtensible Markup Language*)² vem sendo reconhecida como relevante. XML permite representar dados em formato semi-estruturado, o que ocorre com frequência no mundo real. Entretanto, XML por si mesmo, não permite acrescentar significado a tais estruturas. Ao usar XML

¹Disponível em <http://www.w3.org/2000/Talks/1206-xml2k-tbl/slide10-0.html>

²eXtensible Markup Language - Language Syntax Specification, <http://www.w3.org/TR/REC-xml>

como sintaxe para transmissão de dados semi-estruturados, a descrição do significado deve ficar a cargo de alguma linguagem de especificação semântica.

Na próxima secção, aborda-se a especificação da semântica.

3. Topic Maps

XML Topic Maps (XTM) [PM01] é um formalismo para representar conhecimento acerca da estrutura de um conjunto de recursos de informação e para o organizar em *tópicos*. Esses tópicos têm ocorrências e associações que representam e definem relacionamentos entre os tópicos. A informação sobre os tópicos pode ser inferida ao examinar as associações e ocorrências ligadas ao tópico. Uma coleção desses tópicos e associações é chamada *topic maps*. Também pode ser visto como um paradigma que permite organizar, manter e navegar pela informação, permitindo transformá-la em conhecimento.

Topic maps pode ser definido como uma descrição de um ponto de vista sobre uma coleção de recursos, organizado formalmente por tópicos, e pela ligação de partes relevantes do conjunto de informação aos tópicos apropriados [Pep00].

Um mapa de tópicos expressa a opinião de alguém sobre o que os tópicos são, e quais as partes do conjunto de informação que são relevantes para cada tópico. *Charles Goldfarb* [GP01] (o pai das linguagens de anotação) geralmente compara topic maps com GPS (*Global Positioning System*) aplicado ao universo da informação. Falar sobre topic maps é falar sobre estrutura de conhecimento.

Permitindo criar um mapa virtual da informação, os recursos de informação mantêm-se em sua forma original e não são modificados. Então, o mesmo recurso de informação pode ser usado de diferentes maneiras, por diferentes mapas de tópicos. Como é possível e fácil modificar um mapa, a reutilização da informação é conquistada.

3.1. As características do modelo XTM

Tópicos são o ponto principal de topic maps [PHE03]. Em um sentido mais genérico, podem ser qualquer coisa: uma pessoa, uma entidade, um conceito. Eles constituem a base para a criação de topic maps. Podem ser vistos como um link-múltiplo, o qual aponta para todas as suas ocorrências [BN99].

Cada tópico tem um tipo de tópico (*topic type*), ou talvez múltiplos tipos. Cada tipo de tópico pode ser visto como uma típica relação classe-instância. Os tipos representam as classes onde os tópicos estão agrupadas, i.e., a categoria de cada instância tópico. Segundo o modelo de topic maps, os tipos de tópicos também são tópicos.

Um tópico pode ter um ou mais nomes. A opção de especificar mais de um nome ao tópico pode ser utilizada em diferentes contextos (*scopes*), como idiomas, estilos, domínios, área geográfica, período histórico, etc.

Um tópico pode ter uma ou mais ocorrências. Um ou mais recursos de informações endereçáveis de um tópico constituem o conjunto de ocorrências de tópicos (*topic occurrences*). As ocorrências de tópicos podem ser, por exemplo, um artigo, uma imagem ou vídeo. Uma ocorrência representa a informação que é especificada como relevante para um certo tópico [Pep00]. As ocorrências podem ser endereçáveis através de uma URI (*Universal Resource Identifier*).

Ocorrências e tópicos existem em duas camadas diferentes, mas elas são "conectadas" entre si. As ocorrências estabelecem rotas dos os tópicos para os recursos de informação, possi-

bilitando também prover uma razão de o porque que a rota existe. Neste ponto, a separação em duas camadas é percebida: tópicos e suas ocorrências; uma das grandes vantagens de XTM.

Entre todas as ocorrências de um tópico, uma distinção pode ser feita através de subgrupos. Cada subgrupo é definido por um papel de atuação (*role*) em comum. Os papéis de atuação em ocorrências (*occurrence role*) podem ser utilizados para distinguir gráficos de texto, menções de definições, etc. Os papéis de atuação em ocorrências são definidos pelos usuários, sendo assim podem variar em cada topic maps [BN99]. Para fazer a real distinção entre diferentes tipos de ocorrências, topic maps também utiliza o conceito de tipo de papel de atuação em ocorrência (*occurrence role type*) [Pep00].

As associações (*associations*) são responsáveis pelos relacionamentos entre os tópicos. Elas são ligações independentes da fonte dos documentos onde as ocorrências de tópicos são encontradas; elas representam a base do conhecimento, a qual contém a essência da informação que alguém criou e atualmente representa seu valor essencial. Um ilimitado número de tópicos podem ser relacionados por uma associação.

