Decision Follow Up Using Task Management and Semi-Structured Messages

Von der Fakultät für Mathematik, Informatik und Naturwissenschaften der Rheinisch-Westfälischen Technischen Hochschule Aachen zur Erlangung des akademischen Grades einer Doktorin der Naturwissenschaften genehmigte Dissertation

vorgelegt von

Diplom-Informatikerin

Carla Valle

aus Petrópolis, Brasilien

Berichter: Universitätsprofessor Prof. Dr. Matthias Jarke
Universitätsprofessor Prof. Wolfgang Prinz, Ph.D.

Tag der mündlichen Prüfung: 3. Mai 2006

Fraunhofer Series in
Information and Communication Technology

Band 1/2007

Carla Valle

Decision Follow Up Using Task Management and Semi-Structured Messages

Shaker Verlag
Aachen 2007

Bibliographic information published by the Deutsche Nationalbibliothek
The Deutsche Nationalbibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data are available in the Internet at http://dnb.d-nb.de.

Zugl.: Aachen, Techn. Hochsch., Diss., 2006

Printed in Germany.

ISBN 978-3-8322-6085-9
ISSN 1612-4863

Shaker Verlag GmbH • P.O. BOX 101818 • D-52018 Aachen
Phone: 0049/2407/9596-0 • Telefax: 0049/2407/9596-9
Internet: www.shaker.de • e-mail: info@shaker.de

Die Institute des Fraunhofer-Verbunds Informations- und Kommunikationstechnik (IuK) entwickeln gemeinsame Strategien für die anwendungsorientierte Forschung.
In der Reihe „Fraunhofer Series in Information and Communication Technology“ publizieren die IuK-Institute Dissertationen, Habilitationen sowie herausragende Diplomarbeiten, wissenschaftliche Monographien und Forschungsberichte. Diese Publikationen dienen der Fundierung der anwendungsnahen Fraunhofer-spezifischen Forschung und Vorlaufforschung.

Vorwort des Institutsleiters

Das Fraunhofer-Institut für Angewandte Informationstechnik FIT beschäftigt sich seit seiner Gründung Anfang der 80er Jahre intensiv mit Fragen der rechnergestützten Kooperation und hat den Forschungsbereich CSCW zeitgleich mit den ersten Aktivitäten am MIT in Europa begründet. Das erste Email-System und das erste Workflow-System in Europa waren ebenso Ergebnisse dieser Forschung wie das bis heute im Einsatz befindliche Informations- und Kommunikations-System PARLAKOM des Deutschen Bundestags und das offene Internet-Groupwaresystem BSCW (Basic Support for Cooperative Work), das 1996 den Europäischen Software-Innovationspreis gewann und mittlerweile von fast einer Million Menschen weltweit genutzt wird. An diese Tradition knüpft auch die vorliegende Arbeit an.

Ausgangpunkt der Forschung von Dr. Carla Valle war die Beobachtung, dass zwei wichtige Anwendungsbereiche der Informatik – die Entscheidungsunterstützung (Decision Support Systems, DSS) und die Unterstützung der Handlungskoordination (Workflow) – bisher weitgehend getrennt betrachtet worden sind. Die in der Praxis offensichtliche Tatsache, dass die mit DSS entstandenen Entscheidungen durch kooperatives Handeln auch umgesetzt werden müssen und dass DSS als Entscheidungsbausteine in Workflows eingebaut werden können und müssen, ist zwar abstrakt in einigen wenigen konzeptionellen Prozessmodellen berücksichtigt, aber die entsprechenden Unterstützungskonzepte und Werkzeuge sind nicht wirklich miteinander verzahnt.

Das Buch widmet sich genau dieser Verzahnung von Entscheidungsunterstützung und kooperativer Aufgabenunterstützung. Dabei geht es vorwiegend vom Szenario zeitsynchroner gemeinsamer Entscheidungen in Kombination mit einer durch Aufgaben-Management und halbstrukturierte Nachrichtensysteme unterstützten asynchronen Umsetzungsphase aus. Wesentliche Forschungsbeiträge sind ein integriertes konzeptuelles Modell, das argumentative Entscheidungsfindung, Meeting-Support, daraus abgeleitetes sprechaktbasiertes Projektmanagement und das dazu gehörige Dokumentationswesen miteinander verknüpft. Eine gelungene prototypische Umsetzung und ihre umfassende projektbegleitende Evaluierung demonstrieren, dass dieses Konzept durch eine geschickte Kombination relativ einfacher Erweiterungen von Standard-Werkzeugen wie dem BSCW und der Microsoft-Email praktisch umgesetzt werden kann, ohne dass dem Benutzer dadurch wesentlicher zusätzlicher Lernaufwand entsteht.

Das vorliegende Buch wurde im Mai 2006 als Dissertation an der RWTH Aachen angenommen. Erstgutachter war Prof. Dr. Matthias Jarke, weitere Gutachter: Prof. Wolfgang Prinz Ph.D. und Prof. Dr. Carla Simone aus Mailand. Für die LeserInnen dürfte neben der Ableitung und Darstellung des eigenen Ansatzes auch die vergleichende Darstellung existierender Ansätze in den Bereichen Decision Support und Meetingware von besonderem Interesse sein, die in dieser Breite anderswo kaum vorzufinden ist. Ich wünsche dem Buch eine zahlreiche Leserschaft!

Aachen und Sankt Augustin, im Februar 2007

Prof. Dr. Matthias Jarke
Institutsleiter Fraunhofer FIT

Zusammenfassung

Die Beschaffung von Informationen aus verschiedenen Quellen und der Austausch dieser Informationen mit anderen ist für viele Menschen eine wichtige Anforderung für die Durchführung ihrer beruflichen Tätigkeiten geworden. Daraus ergeben sich ebenfalls gestiegene Anforderungen für Koordinierung und Zusammenarbeit, insbesondere in Teams. Ein Aspekt dieses Zusammenhangs, der seit einigen Jahrzehnten untersucht wird, betrifft den Entscheidungsfindungsprozess und die Verwendung von Systemen zur Unterstützung dieses Prozesses. Die Phase der Entscheidungsumsetzung wurde insbesondere für Entscheidung, die aus Gruppensitzungen resultieren wurden dagegen bisher nur selten untersucht.

Diese Arbeit analysiert die Konzepte, die mit diesem Problem verbunden sind vom Standpunkt der Entscheidungsunterstützungssysteme (Decision Support Systems) und der Unterstützung von Gruppenarbeit durch Computer (Computer Supported Cooperative Work). Auf der Basis einer Analyse einschlägiger Arbeiten und Literatur zeichnen wir die Entwicklung zum aktuellen Stand in diesen beiden Bereichen nach und zeigen, dass die Entscheidungsumsetzung dabei nicht häufig betrachtet wurde

Anschließend definieren wir den Fokus und Umfang unseres eigenen Ansatzes und schlagen eine Lösung vor, die Unterstützung in Form automatisierter Mechanismen für die folgenden Tätigkeiten anbietet: i) die Formalisierung des Sitzungskreislaufs bestehend aus den Phasen vor, während und nach einer Sitzung) ii) die explizite Notation von Entscheidungen iii) die effektive Nachverfolgung durch einen integrierten Ansatz aus Aufgabenverwaltung und E-Mail-Austausch. Wir erläutern das konzeptuelle Modell und die Theorien, die uns für unsere Forschung als Grundlage gedient haben und wir zeigen wie wir sie in Form einer Erweiterung für ein Dokumentenmanagementsystem umgesetzt haben.

Abschließend stellen wir die durchgeführte Evaluation des Systems und die erhaltenen Ergebnisse vor, zeigen den Mehrwert unseres Ansatzes und die bei der Umsetzung zu berücksichtigenden Grenzen auf und leiten daraus einige Hinweise für weiterführende Forschung ab.

Abstract

Presently, organizational scenarios demand employees to exchange and obtain information from various sources to accomplish their duties. Said in other words, there is an increasing need for team work, for collaboration and for coordination. One aspect of this scenario that has been under investigation for decades concerns the decision making process, especially the use of decision support systems. However, very little has been done regarding the decision implementation phase, in particular about decisions made in meetings.

This work analyses the concepts related to this problem looking at it from the viewpoints of Decision Support Systems and the Computer Supported Cooperative Work. We visit the literature and related work in both areas and show how they evolved to the state-of-the-art, and how decision implementation was not often included in the various initiatives.

After that, we define the focus of our own approach and scope, and propose a solution based on automated mechanisms to assist the activities of formalizing the meeting cycle (i.e. pre-meeting, meeting and post-meeting phases), representing decisions explicitly, and obtaining follow ups through an integrated approach based on task management and e-mails exchange. We present the conceptual model and theories that were the basis of our research and show how we have implemented them as an extension of a Document Management System.

Finally, we discuss the evaluation realized with the system and the results obtained. We conclude this document pointing out the added value of our approach, the limits we had to deal with along the process and we indicate some directions for future research.

ACKNOWLEDGEMENTS

Every journey starts with a first step. Thus, I believe this work started a long time ago, probably motivated by my early Science classes while I was at Escola Viva and CEN, both in Brazil. It was there, where my curiosity grew; where I wanted to understand everything. The years passed, I grew up and there I was, starting my research in Rio. I was still curious, but more “modest”, since I did not want to understand everything anymore, “just” CSCW. A little later and I took a big step, and moved to Germany, where almost all the concrete parts of this thesis were realized. There I learned a few things about CSCW. I have also learned that a Ph.D. thesis starts on a large scale, then it gets focused, and if it is possible, focuses even more…. and here I am, with about 200 pages. I guess I still have some lessons to learn! Years went by and there it was time for a new move to the USA, where I have finalized this work. And now I can see I am still curious, “Gott sei Dank”, but more conscious that I can only understand a few things—and that is already a lot. This thesis is the result of this process.

In a world full of information, I ask the question: what is the relevance of my PhD thesis? Independent of the answers to this question, which may vary, all these years were very relevant to me. I gained a lot, but I also lost a lot. I had to deny many things and moments with beloved ones, and I questioned in various occasions if it was worth it. Here we are! Therefore, I would like to use this space to apologize for my lack of attention and time, and to thank all of you who helped me overcome the barriers to arrive here today.

Prof. Matthias Jarke thank you for supporting my stay at FIT and in Aachen, for encouraging my Ph.D. ideas, for the opportunities in conferences, and mainly for helping me get to the end of this thesis.

Prof. Wolfgang Prinz thank you for all your support over these years! Since our first meeting at CSCW 2000 you have provided me with great opportunities, interesting conversations and thoughtful comments, motivation whenever necessary, and all these factors have contributed to help me make my dreams come true.

Prof. Carla Simone thanks for the interesting exchanges and for accepting being one of my evaluators.

Markus, my dearest, thanks for the shared coffees, beautiful smiles, support, conversations and mainly, for your love.

Mãe, pai, meus irmãos Jackson e Mac, eu sem vocês, não sei nem por quê... porque sem vocês simplesmente não faria sentido!!! (Mom, dad, my brothers Jackson and Mac, without you I would not know the reasons, because without you there would be no reasons!!!).

My nieces: Isabella, Maria Victoria and Raquel, I hope I assist you in seeing how important it is to believe your dreams can come true. Your smiles are my inspiration! Thanks for loving me even when I am a “disappearing aunt”.

I would like to especially thank four friends of mine, four women, who also hold a Ph.D., and have provided me with the spiritual energy to finish this work. I guess they do not know how important their examples were and are for me! Ana Regina C. da Rocha, Elaine Raybourn, Claudia Gama and Kathia Marçal: you were present (in my thoughts, most of the time, but also physically sometimes) in every good and difficult moment I had to finish this thesis. When I got a step done, I was smiling and thinking of you four! When I was thinking "I cannot do it anymore…" I was remembering our exchanges and your own strengths and using them as examples to keep me going further.