O poder de topic maps aumenta com a criação de associações porque, deste modo, é possível agrupar um conjunto de tópicos que de algum modo são relacionados. Isso é de grande importância ao prover interfaces intuitivas e amigáveis para a navegação de grandes quantidades de informação.

Assim como os tipos de tópicos agrupam vários tópicos e tipos papéis de atuação também suportam várias ocorrências, as associações entre tópicos devem ser agrupadas de acordo com seu tipo de associação (*association type*). Cada tópico que participa em uma associação tem um papel (*association role*) que expressa a sua atuação nessa associação. Os papéis de atuação em associação também são vistos como tópicos no modelo XTM.

3.2. Como definir um Topic Map

Antes de iniciar, é necessário definir exatamente o que é que será representado no Topic Map. E para isso, existem duas partes: a definição do escopo do Topic Map, isto é, decidindo a extensão do domínio que ele deve cobrir; e o projeto da ontologia básica. Em Topic Maps, uma ontologia é uma precisa descrição dos tipos de coisas as quais são encontradas no domínio coberto pelo topic map. Em outras palavras, o conjunto de tópicos que é usado para definir classes de tópicos, associações, papéis de atuação e ocorrências. Este termo também é usado em outras áreas, tal como na filosofia, onde tem outro significado.

Para ilustrar todas as idéias introduzidas e para descrever o processo de construção de um TM, é apresentado o caso de estudo onde o assunto é uma workshop. Esta workshop é o XATA - XML: Aplicações e Tecnologias Associadas realizado na Universidade do Minho, em 2003. Considerando este exemplo, tem-se quatro tipos de tópicos: Instituições, Artigos, Inscritos e Sessões.

Nos exemplos que se segue, é assumido que o participante de nome *Pedro Rangel Henriques* está inscrito no XATA como pertencente ao Departamento de Informática da Universidade do Minho, e é autor do artigo *Especificação XML de Aplicações para WWW*, que foi apresentado na sessão *Dialectos XML*. A ontologia básica consiste dos tipos de tópicos descritos no parágrafo anterior, além dos papéis de associações *pertence/contém*, *está na sessão/contém o artigo* e *é autor de/foi escrito por*, assim como os tipos de associações *Artigo e Sessão*, *Inscrito e Instituição* e *Inscrito e Artigo*.

Primeiramente, é definido os tópicos (os tipos de tópicos e suas instâncias), especificando seus identificadores e seus nomes. Abaixo, está um exemplo incompleto:

```
<?xml version="1.0" encoding="UTF-8"?>
<topicMap xmlns="http://www.topicmaps.org/xtm/1.0"
```

```

        xmlns:xlink="http://www.w3.org/1999/xlink">
<topic id="Inscrito">
  <baseName>
    <baseNameString>Inscrito</baseNameString>
  </baseName>
</topic>
<topic id="pedrorangelhenriques">
  <instanceOf>
    <topicRef xlink:href="#Inscrito" />
  </instanceOf>
  <baseName>
    <baseNameString>Pedro Rangel Henriques</baseNameString>
  </baseName>
</topic>
</topicMap>

```

Aqui apenas é demonstrado a definição do tópico *Pedro-Rangel-Henriques* e seu tipo *Inscrito*. Os outros tópicos e seus tipos são criados de uma forma similar.

Após os tópicos, é adicionado as ocorrências referentes aos próprios tópicos, usando o elemento *resourceRef* que contém a URL (endereço para o recurso) como o valor do atributo *xlink:href*. Por exemplo, uma ocorrência para o tópico *Pedro Rangel Henriques* pode ser o seu e-mail, como mostrado abaixo:

```

<topic id="pedrorangelhenriques">
  ...
  <occurrence>
    <instanceOf>
      <topicRef xlink:href="#email" />
    </instanceOf>
    <resourceRef xlink:href="prh@di.uminho.pt" />
  </occurrence>
</topic>

```

Note que o uso de *#email* as a ; isso é possível porque *#email* também é um tópico, mais precisamente um tipo de ocorrência.

O "...” do código acima segue a definição do tópico *Pedro Rangel Henriques* como apresentado no fragmento anterior; apenas não é repetido para tornar o exemplo mais simples.

Num terceiro passo, é definido as associações entre os tópicos, expressando seus tipos e seus membros (um tópico com um papel de atuação explícito). No exemplo abaixo, é definida associação Autor e Artigo entre *Especificação XML de Aplicações para WWW* e *Pedro Rangel Henriques*. O primeiro desempenha o papel de *é escrito por* e o segundo, o papel de *é autor de*.

```

<association>
  <instanceOf>
    <topicRef xlink:href="#Inscrito-e-Artigo" />
  </instanceOf>
  <member>
    <roleSpec>
      <topicRef xlink:href="#ID-escrito_por" />
    </roleSpec>
    <topicRef xlink:href="#especificacaoxmldeaplicacoesparawww" />
  </member>
  <member>
    <roleSpec>
      <topicRef xlink:href="#ID-autor_de" />
    </roleSpec>
    <topicRef xlink:href="#pedrorangelhenriques" />
  </member>
</association>

```

As referências *ID-escrito_por* e *ID-autor_de* são papéis de atuação em associações e são declaradas como tipos de tópicos, isto é, com um identificador e um nome, somente. A referência *Inscrito-e-Artigo* é um tipo de associação que define o tipo desta associação. A declaração do tópico é mostrada abaixo:

```

<topic id="Inscrito-e-Artigo">
  <baseName>
    <baseNameString>Inscrito e Artigo</baseNameString>
  </baseName>
  <baseName>
    <scope>
      <topicRef xlink:href="#ID-escrito_por"/>
    </scope>
    <baseNameString>é escrito por</baseNameString>
  </baseName>
  <baseName>
    <scope>
      <topicRef xlink:href="#ID-autor_de"/>
    </scope>
    <baseNameString>é autor de</baseNameString>
  </baseName>
</topic>

```

O TM acima significa que *Pedro Rangel Henriques* é autor de *Especificação XML de Aplicações para WWW* e, de modo inverso, indica que *Especificação XML de Aplicações para WWW* foi escrito por *Pedro Rangel Henriques*.

As razões para o uso de topic maps são, primeiramente, o facto de que existem muitas ferramentas baseadas em topic maps, o que contribui para a distribuição do processo. Dentre essas ferramentas, pode-se citar as apresentadas na tabela 1.

Ferramenta	Descrição
<i>Ontopia Omnigator</i>	Browser conceptual sobre XTM. Necessita do TomCat.
<i>DINavigator</i>	Browser conceptual sobre XTM. Baseado em XSLT.
<i>TMNav</i>	Editor gráfico de XTM.
<i>Mapalizer</i>	Gerador de Topic Maps a partir de um conjunto de documentos de entrada.
<i>TM-Designer</i>	Editor gráfico de ISO Topic Maps.
<i>TM4J</i>	Processador de Topic Maps.

Table 1: Algumas ferramentas baseadas em Topic Maps.

Segundo, topic maps é uma excelente tecnologia e um padrão que une vários níveis de profissionais, de técnicos de marketing até arquitectos de informação. Terceiro, uma grande parte do trabalho dos autores deste artigo é a criação de sites e protótipos de vários projectos e estilos, então, para facilitar o processo, foi construído o *DINavigator* que apresenta o conhecimento extraído das relações através de menus contextuais. Concluindo, topic maps mostra ser a perfeita tecnologia para esta tarefa.

Além do *DINavigator*, uma gama de ferramentas para o processamento de topic maps está disponível. Contudo, nenhuma delas é uma distribuição simples de instalar e usar, pois requerem outra tecnologia, como outro software, biblioteca ou linguagem de programação. É o caso do *Tomcat* que é necessário para o uso do *Omnigator*, assim como o *TM4J* se faz necessário para o uso do *Nexist*, ou a instalação do *Python* para o uso do *SemanText*, assim como um plug-in no *Protégé 2000* para a edição de topic maps. Assim, *DINavigator* provém uma simples solução para o processamento de topic maps, pois apenas tem a dependência de um simples parser XSLT.

O documento XTM é onde todos os metadados sobre a ontologia estão armazenados, e o *DINavigator* usa isto para a estruturação, relações e tudo mais o que for necessário. O documento XTM gerado pelo TM-Builder não é indicado para a edição manual, pois na próxima geração do XTM, todas as alterações feitas manualmente serão sobrescritas pelo processo.

4. Arquitectura do Metamorphosis

A motivação para o desenvolvimento do *Metamorphosis* veio de duas situações que surgiram no contexto de alguns projectos de desenvolvimento de software:

- Muitas vezes, para se testar algumas funcionalidades de um sistema que se está a desenvolver é necessário criar uma interface Web para realizar esses testes. Normalmente, estas interfaces não têm grandes requisitos quanto à aparência, o que se pretende é que sejam desenvolvidas o mais rapidamente possível.
- Quer-se expôr na Web um sistema de informação. Este sistema é composto por bases de dados, documentos XML ou documentos PDF. Quer-se que todos os itens de informação estejam acessíveis via Web e para isso constroem-se os primeiros índices. Estes índices acabam por ser enormes excedendo a capacidade dos browsers da Internet. Podia pensar-se em fraccioná-los alfabeticamente, mas há situações em que isso não é possível nem recomendável. Mas, é sempre possível arranjar um método para fraccionar a informação conceptualmente. É aqui que se começa a discutir a organização dos recursos de informação e é aqui que se introduz o *Metamorphosis*. Não é necessário alterar nenhum dos recursos de informação, apenas é necessário criar uma ontologia para esses recursos que reflita a visão que se quer para eles.