Last and definitely, not least, I would like to thank the support of all my friends and colleagues, who have contributed to this process in various ways:

Prof. Marcos Borges for the support, discussions and opportunities which helped me start my Ph.D. research, grow, and finish this work.

John Koh thanks for helping me with the implementation, ideas, discussions and cultural exchanges. Christian Schmitt thanks for helping with our cute report! Deivyd, thanks for the distant assistance during the weekends!

From Brasil: Vó Creuza, Tia Inez, Tia Regina, Tia Rose-May, Catarina, Aninha (Cog), Beto (BBB), Débora Matos, Jorge, Mônica Ferreira, Bruno, Carla Delgado, Simone, Hugo, "WEB-RSL" :-), Pedro Paulo, Geraldo, Nicolas, Mariella, Zé Maria e Flavia.

From Europe: Ana, Dado, Lucia, Luciano, Jasmina, Kresimir, Alexander, Jasminko, Claudia, Rolf, Martin, Christoph, Gauri, Fabio, Jalusa, Ohm, Gernot, Ismael, Yone, the Multi-Culti group, Benoît and Marcello!

From my time at FIT-CSCW: Gabi, Peter Seuffert, Herr Hermann, Uta (especially for your availability and support during the evaluation phase, many thanks!), Wolfgang Gräther (for all your support and various discussions at the famous "Bonn Goût"), Elke, Thomas Koch, Leonie, Tom Gross, Gerry, Karl-Heinz, Rudolf (vielen, vielen Dank!), Gerd, Wolfgang Appelt, Sabine, Konrad, Thomas Kreifelts, Peter Mambrey, Irma, Jan, Wido, and all the others for their smiles, the joint lunchs and their friendly "Guten Tag!". How could I forget the glasses of Kölsch we drank together?!

From FIT-ICON: Barbara, Reinhard, Marcus K., Marcus S., Markus E., and Milos thanks for the shared moments and for the support you gave to me in the evaluation phase!

From the RWTH: Hoeppi, Aida, Michalis, Dominik L., Ralf, Christoph, Tatiana and Reinhard for the assistance over the months I worked with you!

TABLE OF CONTENTS

List of Figures

List of Tables

1 Introduction

Presently, organizational scenarios demand that employees exchange and obtain information from various sources to accomplish their duties. Said in other words, there is an increasing need for team work, for collaboration and for coordination. Consequently, the use of collaborative (Mark & Poltrock, 2003) and coordination technology (Malone & Crowston, 1990 and 1994) becomes a reality.

However, the existing group support systems solve only part of these complex scenarios. But why aren't these systems enough? One reason for that is the lack of integration among them. There are many tools available, but one does not exchange information with another. Consequently, there is information and data stored in different systems, but integrated views, which could bring contributions to collaboration and coordination activities, are hardly seen.

This thesis aims at discussing specifically one of these problems: the problem of decision follow up, and how a set of integrated groupware, added of a decision management module, can provide means to represent and trace decisions made in meetings.

Thus, we extended a groupware, so that it could support meeting and decision management. Additionally, we have integrated them to a task management module, and to a semi-structured message module. The decisions' follow ups are mainly achieved through this integrated approach together with a set of specific reports.

1.1 Problem Description

One important area vastly investigated in the literature is the area of Decision Support Systems (DSS). There are many definitions of DSS in the literature. According to (Sprague & Carlson, 1982) cited in (Power, 2004a), DSS comprise a class of information systems that draws on transaction processing systems and interacts with the other parts of the overall information system to support the decision-making activities of managers and other knowledge workers in the organization. According to (Power, 2004a), a DSS is a specific software/hardware system for use in specific situation as part of a decision process.

More important than trying to find a unique definition of DSS is to understand how they evolved over the years. For several decades, domain specific or generalist research has been done in this area, but many problems remain open to be investigated and solved. One of them is related to decision implementation. Once a decision is made, there is lack of information about how its implementation take place, who is involved and making use of which resources, what are the current problems.

The mechanisms used nowadays to provide information about decision implementation are based more often on user's efforts than on systems' efforts, i.e. through informal face-to-face conversations, or via telephone, or via electronic communication, like emails and instant messaging. However, a lot of data is constantly created by users, either through documents, or through electronic communication, which could be used to automatically provide some levels of follow up to decision makers and to decision implementers.

One aspect of this problem is how to get the right information from these electronic elements to provide improvements on decision implementation. Or more specifically, how can technology help decision makers and decision implementers to: (i) be reminded of things to be done that have impacts on decisions outcomes, (ii) answer questions on how decisions were made and why, and (iii) get in an easy and quick view the understanding about the status of decisions already made. As a consequence of this track support, project managers can possibly learn lessons and improve the quality of future decisions.

Since we cannot discuss all means where people make decisions, in order to define the follow up possibilities, we have reduced the scope of this thesis only to decisions made in meetings. Our choice is based on the fact that meetings are one of the most common places, for project team work, where decisions are made and, mainly, possible to be documented electronically. Doing so, we embraced in our research many aspects of a second research area, Computer Supported Cooperative Work (CSCW), in general, and specifically the aspects related to coordination and meetings.

Good coordination is nearly invisible, and we sometimes notice coordination most clearly when it is lacking (Malone & Crowston, 1990). Nowadays, with the dissemination of globalized organizations the role of coordination became even more important than in the past. Teams are working in geographically distributed sites, teams are cross-cultural and multi-disciplinary. In order to support this type of project team work, we still need to research and make proposals on methodologies, methods and tools to support the project management itself, as well as core activities like communication and exchange of information, which are all part of coordination. In our specific case, we are dealing with the coordination of artefacts that are related to decisions made in meetings, especially using elements which are already part of many work scenarios.

Coordinating how decisions are implemented with the support of technology, in order to provide better project outcomes, is one of the needs of a project manager. Much work has been proposed in the area of CSCW to support meeting activities, and these systems are known as “meetingware”. After an extensive review of the relevant literature and the investigation of products available in the market, we concluded that a solution to support decision follow ups, of decisions made in meetings, would have to encompass not only the meeting itself, but also the activities for preparing a meeting and, mostly, the activities after the meetings. Thus, a more holistic and integrated approach had to be considered, involving a complete meeting cycle, which is composed of pre-meeting, meeting and post-meeting support. In all these phases, decisions are made explicit and possible to be managed.

1.2 Proposal and Research Questions

Several initiatives were and still are done in the direction of capturing the rationale generated during interactions in a decision-making process. This research considers a different perspective, which is focused on the steps that take place after decisions are made, and our main emphasis is on decisions made in meetings.

The proposal is based on the formalization of meeting life-cycle (pre-meeting, meeting and post-meeting), with decisions made explicit, and a set of mechanisms integrated, in order to provide awareness information about decision implementation, based on tasks and conversations, i.e. sets of messages.

With these ideas, our main goals are:

- To improve the communication between decision makers and meeting participants with decisions implementers.
- To provide means to support the track of decisions, using the data team members already generate "naturally" (i.e. tasks and emails).
- To improve users' awareness of problems in decision implementations, through a set of reports.

We think that the information and knowledge generated during a decision implementation, if captured, contextualized and put on disposal, can contribute to improve coordination along project development.

The research questions of this thesis are the following:

Research Question 1

Can a set of automated group support mechanisms that enables the link between pre-meeting, meeting and post-meeting activities, representing decisions explicitly, provide a basis for decision tracking, allowing for better coordination during project development?

Research Question 2

Can common electronic elements, which belong to daily project scenarios, i.e. tasks and emails, provide means for creating awareness of problems that occur during decision implementation?

Our intention is motivated by the fact that through these experiences of improving the coordination of decisions, while they are being implemented, together with the documentation we are proposing, project members can review their experiences to allow for improvements in future processes, and possibly, to improve new decisions that are still to be made.

1.3 Research Contributions

The first motivation for this thesis started with the practical experience of the author as a member of different software development teams, either as a programmer, system analyst, researcher or coordinator. Through informal observations, the author saw herself and other team members with difficulties in knowing the status of decisions made in the projects, and the negative consequences of this lack of information. Redundant discussions, rework and conflicts are some of them. This first interest lead to the research realization, which included the review of the literature in the areas of DSS and CSCW concerning, especially, the support of decision implementation and meeting support systems.

The result of these activities confirmed the gap of solutions focused on decision implementation, in general, and also, in particular, of decisions made in meetings. We did not find anything specifically addressing this problem, but various initiatives deal with important aspects of the problem.

Additionally, a set of in-depth interviews with project managers was realized. Our main goal with these interviews was to find out if the problem of decision follow up was valid and worth being investigated from a practical point of view.

We got two results from these activities. First, that the problem was valid and worth being further analysed. Second, a list of requirements for building a computer supported solution for this problem.

Based on those, we made an integrated system proposal that includes a set of elements we considered of high priority to solve part of the problem, and according to this proposal, we have built a prototype used in our evaluations.

The main contributions of this thesis are the following:

From the DSS perspective, we have proposed a concrete way of tracking decision implementation, through the representation of decisions made in meetings, and through the integration of task management support and emails to a document management system.

From the CSCW perspective our main contribution is a proposal that encompasses the whole life cycle of meetings. First, proposing automated and structured support for meeting preparation and meeting documentation, and second, the integration of these two phases with a post-meeting support through group systems that are normally used separately. Another contribution still in this area is the use of asynchronous communication and the language action perspective in the problem of coordinating decision implementation.

1.4 Thesis Outline

Chapter 2: Provides a detailed study of decisions, decision support systems and related concepts, starting from the fundamental work of (Anthony, 1965), (Mason, 1975), (Simon, 1976), and (Alter, 1980), passing through the areas of GDSS, with relevant work from (Bui & Jarke, 1986), (DeSanctis & Gallupe, 1987), and of ODSS from (Santhanam et al., 2000), going until more recent contributions from (Balasubramanian et al., 1999), (Courtney, 2001) and Power (1999, 2000, and 2004). In this chapter we also show how the definition of DSS has changed over the years, until its more recent and modern definition, which classified other systems like document management systems, knowledge management systems, amongst others, as DSS. With this study we have learned that DSS are still an interesting research area with open questions to be solved. One of them is of our particular research interest: the decision follow up during the decision implementation phase. Based on this interest, we have discussed the elements which are part of the problem in the second part of the chapter. The third part of the chapter discusses a list of tools available in the market, using the taxonomy suggested in (Power, 2004a) of modern DSS. At the end of the chapter we show a comparative table where we can observe the qualities and the deficiencies of these tools regarding the support for decision follow up activities.

Chapter 3: Discusses the related work in the field of CSCW, especially concerning meetingware. We start the chapter discussing shortly the beginning of the CSCW area from (Ellis et al., 1991), (Johansen, 1988) and (Grudin, 1994) and how this area and the area of DSS influenced each other with the definition of GSS and GDSS. Then we dedicate some sections to discuss specifically the concepts of Electronic Meeting Systems (EMS) using the work of (Dickson et al., 1989), (Streitz et al., 1994), (Nunamaker et al., 1995), (Nunamaker et al., 1996), and (Rodriguez & Favela, 2001). In these sections we provide information about the elements which are involved in the research and usage of these applications, like calendars, agendas, meeting minutes, etc. Then, we dedicate a couple of sections to discuss in more detail the problem of our interest, i.e. the join between meetings and decision follow up. Thus, we provide indicators of the main elements that should be taken into account when studying the scenario of our research, and we emphasize the discussion in the aspects related to electronic communication and task management, which are especially related to our proposal approach. The third part of the chapter discusses a list of tools available in the academia and in the market. At the end of the chapter we show a comparative table where we can observe the qualities and the deficiencies of these tools regarding the support for the complete meeting cycle and especially for the decision follow up problem.