A figura 1 vem reforçar esta ideia e dá uma visão pictórica da arquitectura do *Metamorphosis*.

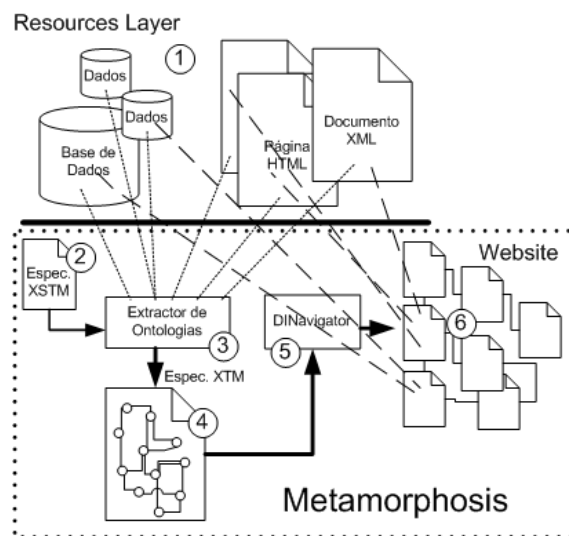


Figure 1: Metamorphosis

Esta arquitectura pode ser descrita da seguinte maneira:

- (1) **camada de recursos** Este componente é composto pelos vários recursos de informação: documentos XML, páginas Web, bases de dados, ... O *Metamorphosis* não interfere com nenhum deles, apenas utiliza parte da informação de cada um para construir a ontologia ou rede semântica.
- (2) **especificação XSTM** É um documento XML que fornece informações precisas sobre onde aceder para extrair as partes da informação necessárias para a construção da ontologia. Esta especificação descreve todos os tópicos a serem extraídos dos recursos de informação e as associações entre estes tópicos; este componente é descrito em detalhe em [LRH03b].
- (3) **Ontology Builder/Extractor** Este componente utiliza a especificação XSTM para ir aos recursos de informação buscar a informação de que necessita para construir a ontologia.

Está implementado como uma folha de estilo XSL e vai tirar partido da organização da informação: por exemplo, quando se está a trabalhar com uma base de dados relacional, procura a informação relativa às relações e vai gerar automaticamente todas as associações entre os tópicos relevantes.

- (4) **especificação XTM** Esta é a especificação do topic map gerada automaticamente pelo Ontology Builder/Extractor.
- (5) **DINavigator** Este é o componente que está a ser discutido com detalhe neste artigo. Também implementado em XSL, toma como entrada um *Topic Map* e produz um website inteiro de acordo com algumas regras.
- (6) **Website** Este é website final através do qual é possível navegar por entre os vários recursos de informação que compõem o sistema de informação original.

Na próxima secção, descreve-se em detalhe o *DINavigator* e a geração do website.

5. DINavigator - o componente de navegação

Até agora, as ontologias especificadas em XTM são um conjunto de registos, onde cada registo representa um conceito, que identifica recursos (informação física) e participa de relações (associações). Então, primeiramente, usa-se essas relações entre os conceitos para navegar na informação e, posteriormente, quando encontrado o recurso de informação desejado, recuperar essa informação.

A ideia principal da navegação conceptual pode ser descrita como: quando se está posicionado sobre um certo conceito, a ferramenta de navegação mostrará as informações associadas a este conceito em particular; se for escolhido algum dos outros conceitos relacionados, a posição muda para o conceito e a visão muda de acordo com a escolha; se for escolhido algum dos recursos de informação, o sistema mostrará o próprio recurso.

Em termos de implementação, foi estabelecido que a navegação se faria através de web browsers. Então, todas as visões dos conceitos seriam apresentadas em páginas HTML. Assim, a implementação desse componente de navegação é reduzida a transformações XML. Contudo, há duas possibilidades: transformação a pedido em tempo de execução (website dinâmico) e transformação em batch (website estático).

Este navegador é definido como *DINavigator*, e é apresentado na figura 2, onde é encontrada a visualização do topic map referente ao *DI*, (criado pelo TM-Builder) no *DINavigator*. Este navegador fornece o total acesso à ontologia extraída, permitindo a navegação através dos conceitos definidos na especificação em XSTM.