Chapter 4: Defines the conceptual design for a decision follow up support mechanism. We start the chapter showing a list of requirements and barriers used as a basis for our system design. We then discuss two theories which influenced and guided the development of the concepts in our work: the theory of Coordination from (Malone & Crowston, 1994) and the theories of language, starting with the work of (Austin, 1962), (Searle, 1969), and more recently the work of (Habermas, 1984) and (Winograd & Flores, 1986). While discussing these two theories we show similar work from the areas of CSCW and IS, for e.g. (Malone et. al, 1986a), (Michelis & Grasso, 1994), (Simone & Divitini, 1999) and (Schoop et al., 2003). The most extensive part of this chapter is the description of each relevant concept of our proposal, presented in a top-down approach, starting from the concept of a project coordination repository, including three highly important concepts of issues, decisions, and tasks, then, the discussion of the role of conversation in our model, the complete meeting support cycle, ending with the means of information visualization proposed as a set of reports.

Chapter 5: Discusses the details of the implementation of the concepts presented in chapter 4. We propose a system architecture, which was mainly implemented as an extension of a document management system, the BSCW system, and its integration with an email client—Microsoft Outlook. The COORDE prototype is then explained in details, and we illustrate how the system could be used through a ficticious scenario.

Chapter 6: Describes the various aspects used in the evaluation phase. We start the chapter discussing the objectives of our evaluation related to the research questions. Then, we briefly provide an explanation about qualitative evaluation methods, compared to quantitative methods, and we show which methods were selected in our particular case. We present the activities and results obtained in each evaluation activity, starting with an early formative evaluation done, at the requirements definition phase, and ending with two summative complementary evaluations.

Chapter 7: Summarizes the results of the thesis. We discuss the role of the two theoretical approaches, and propose some future research directions that can be added to the present work.

2 Decisions and related concepts

2.1 Introduction

The first concept to be clarified in the scope of this research is the concept of decision, which is related to the verb "to decide" (*lat. decidere*). The verb "to decide" means *to deliberate, to determine, to solve* and it normally involves, as a consequence of the "decision(s)", a list of actions or deliberations to be executed.

For (Mason, 1975) **decisions are said to consist of source, data, prediction and inference, value and choice, and action components**. Balasubramanian (Balasubramanian et al., 1999) defines **decision as an action typically made to achieve a goal, to solve a problem, or to implement a plan**. And in another source, the Georgetown Law Library (Williams, 2001), **decision is judge's written conclusion that is arrived at after some period of consideration**.

For Simon (Simon, 1976), the **rational decision making**, is a process of choosing among alternatives that involves a series of steps:

- Listing all the alternative strategies;
- Determining all the consequences that follow upon each of these strategies;
- Comparatively evaluating these sets of consequences.

Nevertheless, the same author asserts that the word "all" is not completely adequate to represent the reality, since nobody would be able to know or to analyse "all" the alternatives and consequences of a decision. The number of possibilities would be too great, and the information would be so vast that even an approximation of the reality would be hard to be conceived.

The fact is that, within the scope of organizations, managers need to make decisions in a frequency that can be hardly imagined. Decision is a dynamic that takes place everyday in organizations involving groups, using different rationales and experiences to choose the best or more adequate option for their business processes.

Although managers do not deal with "all domains", they still deal with complex ones. They need to leave the "irresolution" scenario to go to a "resolution" scenario, and this process involves decisions. Winograd and Flores (Winograd & Flores, 1986) call this process of going from irresolution to resolutions "deliberation". The principal characteristic of deliberation is that it is a kind of conversation (in which one or many actors may participate) guided by questions concerning how actions should be directed. Sometimes it is possible to specify conditions of further inquiry to attain a resolution. On the other occasions the resolution will come from a more prolonged hesitation and/or debate. Only in some of these cases will the phenomenon of choosing between alternatives occur, and a process of ranking according to some metric or other criterion may occur even less frequently.

But, as asserted by Winograd and Flores, the **decision process involves other dimensions**, such as the **type of the decision**, **the number of alternatives and their**

complexities, the number of people related directly or not, and the experience of those involved, just to list a few examples.

Regarding the **type of decision**, an interesting approach was cited in (Aggarwal, 2001), which classifies decisions as:

- Independent – where one person has full responsibility and authority to make a decision, i.e., a person performs the task without interacting with other persons;
- Sequential interdependent – a decision maker makes part of a decision, which is then passed on to the next person.
- Pooled decision – where decision results from negotiations between two or more people.

Understanding to which organizational level, i.e. strategic, management or operational control levels, the decision is related to is a relevant aspect while dealing with decision making processes (Anthony, 1965).

The *strategic control level* is defined in (Anthony, 1965) as the process of deciding on objectives of the organization, on changes in these objectives, on the resources used to attain these objectives, and on the policies that are to govern the acquisition, use, and disposition of these resources. A major problem in this area is predicting the future of the organization and its environment. Besides that, it normally involves a small number of high-level people who operate in a non-repetitive and often very creative way.

The *management control level* is defined in (Anthony, 1965) as the process by which managers assure that resources are obtained and used effectively and efficiently in the accomplishment of the organization's objectives. The activities embedded in this level involve a lot of personal interaction and, additionally, they take place within the context of the policies and objectives developed in the strategic planning process. Finally, the paramount goal of management control is the assurance of effective and efficient performance.

The *operational control level* is defined in (Anthony, 1965) as the process of assuring that specific tasks are carried out effectively and efficiently. The basic difference between operational and management control is that the first is concerned with tasks (such as manufacturing a specific part) whereas management control is most often concerned with people.

As pointed by Anthony (Anthony, 1965), the boundaries among these three phases are often not clear. Nevertheless, respecting this division helps us understand where computers can be useful and how. For example, in (Gory & Scott Morton, 1975) the authors state that information requirements demanded in each areas are very different. For instance, the information needed by strategic planners is aggregate information, and obtained mainly from sources external to the organization itself. Both the scope and variety of the information are quite large, but the requirements for accuracy are not particularly stringent. Finally, the no routine nature of the strategic planning process means that the demands for this information occur infrequently. While the information need for the operational control area stand in sharp contrast to those of strategic planning. The task orientation of operational control requires information of a well-defined and narrow scope. This information is quite detailed and arises largely from sources within the organization. Very frequent use is made of this information, and it must therefore be accurate. While the information requirements for management control fall between the extremes for operational control and strategic planning. In addition, Rappenport (Gory & Scott Morton, 1975) complements its definition suggesting that much of the information relevant to management control is obtained through the process of interpersonal interaction.

A classic reference from the decision making literature (Simon, 1977), defines that a **decision can be structured** or **unstructured**, also known as **programmed** or **non-programmed.** For this author decisions are programmed to the extent that they are repetitive and routine, to the extent that a definite procedure has been worked out for handling them so that they don't have to be treated *de novo* each time they occur. Whereas decisions are non-programmed when they are novel, unstructured, and consequential. There is no cut-and-dried method of handling the problem because it hasn't arisen before, or because of its precise nature and structure are elusive or complex, or because it is so important that it deserves a custom-tailored treatment. By non-programmed the author explains it as a response where the system has no specific procedure to deal with situations like the one at hand, but must fall back on whatever *general* capacity it has for intelligent adaptive, problem-oriented action.

Following the three different organizational levels cited in (Anthony, 1965), at the strategic control level, examples of decisions and their respective types are: warehouse of factory location (structured), mergers and acquisition (semi-structured), new product planning (unstructured). In the management control level examples of decisions and their respective types are: budget analysis (structured), overall budget (semi-structured), and sales and production (unstructured). While in the operational control level, the examples of decisions and their respective types are: receivement of orders (structured), production scheduling (semi-structured) and grievances (unstructured).

Another approach that comes from the decision theory literature citing Raiffa in (Svenson & Benthorn, 1992) classifies decisions as under **uncertainty** or risk and decisions under **certainty**. Gambles and lotteries are the prototypes for risky decisions, while decisiond on pieces of furniture, for example, are decisions under certainty. This distinction reflects decision theory's emphasis on separating treatments of the evaluation of a consequence of a decision.

Understanding what a decision is and describing some of its dimensions, we can pass then to the understanding of the second concept involved in this research: the decision making process.

2.2 Decision making process

Similar to the concept of decision, decision-making process has also various definitions. Ijiri (Ijiri et al., 1975) provides a simplistic, but interesting definition of the **decision making process**, **as a process characterized by three factors: (a) decision inputs, (b) decision outputs, and (c) a decision rule**. Decision inputs are factors, which are considered by the decision-maker in making his decision. Decision output are decisions made by the decision-maker. A decision rule is a rule by which a set of decisions is associated with a set of decision inputs. For example, in an investment decision, let us assume that investments are to be based only upon the rate of return of the project (say, invest if the rate of return is greater than or equal to 15 per cent per year; otherwise, do not invest). The decision rule is to associate all rates of return greater than or equal to 15 per cent per year with the decision output "invest" and all other rates of return with the decision output "do not invest". Of course this scenario could be much more complicated than explained by Ijiri. But as a matter of fact, what this author wants to emphasize is the existence of a relation between inputs, outputs and relevant factors.

Simon (Simon, 1977) offers another way of representing the **decision making process as a set of four activities**, which are related to a definition from 1910 from John Dewey:

- What is the problem?
- What are the alternatives?
- Which alternative is best?
- How to carry out the decision?

Balasubramanian (Balasubramanian et al., 1999) defines **decision-making as a process of selecting from a set of options the alternatives that are most likely to lead to desired outcomes**. The process entails various steps and stages that decision-makers engage in, either explicitly or implicitly. It involves several steps: the definition on the context and purpose of the decision, the identification or generation of options to be considered, the specification of factors, assumptions, reasons and other relevant information to be considered, the assessment of options against relevant factors, assumptions and other variables to make the decision and the enact of the decision and the reviewing the results.

Courtney (Courtney, 2001) defines the **decision-making process supported by a Decision Support System** (DSS) Environment, as presented in Figure 2-1. **In this model the emphasis came to be on model development and problem analysis**. Once the problem is recognized, it is defined in terms that facilitate the creation of mathematical models. Alternative solutions are created, and models are then developed to analyze the various alternatives. The choice is then made and implemented.

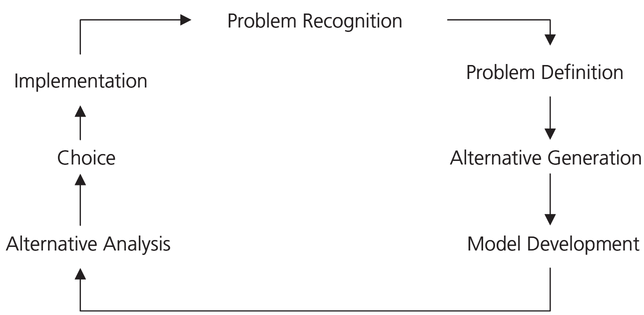


Figure 2-1: The conventional DSS decision-making process (Courtney, 2001)

The same author comments that normally, decision-making processes are not so linear and regular as the figure suggests. Actually, most of the time, phases overlap and blend together and there will be recycling to earlier stages, as more is learned about the problem, as solutions do not work, and so forth.

The DSS area has evolved a lot in four parts of the cycle proposed in (Courtney, 2001), i.e., alternative generation, model development, alternative analysis and choice. But there is a big gap of support tools to assist the problem recognition, the problem definition and the implementation phases.

Courtney refines its suggestion about a decision-making process, based on a previous work of Mitroff and Linstone (Mitroff & Linstone, 1993) presented in Figure 2-2. But once more, there is not expected support for the phases mentioned in the previous paragraph.

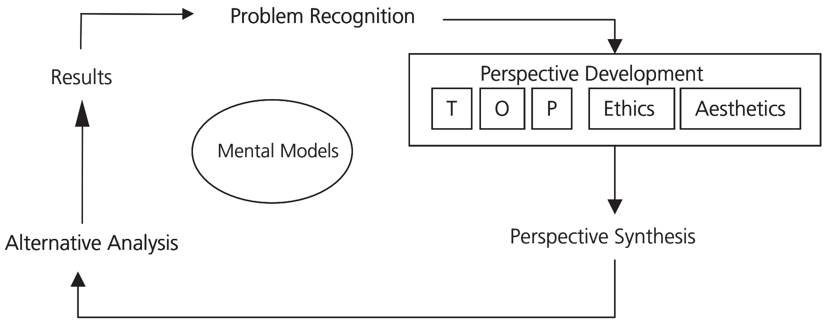


Figure 2-2: A new decision paradigm for DSS (Courtney, 2001)

Here the authors based their ideas on the fact that **decision support systems should be capable of handling much "softer" information and much broader concerns than the mathematical models have been capable of handling in the case in the past**. An analysis about the differences between the two schemes was proposed in Shim (Shim et al., 2002). In this work the authors state that the primary difference between them is the development of multiple and varied perspectives during the problem formulation phase. Mitroff and Linstone suggest that perspectives should be developed from organizational (O), personal (P) and technical (T) positions. In addition, ethical and aesthetic factors should be considered as well.

In our opinion, one possible improvement for the decision making process is to think about the holistic view, i.e., to consider all cycle phases important and then providing automatic support, if it is possible, for the whole cycle. Ideally, making the support an integrated solution. We do not deny the importance of mathematical models, which are fundamental for several simulations and analysis, but the cycle and its support tools should be seen as more than modeling the "world" in a mathematical model. There are several human related aspects in the cycle, which might be difficult to be supported by computers, but there are also other types of data, e.g. knowledge, experiences and communication, which are not included in most of the approaches and they could be somehow supported by automatic tools to contribute for making better decisions.

2.3 Decision Support Systems

Looking at decision-making processes and their related support tools, it seems that Decision Support Systems (DSS) are the tools that reached closer and better results to solve part of the decision making scenarios' issues.

The philosophy underlying DSS is that it is every bit as legitimate to use computers to improve or expedite the processes by which people make and communicate decisions (Alter, 1980). A more detailed definition, from the same reference, proposes DSS as systems that:

- **are designed specifically to facilitate decision processes,**
- **should support rather than automate decision making, and**
- **should be able to respond quickly to the changing needs of decision makers**.

The most general activities supported by these systems are:

- retrieving a single item of information,
- providing a mechanism for ad hoc data analysis,
- providing prespecified aggregations of data in the form of reports,
- estimating the consequences of proposed decisions,
- proposing decisions, and
- making decisions.

Sprague and Carlson (Sprague and Carlson, 1982) define **DSS as a class of information system that draws on transaction processing systems and interacts with the other parts of the overall information system to support the decision-making activities of managers and knowledge workers in organizations**.

Or according to Power (Power, 2004a), who proposes distinct definitions:

- **DSS are a specific class of computarized information system** that supports business and organizational decision-making activities.
- **DSS is an interactive software-based system intended to help decision makers** compile, analyze and manipulate information raw data, documents, knowledge frameworkds, and/or business models to identify and solve problems and make decisions.
- **DSS is a specific software/hardware system** for use in a specific situation as part of a decision process.

DSS include agenda setting, decision modeling methods (such as decision trees, risk analysis, forecasting methods, and multiattribute utility functions), structured groups methods (such as the Nominal Groups and Delphi techniques), and rules for directing group discussion (DeSanctis & Gallupe, 1987).

Decision Support Systems have evolved from two main areas of research: the theoretical studies of organizational done at the Carnegie Institute of Technology during the late 1950's and early 60's and the technical work on interactive computer systems, mainly carried out at the Massachusetts Institute of Technology in the 1960's (Keen & Scott Morton, 1978).

In the middle and late 1980's, Executive Information Systems (EIS) and Group Decision Support Systems (GDSS) evolved from the single user and model-oriented DSS. In the 1990's the demand for improving strategic decision making at the management level with respect to current domestic social and economic condition contributed to the evolution of DSS into data warehousing and On-Line Analytical Processing (OLAP), which began broading the realm of DSS. However, the concept of data warehousing differs from a DSS, since data warehousing systems have an enterprise-wide scope (Power, 1999).

A parallel concept, which was and is used as a basis in DSSs, was developed in the 1970s by Kunz & Rittel (Kunz & Rittel, 1970). The Issue-Based Information System (a.k.a. IBIS) was a hypertext environment used for structuring discussions' arguments. IBIS uses a classification scheme to organize the data, based on three types of node (i.e. issues, positions, arguments) and nine types of link (i.e. responds-to, questions, supports, objects-to, specializes, generalizes, refers-to, replaces). *Issues* represent anything that needs to be discussed, deliberated, and put into argumentation during a design project, or during a discussion. *Positions*, which are means to solve issues and *Arguments* that are statements supporting or objecting positions.

Several DSS have been developed since then, using the principles of IBIS. Some of them are gIBIS (Conklin & Begeman, 1988) and SISCO (Borges et al., 1999).

The IBIS model seems to be particularly useful when the problem being discussed is poorly defined, or unknown. IBIS follows a method of decomposition of the problem. The key to the system is the idea of an issue, which is in effect a question. As the discussion takes place, knowledge artifacts are generated and recorded in a schematic way. Thus, allowing for first, a better understanding of the various paths followed in the discussion, and second, the documentation of the discussion is done as part of itself. In the DSS this strategy can be used to store the rationale of decision making processes.

Although the IBIS model's popularity and its clear contribution on structuring arguments, the usage of such approach has two main drawbacks. As discussed in (Louridas & Loucopoulos, 2000), the size of the resulting rationale records, and the overhead placed on the users of the approaches when producing them. Thus, contributions on the decision representation as well on how to represent the rationale used in the decision making process are still research topics.

2.3.1 Taxonomies of DSS

There are several DSS' taxonomies proposed in the literature. Power (Power, 2004a) classifies the DSS according to the type of focus the system follows, e.g. Data-driven or Communication-driven (Table 2-1), **which is the taxonomy we are using in our work**. Another classification proposed by the same author in another reference (Power 2000), makes the differences based on specific characteristics of these systems, for example the type of users, or the type of information representation (Table 2-2).

Table 2-1: Taxonomy of DSS (Power, 2004a)

DSS Type	DSS Focus
Data-driven or Data-oriented DSS	Emphasize access to and manipulation of a time-series of internal company data and sometimes external data. Executive Information Systems and Geographic Information Systems (GIS) are special purpose Data-driven DSS.
Model-driven	Emphasize access to and manipulation of a statistical, financial, optimization or simulation model.
Knowledge-Driven	Has specialized problem-solving expertise stored as facts, rules, and procedures or in similar structures. Knowledge-Driven DSS can suggest or recommend actions to managers. These DSS are person-computer systems with specialized problem-solving expertise. The "expertise" consists of knowledge about a particular domain, understanding of problems within that domain, and "skill" at solving some of these problems. A related concept is Data Mining. It refers to a class of analytical applications that search for hidden patterns in a database. Data mining is the process of sifting through large amounts of data to produce data content relationships. Tools used for building Knowledge-Driven DSS are sometimes called Intelligent Decision Support.
Document-Driven	Manage, retrieve, and manipulate unstructured information (e.g. oral, written or video docs) in a variety of electronic formats. This type of DSS assists in knowledge categorization, deployment, inquiry, discovery and communication.
Communication-Driven	These are systems that support communication, collaboration and coordination for more than one person working on a shared task; examples include integrated tools like Microsoft Net meeting, discusssion forums and e-Mail.

Another dimension that can be analysed in DSS is the basic technology which they are based-on. The system can be centralized, client-server, or web-based. Actually, this last one has been considered a possibility for improving the rapid dissemination of “best practices” analysis and decision-making frameworks and can promote more consistent decision-making on repetitive tasks. Web-based DSS can help retrieve, analyze and display structured data from large multidimensional or relational databases, provide access to a model or expert system, provide access to multimedia documents and unstructured data, and facilitate communication and decision making in distributed teams (Power, 2004a). The same author comments that the web technologies have expanded the scope of all DSS and especially Communications-Driven and Group Decision Support System (GDSS) and that this is a good platform for delivering decisions support to a company’s customers, suppliers and other stakeholders (Power, 2004a).

Table 2-2: Taxonomy of DSS (Power, 2000)

DSS' characteristics	Information representation
Users	Actual decision makers Some intermediaries
Focus	On data On model On communication
Scope	Organization Group Individual
Domain oriented	Financial Health care Airlines Cargo
Information Representation	File Drawer Systems Data Analysis Systems Analysis Information Systems
Models	Accounting and Financial models Representational models Optimization models Suggestion models

There is a more general classification of DSS, which is related to the number of users that make use of the system, as well as the scope reached by the system and the organizational impact it has. This classification proposes three main groups of DSS, the classical DSS, Group Decision Support System (GDSS) and Organization Decision Support System (ODSS), respectively discussed in sections 2.3.1.1 and 2.3.1.2.

The defition we chose to use in this thesis is based on one of Power's definitions (Power, 2004a):

➲ **DSS are a specific class of computarized information system that supports business and organizational decision-making activities.**

This definition is broad enough to encompass the supports on activities, that match the scenario of projects and meetings, which are part of our research scope.

2.3.1.1 GDSS – Group Decision Support Systems

Group Decision Support Systems are interactive, computer-based systems that facilitate solution of unstructured problems by a set of decision-makers working together as a group. It aids groups, especially groups of managers, in analyzing problem situations and in performing group decision making tasks.

DeSanctis and Gallupe (DeSanctis & Gallupe, 1987) describes GDSS as a combination of communication, computing, and decision support technologies to facilitate formulation and solution of unstructured problems by a group of people. Bui and Jarke (Bui & Jarke, 1986) define GDSS as systems which integrate traditional DSS features, e.g. analytical tools and focus on issues more than on persons, together with computer conferencing, email, etc—i.e. communication support elements. GDSS includes decision models like rating or brainstorming and support for communication, collaboration, and coordination (Power, 1999).

A GDSS aims at improving the process of group decision making by removing common communication barriers, providing techniques for structuring decision analysis, and systematically directing the pattern, timing, or content of discussion. The more sophisticated the GDSS technology, the more dramatic is the intervention into group's natural (unsupported) decision process. Of course, more dramatic intervention is not necessarily better for the group (DeSanctis & Gallupe, 1987).

In particular, the use of GDSS alters the nature of participation within the group, which, in turn, impacts decision quality and other outcomes of a decision practices. There is even substantial variation in the development stages experienced by successful and unsuccessful problem-solving groups. The only difference between these two extreme groups is that successful groups tend to devote adequate time to problem formulation and planning of decision practices strategy, whereas unsuccessful groups tend to immediately begin to search for alternative solutions (DeSanctis & Gallupe, 1987).

The taxonomy provided in (DeSanctis & Gallupe, 1987) divides GDSS into three groups:

- **Level 1 GDSS** – provide technical features aimed at removing common communication barriers, such as large screens for instantaneous display of ideas and voting. These systems improve the decision process by facilitating information exchange among members.
- **Level 2 GDSS** – provide decision modeling and group decision techniques aimed at reducing uncertainty and "noise" that occur in the group's decision process. Level 2 thus represents an enhanced GDSS, as opposed to Level 1.
- **Level 3 GDSS** – are characterized by machine-induced group communication patterns and can include expert advice in the selecting and arranging of rules to be applied during a meeting.

DeSanctis and Gallupe also comment that GDSS should also be classified under the dimensions of group size, member proximity, and the task confronting the group. The design and its impacts will be somewhat different when group members are remote from one another rather than meeting face-to-face.