A visualização de um documento XTM no *DINavigator* apresenta no seu lado esquerdo, a árvore obtida a partir da ontologia, na forma de um menu hierárquico. No corpo do navegador, pode-se navegar a partir da própria ontologia, por um índice alfabético de todos os tópicos ou pela sua hierarquia de classes, visualizada graficamente.

5.1. Transformações em tempo de execução

As transformações em tempo de execução foram implementadas com transformações XML com auxílio de uma CGI. A página inicial é uma chamada a uma CGI parametrizada com o principal conceito da ontologia (que vai servir de ponto de entrada para a navegação). A CGI aplica a transformação na ontologia e produz uma visão HTML. Nessa visão, todos os links gerados são chamadas à mesma CGI mas com diferentes parâmetros, para novas posições na ontologia.

Esta é a melhor solução porque somente necessita de dois ficheiros: a ontologia (especificada em XTM) e a CGI. Contudo, as transformações em tempo de execução consomem muito tempo. Por isto, esta solução tem sido utilizada apenas em pequenos protótipos.



Figure 2: Visualização do Topic Map do XATA no *DINavigator*

5.2. Transformações em batch

Essa solução corresponde ao atual componente que está sendo usado no *Metamorphosis*. Foi criada uma stylesheet XSL que transforma a ontologia num website. Esse website corresponde a todas as possíveis visões da ontologia.

Assim, o *DINavigator* é visto como um gerador de websites, baseado em XSL e usando topic maps para esse propósito. Esta é uma maneira simples de criar de websites inteiros, incluindo o design, o conteúdo e os links aos tópicos. Em outras palavras, o *DINavigator* é simples e poderoso.

O *DINavigator* foi criado como uma consequência directa da criação de um *framework* em XSLT para a linguagem XSTM [LRH03b]. Quando essa linguagem foi desenvolvida, houve a necessidade de criar uma ferramenta (ou protótipo) para esse trabalho, para validar os documentos XTM gerados a partir do TM-Builder. O *DINavigator* foi criado por três razões:

- Para ser um protótipo eficiente neste trabalho, e diminuindo o tempo de criação de um navegador para um XTM;
- Para criar uma ferramenta que usa topic maps, e que é facilmente distribuída, instalada e usada por todos;
- Para unir as linguagens técnicas e métodos de projetistas, programadores e arquitetos de informação.

O *DINavigator* oferece um *framework* para permitir rápidas modificações em todo o website. O website gerado pelo *DINavigator* é transformado através de parsers XSLT, e o resultado é cuidadosamente verificado até à produção completa das páginas HTML. Isto simplesmente significa que é possível a rápida criação de protótipos e a expansão usando essa mesma ferramenta para se obter um ambiente de produção. Isso diminui radicalmente o custo e o tempo de desenvolvimento.

O Topic map é usado no *DINavigator* para todas as estruturas, relações, preparação de conteúdo, recursos e ocorrências, ou seja, o topic map expressa a ontologia. Isso significa que qualquer outro topic map, independente de ter sido obtido através do *TM-Builder*, pode ser usado no *DINavigator* para a obtenção de um completo mapa de metadados.

6. Caso de estudo: XATA 2003

A fim de demonstrar o uso do TM-Builder para a extração de ontologias a partir de uma fonte XML, esta seção apresenta o caso de estudo da workshop *XML, Aplicações e Tecnologias As-*

sociadas (XATA³), que ocorreu na Universidade do Minho, em Braga - Portugal, nos dias 13 e 14 de Fevereiro de 2003, visando reunir a comunidade XML de língua portuguesa. Participaram pesquisadores e utilizadores de XML, tanto oriundos de universidades quanto de empresas, permitindo assim um compartilhamento de informação entre o mundo acadêmico e o mundo profissional.

Nesta workshop, vários artigos foram submetidos para avaliação; e os aprovados foram devidamente apresentados durante a conferência. As apresentações dos artigos foram divididas em sessões, cada qual com um tema associado, como Tecnologia e Web Services, XML e Base de Dados, entre outras. Após cada exposição, uma mesa redonda era formada para debater o assunto do artigo, consoante as dúvidas que o mesmo havia criado.

Como não poderia deixar de ser, o evento foi todo baseado em XML, desde sua divulgação quanto sua produção. Portanto, todas as informações referentes ao XATA estão armazenadas em documentos XML. A parte interessante, a este caso de estudo, do XML-Schema do documento que contém as informações essenciais sobre o evento é apresentado na figura 3. Obviamente este XML-Schema é mais completo, porém, para nosso caso de estudo, o importante está ressaltado nessa figura.