The "group" dimension of DSS is the most complex dimension of this type of system, since the design of such system is complex as groupware users are different in background and roles and as a consequence, it is challenging to design an interface that can meet all users' requirements. Apart from that group dynamics is difficult to understand and draw general conclusions, group processes are often uneven and context sensitive and disclose over a relatively longer time frame than the individual activities, group behavior cannot be generalized to other groups, and groups' behaviour is highly influenced by the observation conditions, as cited in (Basoglu et al., 2001).

2.3.1.2 ODSS – Organization Decision Support Systems

Organization Decision Support Systems (ODSS) are large decision aiding systems, which provide organization-wide support for business processes. An ODSS shares some characteristics with other management support systems, but it has distinctly different objectives, scope or components (Santhanam et al., 2000). ODSS are multiparticipant DSS designed to support a decision maker in a setting that has a more elaborate infrastructure than a group (i.e. involving specialized roles, restricted communication patterns, differing authority levels).

According to (Santhanam et al., 2000) the key features of an ODSS are:

- It focuses on an organizational task or decision that affects several organizational units.
- It cuts across organizational functions and hierarchical layers.
- It involves computer-based technologies and communication technologies.
- It provides multiple users easy access to data and models.

The difference between ODSS and individual DSS, GDSS and EIS is their goals and scope. An ODSS has to support interrelated but autonomous decisions. Their development requires a lot of effort particularly in coordinating the requirements of several units. GDSS focus on single work groups, while ODSS focus on multiple work teams. EIS and ODSS provide critical information to managers, but the objective and scope of these systems are very different, and they cannot be viewed as being the same.

2.3.1.3 Summary of DSS

As shown in Table 2-1, a recent definition of DSS defines them as a broad category of systems that includes Data-Driven DSS, Model-Driven DSS, Communication-Driven DSS, Document-Driven DSS, and Knowledge-Driven DSS (Power 2004a).

However other business related words are used, sometimes, as synonyms of DSS. For example: Data Warehousing, Business Intelligence and EIS. We consider the definition of (Power, 2004a) more comprehensive and therefore we use it within this document. For example, according to his definition, Data-Driven DSS with OLAP provide functionality and decision support that is linked to analysis of large collections of historical data. These systems use quantitative and statistical tools for ordering, summarizing and evaluating the specific contents of a subject-oriented data warehouse. Still according to this definition, in order to make a clear distinction between some of these concepts, Document-Driven DSS integrate a variety of storage and processing technologies to provide complete document retrieval and analysis. A search engine, according to this author, is a powerful decision-aiding tool associated with a Document-Driven DSS. These systems use natural language and statistical tools for extracting, categorizing, indexing and summarizing subject-oriented documents warehouse. While Document-Driven DSS help managers process “soft” or qualitative information, Data-Driven DSS help managers process “hard” or numeric data.

Besides the traditional areas related to the development of DSS, currently, a number of disciplines provide substantive foundations for DSS development and research.

These are the following areas and their respective main contributions: Database (tools and management of data), Management Science (mathematical models) and Cognitive Science (design). Some other areas are also related to DSS in a minor representation, which are: Artificial Intelligence, Human-Computer Interaction, Simulation Methods, Software Engineering and Telecommunications (Power, 1999).

It is interesting to observe the evolution of these systems and why up to now they are still not solving many of our problems. One simple, but valid answer is, because the problems have evolved and became more complex. The types of decisions we used to have in the past are still present nowadays, whereas new types of decisions appeared and brought more complexity to be solved. When one thinks about inventing a DSS, one should consider that s/he is going to face not only all the technical problems the former DSS developers had, but also the current complexity of our scenarios. These scenarios demand more group interaction, more support on tools already available, and not new tools necessarily. Or support to areas rarely touched before. We discuss one of them in the following section.

In (Shim et al. 2002) the authors suggest some reflections regarding DSS research by the year 2007, with the support of new technology. For them, DSS researchers and developers should: (i) identify areas where tools are needed to transform uncertain and incomplete data, along with qualitative insights, into useful knowledge; (ii) be more prescriptive about effective decision making by using intelligent systems and methods; (iii) exploit advancing software tools to improve the productivity of working and decision making time, and (iv) assist and guide DSS practitioners in improving their core knowledge of effective decision support. This process will be enhanced by continued developments in Web-enabled tools, wireless protocols, and group support systems, which will expand the interactivity and pervasiveness of decision support technologies.

Apart from this DSS vision, until now, only a few systems support the phase that comes after a decision is made, the decision implementation and consequent follow-up. The next section discusses the concepts involved in this phase and the needed support for automatic tools.

2.4 Decision follow up

Decision follow-up is an area seldomly discussed in the literature. In this work, *decision follow-up* takes place after a decision is made. This concept is also known as post-decision (Svenson & Benthorn, 1992), consolidation (Svenson, 1992) or enactment of decisions (Balasubramanian et al., 1999), and it is related to the decision implementation phase.

Many researchers are interested in what happens during the decision making process, but what happens after the final decision and its consequent steps? How are the steps related? How can they contribute to improve organizational processes or to help in new decisions? The gap between the moment a decision is made and its corresponding implementation may, in fact, turn several decisions inconsequent, due to the lack of appropriate support to the implementation follow-up (Borges, Pino and Valle, 2002). As pointed by Russo (Russo, 1989), it is not enough to simply make a decision and move on. We must periodically review our decisions and consider that if we fail to track the results of our decisions, and fail to analyze them to reveal key lessons, we are wasting good opportunities for improvement. Also, from the perspective of Hammond (Hammond et al., 1999) a decision is not made in a vacuum, most of them are linked to those that proceed and succeed it.

For Svenson (Svenson & Benthorn, 1992) **the follow-up of this phase is fundamental to the evaluations of the quality of past decisions and as background knowledge for present and future decisions**. The same author justifies his beliefs in another work (Svenson, 1992) stating that decisions should be made to withstand the roughness of the future. In particular, post-decision external events and outcomes, and internal changes in the decision maker himself or herself, could suggest that another decision would have been better. Therefore, post-decision can be seen to serve a similar purpose in bringing the appraisal of an earlier decision more in line with the perception of the decision as reflecting their current attitudes (Conway and Ross, 1984 cited by Svenson 1992).

We consider that capturing and organizing the information and knowledge involved during the decision implementation, and make them available can contribute to avoid past mistakes and to help find out best organizational practices. We base this idea on previous research (Stewart, 1997), (O'Leary, 1998), (Markulla, 1999) and (Tiwana & Ramesh, 2001), that summarized can be point out as the three following assertions:

- Many mistakes are frequent in organizations and some of them are related to the lack of information and knowledge about previous experience;
- Other mistakes are frequent because of the lack of planning and control of decision implementation;
- Knowledge and information are generated during a decision implementation;

From our perspective, technology can help us improve these three assertions with the following ideas:

- Formalizing a decision implementation, using management systems, allows planning, tracking and getting warning signals about problems, providing control over a decision implementation;
- The use of technology in organizing, storing and making available information and knowledge generated during a decision implementation can contribute to improve the quality of future decisions;

On the other hand, it is difficult to formalize and follow decisions, since they can be made in different moments in organizations, such as in meetings or during informal communications. Thus, computer based tools are limited to support the activities that follow a decision made with their respective meaning and contexts. We do not deny that all tools used in organizations nowadays somehow support what we name "decision follow-up". Examples of these tools are document management tools, project management tools, workflow management systems, databases, CASEs, communication tools and knowledge management tools. However, the data provided by these systems are not related to the decisions made. One of the main reasons for that is the lack of formal documentation about decisions made.

If a manager wants to know what is the current status of a decision made and its implementation, he will probably have to search in different sources or to communicate with several people to get the needed information. And in many cases, the time or the accesses to the sources are constraints for this activity, for example, because of the lack of resources, or because the context is not applied anymore. And the activities embedded in this phase can vary from writing a document, organizing a trip, developing software code, having an informal communication, asking for advice from colleagues or constructing a house.

➲ **Decision follow-up considers everything that happens between the action of making a decision and its full implementation or the explicit definition of its end for any reason**.

For these reasons, we consider this phase a *wicked* phase, making an analogy with the concept cited in (Conklin, 2005) about wicked problems. For Rittel cited by (Conklin 2005) *wicked* problems are problems that:

- cannot be understood until you have developed a solution,
- have no stopping rule,
- do not have a right or wrong solutions, simply better, worse, good enough, or not good enough,
- are unique and novel,
- have solutions as a “one-shot operation”, and
- have no given alternative solutions.

As Courtney suggests (Courtney, 2001), with the globalization process it seems the organizations are leaded to deal with wicked planning problems. Methods are desperately needed to help with making effective decision in such situations. This author states that spreadsheets and other models and knowledge-based DSS, and especially GDSS can help with such problems. But it seems that more powerful tools are required. Especially required is a broader perspective in terms of DSS research.

The characterization of decision follow-up as a wicked problem shows us the complexity and the unstructured nature of analysing related scenario of decision follow-up, so that we can be able, or not, to propose coherent solutions for supporting this phase.

The next section discusses some perspectives and elements considered fundamental for the understanding of decision follow-up so that we can think about feasible solutions.

2.4.1 Elements of decision follow up

Traditionally, DSS are systems that support managers in the decision analysis phase, i.e. the time interval when decision makers are selecting a course of action from among multiple alternatives, and aim at **assisting and documenting the decision rationale** (e.g. the list of alternatives suggested and discarded, who made the decision, when, the tradeoffs evaluated, the argumentation that led to the decision, and the justifications for the choices made).

Our approach has a different focus, which starts with a decision made. We want to discuss how to support the activities that take place after a decision is made—i.e.

during the decision implementation—in order to obtain follow-ups. Nevertheless, part of the information discussed in traditional DSS is also relevant in our scope. In order to be able to obtain meaningful decision follow-ups, we need to represent, not only the decision itself, but a set of related elements and properties. Among them are the following ones:

- **The decision representation**
- **The decision type**
- **The organizational level of impact**
- **The decision maker(s)**
- **The location**
- **The time**
- **The related metaphors**
- **The related knowledge during the decision implementation**

2.4.1.1 Decision representation and context

The first element to be represented in a decision follow-up is a **decision**, which can be represented by a sentence in the imperative form, e.g. to buy a car model 2002, or in a negative form, e.g. not to travel outside the country for 3 months, it can be represented by a schema, a graphic, a model, or a combination of them.

➡	**A decision is a choice of one from among a number of alternatives; a statement indicating a commitment to a specific course of action, typically made to achieve a goal, to solve a problem, or to implement a plan**

This is the definition we are using in this work, which was extracted from the Free Glossary of DSS (Power, 2004b), complemented with the definition of decision from (Balasubramanian et al., 1999).

Besides documenting the decision itself, **it might be as relevant to document the alternatives** generated during the decision rationale, and why they were discarded or chosen, as indicated in the traditional DSS definitions. This information can be very relevant in the scenario of decision follow-up as well, when there is a need to trace back the origin and the path a decision has.

The **granularity** of a decision is also an interesting aspect to be analyzed. Imagine the example of building a house. A representation of the final decision could be "to build a house like a physical model representation". But embedded in this model, there are several smaller decisions represented, about the style of the architecture, about the number of rooms in the house, how they are organized, the colors they have to be painted, and so on. Thinking about a solution to support a decision follow-up can be related to all or only to part of these different granularities. The choice of representing "macro" decisions can bring, on one hand, fewer details, but simplicity. On the other hand, the representation of "micro" decisions can bring more details, but more complexity and overload of information.

The **decision type** is also an aspect that has an impact on the decision follow-up. If a decision is structured, the follow-up can be predicted. Otherwise, the lack of structure brings unpredictability to the scenario and different possibilities can be used to implement that decision.

The representation of decisions themselves in a follow-up support is not enough. **The context where a decision is made is as important as its representation**. By context in this work we follow the definition of the Merriam-Webster dictionary (Webster, 2005), which defines context as: *the interrelated conditions in which something exists or occurs*. In our decision follow-up scenario context includes elements such as the process the decision is part of, the areas of the organization that are affected by it, the responsible parts for it, the time and location where the decision is made, and finally its consequences.

Thus, the **organization level** where the decision has more impact should also be considered. Decision follow-ups differ on strategic, management and operational levels. For instance a strategic decision can have a direct impact in how the management control and the operational control levels work. They can demand that sub-decisions are made in the other two levels. While a decision that is related only to the operational level, might demand only some checkpoints to see how things are done.

The documentation of who the **responsible or decision maker** is might be as important as the determination of the decision itself. In the traditional definition of DSS, it is important to document **who**, **in the decision analysis phase**, **was in favor or against an alternative and the reasons why**. Besides, there are organizations and groups where one person is mostly responsible for making decisions, but there are also other cases where groups make decision, and therefore the responsibility is shared among various invididuals.

An important aspect to determine how to support decision follow-ups is the recognition of places where decisions are made, i.e. the **location**. Actually, decisions are possible to be made everywhere. It is impossible to provide support for all different situations, though. But there are some common practices and locations where decisions are more frequently made. Meeting is one example. In the next chapters we will discuss it in more details. But actually, nowadays, because of the nature of globalization and distributed teams, decisions are also made via email, via telephone, or using one of the communication technologies available, for example, in web-based DSS.

The **time** when the decision was made, as well as the time when the support is required to be available represent another dimension of the decision making processes that has changed over the years, mainly concerning the technological viewpoint. Again, the nature of distributed organizations makes the time being a combination of 24 hours a day, 7 days a week, i.e., there is no break necessarily. And so should be the system to support decision follow-ups. Or at least, the mechanisms should monitor user activities and decision tracks considering this time aspect.

Another perspective related to the decision follow-up is how the organization works. What **kind of metaphor** leads the way the organization works? Are people organized in **groups**, **departments**, and **projects**? The activities of implementing a decision are seen as tasks, processes or simple actions without formal connections? Or in a more complex scenario, we could have even a combination of them.

These differences can bring a demand for different follow-ups as well. For an organization that follows the group models, groupware can be the main tools to support the decision follow-ups. While for organizations based on project the use of project management tools might be more adequate to support decision follow-ups.

2.4.1.2 Knowledge

Another aspect of the decision follow-up is its intrinsic relation to knowledge. Actually, during the implementation of an activity, related to a decision, knowledge is needed and generated in several ways.

A classical understanding of what knowledge is maps this concept to the concept of data and information. Data are raw facts or simple observations about the state of the world; information is data in some context, or with some kind of human interpretation applied; and knowledge is information with guidance for action, that is, knowing how to act given the information (Courtney, 2001).

There are several knowledge taxonomies proposed in the related literature, which include the division of procedural vs. declarative, esoteric vs. exoteric, shallow vs. deep, and explicit vs. tacit (Courtney, 2001). **For the problem described in this work, decision follow-up, we are looking at the more traditional division: tacit and explicit knowledge** (Nonaka & Takeuchi, 1995).

2.4.1.2.1 Tacit Knowlegde

Tacit knowledge is highly personal and hard to formalize, making it difficult to share with others. It is embedded in an individual experience and it involves intangible factors such as personal belief, perspective and the value system. Subjective insights, intuitions, and hunches fall into this category of knowledge (Nonaka & Takeuchi, 1995). Tacit knowledge is tricky. And the proof for this interpretation is visible when we try to explain or to represent something we know. Sometimes the difficulty is related to the lack of capacity of recognizing that we know something (for e.g. an innate skill), or because of a lack of vocabulary to represent the knowledge.

We think that great part of the knowledge involved during decision implementation is tacit. Participants in decision implementation face daily problems and solve them using their tacit knowledge. Choices are made, options are discarded and this knowledge normally is not represented through any technology. Great part of the knowledge involved in decision implementation is lost or not available. But where is this knowledge? Definitely, it is inside the mind of the participants of the processes.

In the context of decision implementation, tacit knowledge is composed by the experiences and skills participants can make explicit and that can contribute to future decisions to be made. Experiences about successful or wrong choices are normally not represented in organizational processes, but they are part of the expertise acquired by the participants during processes implementation. But how can organizations make this knowledge explicit and available to others?

Storytelling is one mechanism pointed as a possibility of externalizing tacit knowledge (Valle et al., 2003), (Valle et al., 2002), (Shell, 2001), (Ruggles, 2002). A story can be defined as “a narration of a chain of events told or written in prose or verse". And "narration" comes from the Latin "narrere", meaning "to pass on knowledge". Thus, story is one possible mechanism to transmit knowledge.

Storytelling is as old as the history of man. The Egyptians registered stories through pictures. Later with the invention of writing, stories were registered by several civilizations. Indians still have the oral storytelling as the main way of propagating knowledge inside tribes through different generations. Stories can be used for different purposes, like teaching, entertainment and knowledge transferring; they can have different scripts, linear ones or a broken alternative order; they can cause different emotions like sadness, happiness, excitement and fear. Thus, storytelling is not an optional extra, but an old skill in a new context. The new context is the emergence of Knowledge Management (Shell, 2001).

Stories are a very powerful way to represent complex, multi-dimensional concepts. While a certain amount of knowledge can be reflected as information, stories hold the key to unlocking the vital knowledge, which remains beyond the reach of easily codified information (Ruggles, 2002). Still reinforcing this idea, the same author suggests that stories are great vehicles for wrapping together the many elements of knowledge. A good story combines the explicit with the tacit, the information with the emotion.

Based on this concept, knowledge and story are inextricably connected, and stories are partial, structured memories of observed and articulated reality, aims Roger Schank and Robert Abelson, cited by Davenport and Prusak in (Davenport & Prusak, 1998).

Stories can be created and transmitted by one person or by a group of people. Group storytelling means that more than one person is contributing to create a story, synchronously or not, at the same place or no. We think that group storytelling is more adequate than single storytelling when applied to the scenario of decision follow-up. Since within this context many people are involved in implementing tasks and also, the knowledge generated in the end is the combination of the skills acquired by each participant during the processes execution. We think the group story can contain many more valuable details if more than one person participates in its creation, since an experience about a decision implementation phase normally involves more than one individual.

The choice of defining what is a good story might be subjective and dependent on the participants involved, but some examples of executed tasks and experiences that are candidates of tacit knowledge, which can be transformed into explicit knowledge, are (Valle et al, 2002):

- Procedure reports: how predicted steps were developed in reality, what were the problems encountered and how they were solved.
- Innovative situations: what were the innovative situations and what did they bring to the expertise of the organization or to the groups involved.
- Interdependence between involved groups and processes: justification why certain people were involved in tasks, somehow indicating who the expert in each subject is.
- Difficulties: what were the barriers to execute certain tasks or processes?

- Results: what was the end of the processes, with description of the reached results?

These examples could be part of the experience a senior employee has in an organization. However, there is no formal mechanism to put this knowledge available, except by the normal procedure of a conversation, i.e., asking and listening to reports of such experiences.

2.4.1.2.2 Explicit Knowledge

Explicit knowledge can be expressed in words and numbers, and easily communicate and shared in the form of hard data, scientific formulae, codified procedures or universal principles. It is the type of knowledge that can be readily articulated, written down, codified and shared (Courtney, 2001).

In the context of decision follow-up, explicit knowledge can be expressed by several elements: documents, reports, processes, tasks, presentations, spreadsheet files are examples of those.

Actually, **the support for explicit knowledge in the context of decision follow-up has being addressed by many tools like project, document and knowledge management tools or a combination of them**. However, the data provided by these systems are not explicitly related to the decisions made.

2.4.2 Summary of decision follow up

We could observe along the last sections that decision follow-up is a complex, but interesting topic, which was not extensively explored in the literature. The next section will describe a list of DSS following the taxonomy of (Power 2004a).

2.5 State of the Art of DSS Related Systems

In this section we discuss DSS, of various types, which are products and research initiatives to be used in different phases of the decision making process. We do not aim at providing a thorough state of the art, since this would be an overwhelming task, but to give an overall perspective about the current technologies, describing systems which are classified in the market or in the relevant literature as DSS or DSS related.

The systems listed below are classified as DSS according to a taxonomy proposed by Power in (Power, 2004a), previously presented in this chapter (**c.f. Table 2-1**). However, some of them belong to more than one category in Power’s taxonomy. One of the most difficult distinctions or, positively speaking, overlap between the taxonomy’s definitions, occurs among communication, document and knowledge driven DSS. Their intersection can be observed in various systems, such as those classified as “knowledge management systems” that also support document management and electronic communication. One concrete example is Microsoft SharePoint Portal Server, which, according to Power’s taxonomy, can be seen as a combination of document-, knowledge- and communication-driven DSS.

2.5.1 Lotus Solutions

Lotus (IBM, 2005) is the provider of one of the most popular groupware, Lotus Notes. Nowadays the company has a set of tools related to KM, document management, learning, which can support the decision making process in different ways.

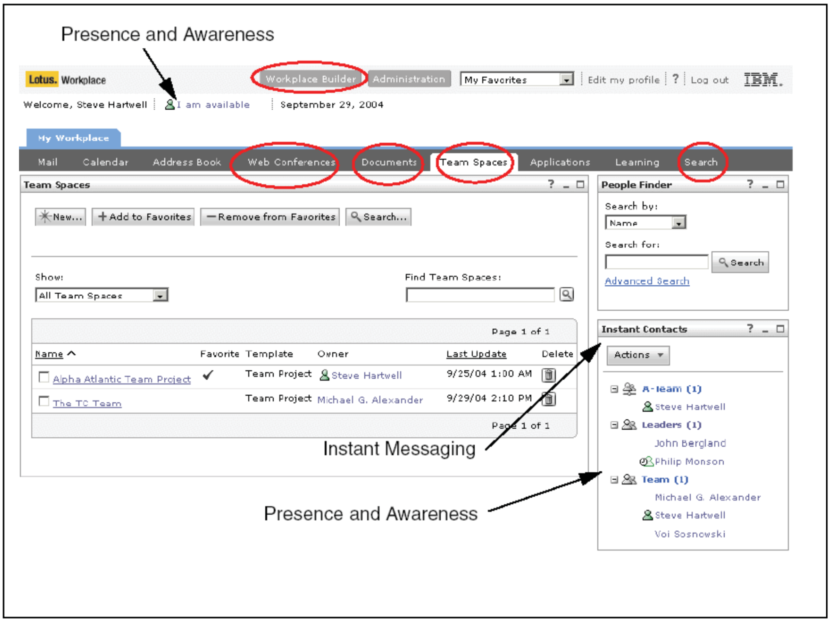


Figure 2-3: Lotus Workplace Team Collaboration Components (IBM, 2005)

Lotus Workplace is the front-end of computing that enables users to aggregate data, applications, business processes, and people into a single work environment that can be accessed from a Web browser, mobile device, or server-managed client. Workplace increases productivity by providing a single workspace for users to find information. It combines new and existing IBM products and technologies, including, Workplace itself, Lotus Notes and Domino, WebSphere Portal, and WebSphere Everyplace.

These are some of the most relevant tools, which are part of the Workplace solution:

- **Lotus Workplace Team Collaboration:** a tool that provides support for online meetings, create libraries, and interact with team members through online chats, threaded discussion forums, and document sharing.

Lotus Discovery Server: is a comprehensive knowledge server for e-business users, which provides search or browse for information and subject matter experts from multiple locations. Users can collaborate with colleagues instantly, increasing the number of opportunities for knowledge sharing and decreasing the time spent looking for needed resources. All these features can be executed in a web browser.