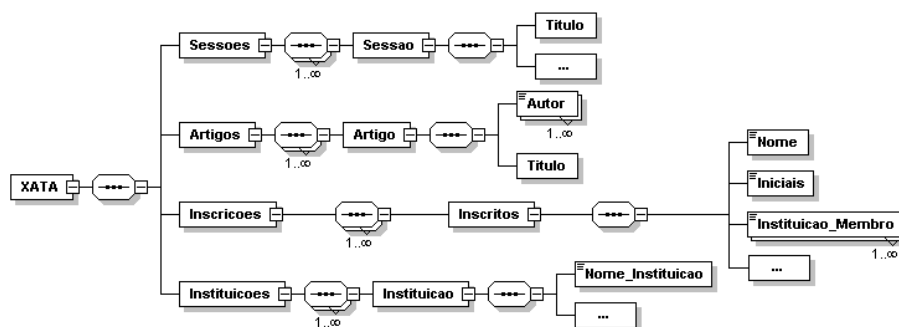


Figure 3: XML-Schema do XATA.

Como a linguagem XSTM depende apenas da estrutura do documento, e não da instância, a partir deste XML-Schema é possível definir a especificação da ontologia do XATA. Portanto, são cinco etapas que devem ser seguidas: definição dos tipos de tópicos, dos próprios tópicos, dos papéis de atuação em ocorrências, dos tipos de associação e, por fim, das próprias associações. This section presents a case study about the workshop *XML, Aplicações e Tecnologias Associadas* (XATA).

6.1. A especificação XSTM para o XATA

O elemento raiz do XML-Schema de XSTM é *xstm*, o qual possui cinco sub-elementos, cada um referente a uma parte da ontologia expressada por XTM. Os seus sub-elementos são: *topicType*, *topic*, *occurrenceRole*, *assocType* e *assoc*.

Inicialmente, são definidos os tipos de tópicos. Nesta ontologia, os tópicos são agrupados em Instituição, Autor, Assistente, Sessão e Artigo. Em XSTM, os tipos de tópicos são declarados pelo elemento *topicType*, contendo um identificador (*id*) – para ser referenciado em outros momentos na especificação – e um nome (*name*) – para a visualização do XTM em um navegador. Como exemplo, é mostrado abaixo a declaração do tipo de tópico *Artigo*.

```
<xstm>
  <topicTypes>
```

³<http://www.di.uminho.pt/~jcr/XML/conferencias/xata2003/>

```

    <topicType>
      <id>ID-Artigo</id>
      <instanceOf>XATA</instanceOf>
      <name>Artigo</name>
    </topicType>
    <topicType>...</topicType>
  </topicTypes>
  ...
</xstm>

```

Enquanto que os tipos de tópicos são conceitos abstratos definidos pela ontologia, tópicos são elementos reais nos documentos XML tomados como entrada. Para defini-los, é usado o elemento *topic* que possui dois sub-elementos: o caminho XPath referente ao próprio elemento (*xpath*) e o seu tipo (*type*). Abaixo, é encontrado a declaração XSTM para a definição dos tópicos referentes ao tipo de tópico *Artigo*.

```

<topics>
  <topic>
    <xpath id="@numero" name="Titulo">//Artigo</xpath>
    <type>Artigo</type>
  </topic>
  <topic>...</topic>
  ...
</topics>

```

Até o momento, todos os tópicos, e seus respectivos tipos, estão declarados. Mas em XTM, tópicos sem qualquer associação relacionada aos mesmos não possuem funcionalidade. Então, é necessário definir as associações entre os tópicos. Várias associações podem ser inferidas a partir do XATA, por isso, vamos tomar como exemplo a associação entre os tipos de tópicos *Artigo* e *Autor*.

Uma vez definidos os tópicos e seus tipos, o próximo passo é a definição dos tipos de associação, que também é um conceito abstrato na ontologia. Ele define o papel de atuação de cada um dos membros das associações. É declarado com o elemento *assocType* que possui um identificador (*id*), um nome (*name*) e os membros deste tipo de associação (*memberAssoc*). Cada membro é definido por um contexto *scope* – o identificador do papel de atuação – e a sua respectiva descrição (*description*). Sendo assim, é visualizado abaixo a especificação do tipo de associação entre *Autor* e *Artigo*.

```

<assocTypes>
  <assocType>
    <id>ID-autor_artigo</id>
    <name>Autor e Artigo</name>
    <memberAssoc>
      <scope>ID-escrito_por</scope>
      <description>é escrito por</description>
    </memberAssoc>
    <memberAssoc>
      <scope>ID-autor_de</scope>
      <description>é autor</description>
    </memberAssoc>
  </assocType>
  <assocType>...</assocType>
</assocTypes>

```

Para finalizar a especificação em XSTM, o elemento *assoc* permite a especificação de todas as associações que envolvem dois ou mais tópicos; elas são encontradas e extraídas a partir do documento XML fonte.