The Lotus solutions briefly described, contribute either alone or in combination with other tools to support the decision making process, since they improve the access to information, to knowledge, they facilitate the access of related topics, from diverse data sources. However, these tools do not address the specific problem of decision follow-up.

2.5.2 Microsoft SharePoint Portal Server

Microsoft SharePoint (Microsoft, 2005) is a server system that provides: discussions, notifications, document management (versioning, routing, publishing, check-in, check-out), and search capabilities, which are based on multiple servers and data types. The following modules compose the server:

Document Management: check-in and check-out ensures that one user can reserve documents for work at a time. Versioning track changes with different version numbers for auditing and rollback purposes. Profiling provides both optional and required metadata. It is fully integrated with Microsoft Office and Windows. Collaboration through the discussions' feature in Office and HTML documents content review. Security features ensure that only users with appropriate access can view and modify a document.

Search Services: crawls content or set of URLs for inclusion in an index, parse or filter documents to extract relevant metadata and content, and perform content searches.

Portal Framework: presents content in a customizable way, to suit the needs and preferences of a user or a group. It allows addition of Web Parts from many other sources.

Web Parts: are customizable, reusable components used to display specific information on a Web page. Web Parts can plug into existing back-end systems, to present customized views to a desktop. Then, from a single interface, users can access internal data, external feeds, and collaboration tools.

The various possibilities regarding integration is one of the biggest advantages of MS SharePoint.

SharePoint is one of the few systems we have analysed which explicitly represents "decisions". Decisions are created in "Meeting Workspaces", i.e. a special folder, which is dedicated to support and document meetings. Besides events calendar with the meeting date, time and description, SharePoint supports the definitions of Agenda (Figure 2-4), Tasks and Decisions (Figure 2-5). The problem of SharePoint in this scenario is similar to the problem of Groove (see below), the lack of relation between tasks and decisions and no possibilities of direct view on follow-ups.

Figure 2-4: Meeting agenda in SharePoint (Microsoft, 2005)

Figure 2-5: Decision representation in SharePoint (Microsoft, 2005)

2.5.3 Groove

Groove is a system that allows teams of people to work together over a network as if they were in the same physical location (Groove, 2005). It supports documents, projects, meetings and data management.

Regarding **document and data management**, Groove allows for creating workspaces, sharing and revising documents, and synchronizing information between different computers in various locations (e.g. at work or at home). Additionally, this module also offers possibilities of synchronous communication support, allowing users to see who is online, receive notification of changes, chat directly within shared folders, and work offline.

The **meeting management** in Groove is quite comprehensive, allowing for: sharing documents, carrying on conversations, assigning action items, working offline, conducting virtual meetings, and keeping track of schedules, tasks and milestones. This tool helps structure, online and offline meeting management. It can be used before meetings to organize agenda and participants, during meetings to conduct and record, and after meetings to track outcomes.

Besides these big modules, Groove contains a set of tools for **general utilities** which can be very useful in Project management and daily work. Among them are calendar, list of contacts, discussion forums, task manager and list of web links. And additionally, users can also make use of more specific tools for instance ActiCam (a conference tool), FlexiVote (a team voting tool), Task Tracker (simple task management tool), TeamWriter (a collaborative writing/editing tool), Veridian CIM (a structured argument & analysis tool), etc.

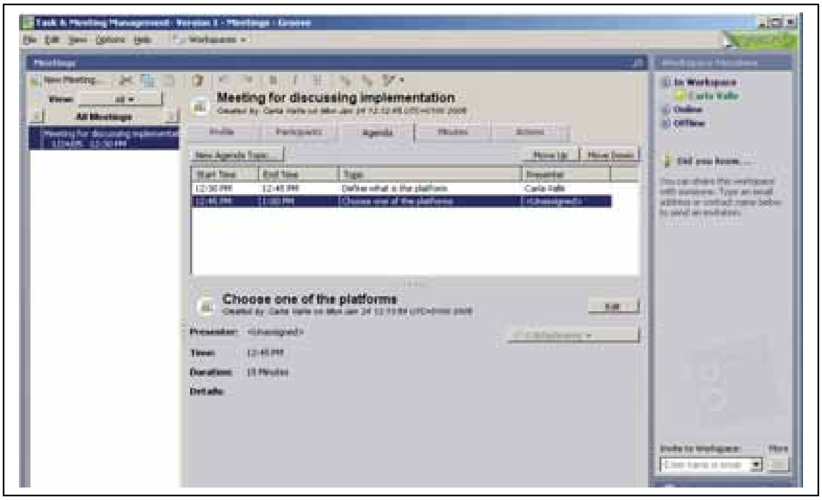


Figure 2-6: Groove agenda definition (Groove, 2005)

The Veridian tool deserves some more details, since it is a tool for helping in the process of decision-making. Veridian CIM is a structured argument tool that improves analytic flexibility; it provides transparency and increases depth of analysis. There are two approaches to using CIM, as a strategic options analysis and priorities tool, and/or to try and figure out competitor plans and intent. CIM is a modeling tool that can be used for representing and analyzing situations for scenario planning, production processes, information security, strategic decision making and economic analysis.

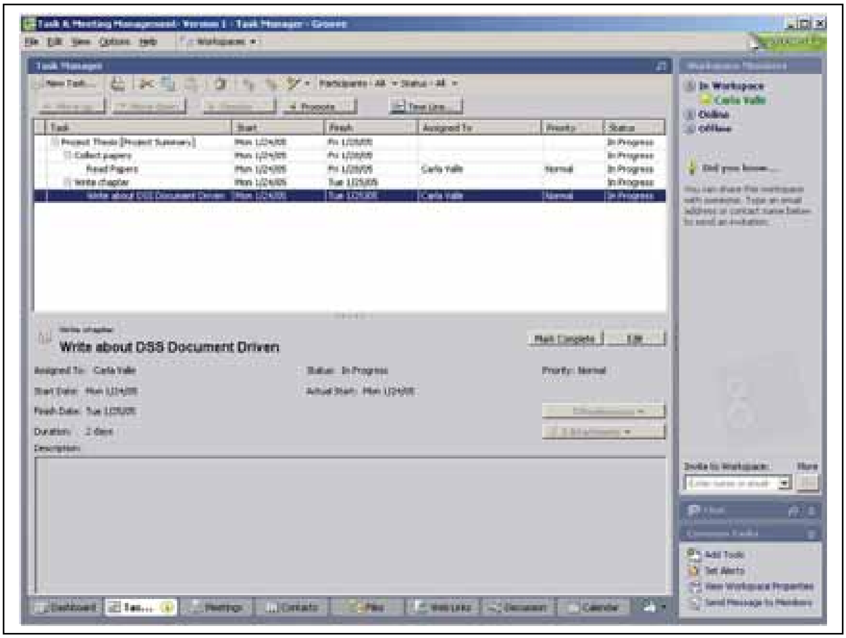


Figure 2-7: Task Management in Groove (Groove, 2005)

Groove seems to be one of the closest systems related to this thesis, since it combines features of document management system, communication support and meeting support. The lack of support for follow-ups, in detailed ways (e.g. with visualizations, searches) and the lack of distinction between tasks and decisions are two features that are not present in Groove.

2.5.4 EXSYS

EXSYS (Exsys, 2005) is cited in the recent book of Power (Power, 2004a) as one the most used DSS tools nowadays. Also known as “Exsys CORVID” the system assists interactive decision-making expertise to Web sites. It allows the logical rules and procedural steps used by an expert to make a decision, to be efficiently stated in a form that is easy to read, understand and maintain.

In an interactive session delivered over the Web, the underlying Inference Engine uses the problem-solving logic to emulate the questions, process, and recommendations of domain experts. People interact with the system as if they were talking to an expert. The decision-making logic is stated in If/Then rules, in much the same way as explaining to another person how to make the decision. The rules are written in English and algebra, making them easy to read, understand and maintain. The system provides tree structures to organize related rules, and logic blocks to organize related trees.

People interact with the system by answering questions. Based on their input and on the system's logic, the system will ask more questions - drilling down where necessary and skipping questions that are not needed. The system may only ask a few questions, or many, depending on the complexity of the problem. Once the system has fully analyzed the user's input, it will provide its conclusions and recommendations.

The set of tools can be integrated to other systems, as Java Applets running on clients, or the Java servlet approach running on server-side. The system is also extensible, meaning that advanced users can develop their own functionality and include them in their own EXSYS solution via API.

This system does not deal with the specific problem of decision follow up, but with the earlier phases of the decision making process.

2.5.5 Autonomy

Autonomy (Autonomy, 2005) is a knowledge management tool that offers high degrees of accuracy and sophistication, using a scalable technology that recognizes concepts—the main differentiator of this tool —which provides powerful conceptual retrieval features. Autonomy is composed of the following tools:

Concerning search possibilities, Autonomy offers various options:

- **Natural Language Retrieval:** which allows the search to be triggered with a piece of content (e.g., a sentence, paragraph or page of text, or the derived contextual information of an audio or speech snippet), which work as input and the system returns references to conceptually related documents ranked by relevance, or contextual distance. This is used to generate automatic hyperlinks between pieces of content.

- **Query by Example:** which is based on sample data such as a paragraph, a page of text, an entire document, the body of an e-mail or even the derived contextual information of an audio or speech snippet, which work as input and the system returns references to conceptually related documents ranked by relevance, or contextual distance. This is also used to generate automatic hyperlinks between pieces of content.

- **Refine by Example:** which is based on the results of the natural language retrieval and users can quickly refine their searches to precisely focus on the context they require.

- **Cross-Language Search:** the system delivers a language independent software infrastructure that enables content to be conceptually retrieved in any language, with multilingual and cross-lingual capabilities.

Concerning data management, this system is rather complete as well. It supports document management repositories, relational database management systems, local file systems, Internet servers, E-mail servers, and news servers. Additionally, the system handles multimedia content, such as videos, broadcasts, audio archives, and news feed streams, and transcribes audio content into text, identifies and ranks the main concepts within it, and automatically personalizes and delivers that information to those who need it any way they want - across the Internet, the extended enterprise or using other digital channels, such as mobile phones or PDAs.

Autonomy is a KM tool currently available in the market. It is a comprehensive and modular tool. Some other KM tools are based on Autonomy's search engine or on its concept's tool. From the perspective of decision follow-up, there is nothing especially designed for that purpose in Autonomy. On the other hand, we cannot deny that the combination of the good features provided by Autonomy (e.g. community support, search engine, agents, wireless support) definitely contribute to improve the decision making process in any phase.

2.5.6 Entopia

Entopia Quantum (Entopia, 2005) is a content management and collaborative system, which supports management and delivery of information for individuals, groups or divisions within an organization. It includes features of instant messenger, approval routing and offline synchronization.

Entopia enables individuals to collect information from various digital sources, allows workgroups to collaborate in project or team-based shared spaces, and support knowledge management through the location of documents, experts, and information sources.

By providing the end user with a choice of either a desktop or web-based tool, increasing the work opportunities, the system helps build a repository of corporate information assets to facilitate knowledge creation and improve business decision-making. The result is improved productivity in the form of shared practices, greater cost efficiencies, better customer relations, faster sales cycles, shortened product time-to-market, and better decision-making.

Entopia Quantum features a 3-tier capability architecture to collect, collaborate and capitalize on knowledge management. The quantum collect module helps users with the ongoing process of gathering, organizing and enriching information from various digital sources. The quantum collaborate module delivers a server infrastructure that enables users within the same organization to easily share information and collaborate with each other and their workgroups. And the quantum capitalize module offers advanced products and services that help enterprise users leverage the intellectual capital in actions that directly impact the bottom line. Quantum capitalize includes the knowledge locator, a tool for locating experts, documents and other sources of information.