Neste âmbito, quando refere-se a relacionamentos entre nodos da árvore XML (elementos e atributos), não está se referindo ao modelo entidades-relacionamento. Portanto, os nomes 1-para-1, 1-para-N e M-para-N não tem exatamente o mesmo significado usado na perspectiva tradicional.

Neste contexto, há quatro tipos de relacionamentos entre elementos (ou atributos) que são descritos por três elementos filhos de *assoc*:

- o elemento *one2one* descreve as associações entre elementos e seus atributos, com seu atributo *type* com o valor *attribute*;
- o elemento *one2one* também descreve as associações entre elementos distintos, com seu atributo *type* com o valor *subelement*;
- o elemento *one2N* define as associações *um para muitos*;
- o elemento *M2N* define as associações *muitos para muitos*;
- o elemento *all2all* define as associações que são definidas por uma tabela intermediária.

A estrutura dos sub-elementos de *assoc* são muito similares. Cada um dos três sub-elementos acima descritos possui o seu tipo (*type*) – o identificador do tipo de associação correspondente – e os membros que fazem parte desta associação (*members*). Os membros possuem dois elementos filhos: *topicAssoc* que identifica o tipo de tópico pertencente a esta associação e *role*, que demonstra o papel de atuação do tópico na atual associação.

O elemento *one2one* expressa relacionamentos que podem ser obtidos a partir de algum elo de ligação entre os tópicos encontrados no documento XML. Como por exemplo, no caso específico da associação entre *Autor* e *Artigo*, os autores de cada artigos podem ser identificados devido ao conteúdo do caminho XPath *//Artigo/Autor*, o qual é uma referência às iniciais dos autores encontradas em *//Inscritos/Iniciais*. Assim, a associação entre os tipos de tópicos *Autor* e *Artigo*, referente ao XATA, foi especificada da maneira abaixo demonstrada:

```
<assocs>
  <one2one type="subelement">
    <type>autor_artigo</type>
    <members11>
      <element>
        <topicAssoc ref="Autor">Artigo</topicAssoc>
        <role>eh_escrito_por</role>
      </element>
      <elementRef>
        <topicAssoc ref="Iniciais">Inscritos</topicAssoc>
        <role>eh_autor</role>
      </elementRef>
    </members11>
  </one2one>
</assocs>
...
</xstm>
```

7. Conclusão e Trabalho Futuro

O *Metamorphosis* tem sido utilizado em vários projectos de pequena e média dimensão.

Até ao momento, provou ser uma boa ferramenta de prototipagem. As interfaces Web são criadas rapidamente e sem grandes dificuldades por parte dos utilizadores.

Até agora pudemos identificar as seguintes áreas de intervenção:

Páginas pessoais - Com o *Metamorphosis* tem sido possível criar rapidamente websites pessoais com uma aparência normalizada mesmo quando os recursos de informação têm uma estrutura diferente.

Conferências e Workshops - Como o caso de estudo indica, o sistema tem sido utilizado para suportar os websistes de algumas conferências, como o XATA (XML: Aplicações e Tecnologias Associadas⁴).

⁴<http://www.di.uminho.pt/~jcr/XML/conferencias/xata2003/>

E-Learning - Neste momento, a equipe do *Metamorphosis* está também a desenvolver uma plataforma para a produção de conteúdos baseada na tecnologia XML. Desta iniciativa, surgiram pequenos projectos que deram origem a várias aplicações XML: uma para a produção de guiões para aulas, outra para a produção de apresentações e outra para a produção de testes e exames [LRH03a]. Para integrar todas aquelas aplicações está-se a utilizar o *Metamorphosis* para a criação da interface Web.

Arquivos - Estão em curso vários projectos, em dois arquivos históricos nacionais, que visam a disponibilização do acervo documental daqueles arquivos na Web. O sistema de informação por detrás de um arquivo histórico é gigantesco e o *Metamorphosis* está a ser utilizado para prototipar a interface Web.

No entanto, o *Metamorphosis* ainda não está acabado:

- Está a ser testado com sistemas de informação de médio e grande porte;
- Está a ser adaptado para trabalhar com qualquer recurso de informação;
- Está a estudar-se um modelo relacional para os *Topic Maps* (se a ontologia fôr muito grande um documento XML não será a maneira ideal de a armazenar);
- Está a ser estudado um componente que irá permitir ao utilizador especificar o aspecto visual do website gerado (por enquanto, apenas há uma configuração da interface).

Neste momento, está a ser criada a página Web do projecto e será divulgada brevemente (também esta será criada com o *Metamorphosis*).

References

- Dan Brickley and R.V. Guha. Resource Description Framework (RDF) Schema Specification 1.0. World Wide Web Consortium, March, 2000. <http://www.w3.org/TR/2000/CR-rdf-schema-20000327/>.
- T. Berners-Lee, J. Hendler, and O. Lassila. The Semantic Web. In *Scientific American*. <http://www.sciam.com/2001/0501issue/0501berners-lee.html>, May 2001.
- Michel Biezunski and Steven R. Newcomb. Topic Maps Frequently Asked Questions, September, 1999. www.infoloom.com.
- Sean Bechhofer, Frank van Harmelen, Jim Hendler, Ian Horrocks, Deborah L. McGuinness, Peter F. Patel-Schneider, and Lynn Andrea Stein. Web Ontology Language (OWL) Reference Version 1.0. World Wide Web Consortium, November, 2002. <http://www.w3.org/TR/owl-ref/>.
- B. Chandrasekaran. What Are Ontologies, and Why do We Need Them? In *IEEE Intelligent Systems and their applications*, volume vl 9, n 1. IEEE, Jan/Fev 1999.
- DARPA. DAML. Darpa Agent Markup Language Program., 2001. <http://www.daml.org/>.
- Charles F. Goldfarb and Paul Prescod. *XML Handbook*. Prentice Hall, 4th edition, 2001.
- Nicola Guarino. Semantic Matching: Formal Ontological Distinctions for Information Organization, Extraction, and Integration. In *Information Extraction: A Multidisciplinary Approach to an Emerging Information Technology*, pages 139–170. Springer Verlag, <http://www.ladseb.pd.cnr.it/infor/Ontology/Papers/SCIE97.pdf>, 1997.
- Giovani Librelotto, José C. Ramalho, and Pedro R. Henriques. ADRIAN – a platform for e-learning content production. In *Second International Conference on Multimedia and Information & Communication Technologies in Education (m-ICTE 2003)*, 2003.

- Giovani Librelotto, José C. Ramalho, and Pedro R. Henriques. TMBUILDER: Um Construtor de Ontologias baseado em Topic Maps. In *XXIX Conferencia Latinoamericana de Informática – CLEI*, 2003.
- Ora Lassila and Ralph R. Swick. Resource Description Framework (RDF) Model and Syntax Specification. World Wide Web Consortium, February, 1999. <http://www.w3.org/TR/REC-rdf-syntax>.
- OIL. Welcome to OIL, 2002. <http://www.ontoknowledge.org/oil>.
- Steve Pepper. The TAO of Topic Maps - finding the way in the age of infoglut. Ontopia, 2000. <http://www.ontopia.net/topicmaps/materials/tao.html>.
- Jack Park, Sam Hunting, and Douglas C. Engelbart. *XML Topic Maps: Creating and Using Topic Maps for the Web*. Prentice Hall, 2003.
- Steve Pepper and Graham Moore. XML Topic Maps (XTM) 1.0. TopicMaps.Org Specification, August, 2001. <http://www.topicmaps.org/xtm/1.0/>.
- José Carlos Ramalho and Pedro Henriques. *XML & XSL Da Teoria à Prática*. FCA Editora, 2002.
- John F. Sowa. *Knowledge Representation: logical, philosophical and computational foundations*. Brooks/Cole, 2000.
- W. Swatout and A. Tate. Ontologies. In *IEEE Intelligent Systems and their applications*, volume vl 14, n 1. IEEE, Jan/Fev 1999.

Índice de Autores

Afonso, Pedro	37
Antunes, Pedro.....	2, 68
Aparício, Manuela	17
Carriço, Luís.....	2, 48
Carvalho, João Álvaro.....	1
Chambel, Teresa	48
Costa, J. Paulo	17
Duarte, Carlos	48
Ferreira, Ricardo	37
Figueiredo, Pedro M.	81
Guimarães, Nuno	48
Henriques, Pedro.....	28, 91, 103
José, Rui	81
Libreloto, Giovani	103
Marcos, Adérito	61, 81
Marreiros, Filipe	61
Martins, Ricardo	28
Mendes, Giovana	91
Mourão, Hernâni	68
Oliveira, Alexandre	81
Pêra, Vítor	37
Ramalho, José C.	103
Rocha, Jorge	28
Sá, Filipe	37
Sá, Vítor	61
Silva, Nuno	91
Simões, Hugo.....	48

COOPMEDIA 2003
2º Workshop de Sistemas de Informação Multimédia, Cooperativos e Distribuídos
ISEP, Porto – Portugal
8 de Outubro de 2003
Actas

Adérito Marcos
Pedro Antunes
Nuno Guimarães
(editores)

Publicação:
Associação CCG / ZGDV – Centro de Computação Gráfica
Rua Teixeira de Pascoais, 596
4800-073 Guimarães, Portugal
Telefone: +351 253 439 300 Fax: +351 253 439 3048
Email: info@ccg.pt URL: www.ccg.pt

Relatório Interno CCG

Impresso em
Outubro 2003