One of the most interesting features of this product is the Entopia K-Bus, a tool for information discovery. K-Bus is a software infrastructure that captures the essence of the available content and employee interaction around it, to deliver unknown business insights through advanced information discovery mechanisms. The infrastructure makes use of business context around content to capture the real-time value of information to specific individuals and groups within the organization. This makes the discovery and delivery of information more precise, meaningful and valuable.

Entopia K-Bus is a scalable infrastructure that consists of components for: managing the communication between different applications (e.g. Emails and transactional applications), managing a centralized metadata repository, enabling different searches, expert discovery, content visualization (e.g. graphical map of the key concepts) and social network mapping, which identifies topic-based social networks to visually represent the flow of information across an organization.

Entopia also allows the use of a set of web service APIs, enabling users to create additional custom applications to meet the specific needs of distinct enterprises. A module that deserves more details is the “Entopia Content Visualization”. This tool support to represent and understand large sets of content. It semantically processes the content within a specified set of information to display a graphical representation of the key concepts – and the relationships between those concepts – found within the content. The resulting knowledge map – known as the Entopia K-Map – allows users to quickly understand the key ideas detailed within hundreds or thousands of pieces of information.

Entopia does not focus on the a direct support for decision follow-up, but the knowledge management features increase the possibilities for knowledge discovery, which somehow, is related to organization memory, i.e. one of the benefits of realizing any type of decision follow-up.

2.5.7 Workflow and decision follow-up

A previous work done by the author of this document proposes a related approach to this research (Borges, Pino and Valle, 2005). The authors discuss the use of a post-meeting support workflow-like solution where working plans can be described and enacted. The proposed solution is a system combining a process design tool with a workflow enactment tool. However, given the ad-hoc nature of the processes described in the paper, the authors suggest that a commercial Workflow Management System (WfMS) alone would not be enough to support the requirements of the scenario of post-meeting phases. Even with the adoption of a WfMS for process enactment, additional monitoring tools are suggested.

The use of a process pattern approach based on WFMS is proposed. In this approach, a number of process patterns is made available to the process designer, based on the characteristics of the domain, in this case, the post-meeting processes. A pattern is defined as a generalized description of a set of recurring rules that can be associated with a workflow schema. This motivation came from the idea that designers can reuse previous experiences to improve the speed and the quality of the schema design process (Casati et al., 2000).

In order to provide decision follow-up some steps should follow the execution of meetings: the design of an implementation plan, the analysis and improvement of the execution plan, discussions about the plan and its execution supported by a WfMS. A process plan defined in the system is associated with each decision. Unlike traditional business processes, the authors assumed that each plan would have only one instance. Therefore, changes to the plan can be done as needed during the process execution.

The system proposed is divided into four modules:

- A process pattern library, created from the initial analysis of meeting decisions;

- A process model tool, which uses the process pattern library and the past process definitions;
- A data driven WfMS which allows applications to access the process data; and
- A subscription mechanism to allow organization members to be kept informed of the decision implementation.

This approach makes the post-meeting phase an explicit phase in organizations, with clear possible benefits: there is a structured follow-up of the decisions made at the meeting, there is explicit awareness of task progress for all involved roles, there is Organizational Memory capture of potentially valuable information, and finally, there is easier identification of several items after the meeting, such as issues for the next cycle, unclear details of the decisions just made, and unsettled issues after some tasks have been done.

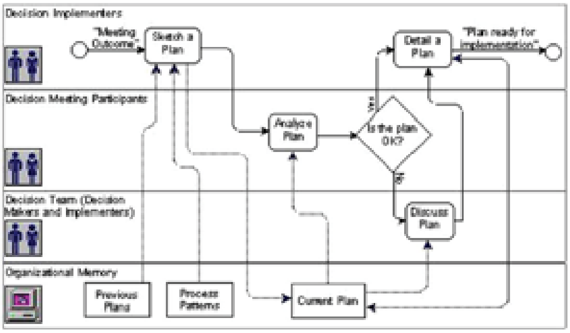


Figure 2-8: Design process of post-meeting implementation plan (Borges, Pino and Valle, 2005).

The main problem of this approach is the lack of integration with tools to support post-meeting, like the traditional ones, such as Project Management Systems (PMS), Document Management Tools, electronic mail, to-do lists and others.

2.5.8 BSCW

Basic Support for Cooperative Work (BSCW) is a web-based document management system (BSCW, 2005). BSCW allows the management of folders (personal, shared or public), documents of different types, users, urls and calendar (individual or group calendar). Some extensions of BSCW allow the use of project management features and light task management features, as well as workflow.

BSCW supports document version control, different levels of access rights, discussion forums, search engine and a light wap client for mobile users.

The system provides awareness mechanisms in two ways: a notification system, formatted as a daily activity report sent by email, which contains information related to workspaces, and meaningful icons, which shows new elements, changes, historical data, version data and shared elements.

Users sharing workspaces (folders) can organize their data the way it is more convenient for them, while the system maintains the integrity with the database objects.

The meeting support in BSCW is also very restricted. The Calendar module allows users to send invitations for meetings, defining location, date, a list of invited people and specifically for synchronous meetings, the technology that is going to be used. The invited people can accept or reject invitations, but the system does not support any special feature for agenda, minutes, tasks and decisions related to meetings.

BSCW is an asynchronous groupware systems, which reached 100.000 users registered only in the main public server in the year 2002. Many of these users use BSCW as a project management tool, as a shared workspace, to put information about decisions made and decisions to be made, but there is no explicit support feature to organize and access this data in a friendly and easy way to users. In other words, it is very difficult to trace any historical data about decisions made using the elements supported by the standard BSCW.

One initiative that was built on top of BSCW and which concerns knowledge management and, in this sense, concerns better understanding of projects, expertise, etc, was the MILK Project (Valle et al, 2003). Again, although this project did not focus on solving any concrete problem of decision follow up, it can be seen as one step forward to that direction. MILK offered a possibility of usage on the PC environment – focused on content creation, on the mobile environment – focused o awareness and on the social environment – focuse on content sharing. But alike other KM tools, MILK did not treat decisions as an explicit element.

2.5.9 Crystal Ball

Crystal Ball (Crystal, 2005) is a tool that supports risk analysis to support decision making processes. This tool provides several functions like: searches for optimal choice for a given business decision; incorporates advanced methods of time-series forecasting by analyzing historical data to predict future outcomes; automates simulations and forecasts in VBA; performs Monte Carlo analysis on spreadsheet models; and automates model-building tasks, simulates variability, defines correlations, among others.

Crystal Ball provides the following tools to support decision analysis:

- Batch Fit
- Correlation Matrix
- Tornado Chart
- Bootstrap
- Decision Table
- Scenario Analysis
- 2D Simulation

This system does not provide any feature specifically focused on decision follow-up, but it is one of the currently available tools used for graphic visualization in the support the early phases of the decision making process. With several simulation and graphic features, managers can analyse data in a better and more optimized way.

Tools like Crystal Ball seem to bring contributions to interpret data using different methods and representation.

2.5.10 Proclarity

Proclarity (Proclarity, 2005) provides a set of tools for decision making support.

The group of tools named “Analytics Family” enable organizations to centrally manage, store and deploy best practices and key performance indicators (KPIs). This is done by providing ad-hoc analytic capabilities, unique data visualization and navigation to users across the entire enterprise. The Analytics Family includes the ProClarity Dashboard Server, Business Logic Server and Analytics Server - designed to centralize information access, business rules and performance monitoring.

These are the main features of Proclarity modules:

- **Dashboard Server:** is a dynamic presentation layer for analytics that are rendered by the ProClarity Analytics Server. The Dashboard Server facilitates business performance monitoring and provides users with a timely snapshot of how they are performing against business objectives. The Dashboard Server, together with the Analytics Server and Business Logic Server, provides the framework for rapidly deploying highly manageable dashboards and scorecards.

- **Business Logic Server:** includes the client-side tools required to create, publish and manage business logic that is stored on the Analytics Server. It allows users to centrally store pre-defined analytic best practices, including business logic and rules; key performance indicators (KPIs); and member and set definitions. Analysts and business users can create, manage and store analytic logic to the Business Logic Server.

- **Analytics Server:** is at the foundation for all of the Analytics Family of products. This middle-tier server manages connections and queries to underlying data cubes; delivers analytics to users by rendering user interface clients; processes queries and centrally manages security administration, scaling and caching.

At the client side, Proclarity can be used with its own solutions, e.g. Dashboard, or a web-standard view. One of the great features of Proclarity is its capability to be integrated to other tool as the Business Intelligence support platform, e.g. with Microsoft SharePoint Portal Server or many others through its extensive API.

This is a DSS tool which helps in different phases of the decision making processes, and not in the follow-up. Scenarios that are suggested by the system’s vendor are: integration of analytic capabilities into existing operational applications and web-based applications and virtually any portal, integration of an organization’s best analytical practices and specific workflows into a guided analytic experience (e.g. a purpose-built business performance monitoring application), externally focused (internet or extranet) applications to help the organization gain value through aiding remote workers, customers or suppliers to understand key information.

2.5.11 Frontline Systems

Frontline (Solver, 2005) has a set of tools to help Excel users, or developers of DSS tools with a specific solver module, i.e. a spreadsheet optimization tool. The main idea of this tool it that optimizing a company model with the solver, you can discover ways to reduce costs, increase profits, or improve quality or response time.

The package named "Premium Solver Platform" contains diverse ways of dealing with multiple variables, up to 2000. Among them are:

- LP/Quadratic solver handles problems up to 2,000 variables
- GRG solver handles smooth nonlinear problems up to 500 variables
- Interval global solver handles global optimization problems up to 500 variables
- Hybrid evolutionary solver handles non-smooth problems up to 500 variables
- New SOCP barrier solver handles linear and conic problems up to 2,000 variables
- New cone constraints and all different constraints offering modeling power
- New model analyzer detects linear, quadratic, smooth, convex and nonsmooth formulas
- New model transformer converts IF, MIN, MAX, other functions to linear equivalents

Examples of usage of such a system are: crew scheduling, office assignment, employee scheduling, workforce composition (e.g. help to decide how many employees to retrain, hire and fire to meet changing workforce composition requirements while minimizing costs or employee turnover) and troops (e.g. help to decide how many troops to move from several camps to several other bases, to minimize movement time or total cost).

2.6 Conclusion

In this chapter we discussed the concepts involved in decision making processes, especially in the phase named decision implementation, looking at aspects of decision follow-up.

We saw that the theoretical decision making area is an old and broad research area that brought important contributions with the assistance of DSS. Systems with different features were proposed and solved part of the problems, but few of them looked at the aspect of decision follow-ups.

It was also explained that some aspects of decision follow-up are already addressed by various tools, nevertheless the support is still far from satisfactory.

We also listed various systems classified as DSS, looking at the five different perspectives proposed by Power (Power, 2004a).

We did not find anything specifically addressing the problem of decision follow-ups based on meetings, and related to asynchronous communication and tasks, but various initiatives deal with important parts of this scenario.

Table 2-3 shows how these systems represent and deal with decisions and where they concentrate their main support in the decision making process. The criteria used in this table were developed by the author of this document based on the literature cited in this chapter and in the next one, with emphasis on the aspects which are relevant to our interest, i.e. decision representation, decision implementation and follow-ups mechanisms. The idea behind this framework for categorizing the systems is to explicitly show where the foci of the systems are, and what they explicit offer to support the problems of the scope of this thesis.

The cells painted in gray, with a bold “x”, are drawing the reader’s attention to the most relevant points which concern this thesis. Thus, we can observe that SharePoint, Groove are the two systems that are closer to dealing with decision follow-up the way we discuss in this thesis. Nevertheless, both systems do not offer such support. Most of the KM and document management systems offer rich options for management, visualization and search that are related to the early phases of the decision making process, but none of them, except the previous work of this author (Borges, Pino e Valle, 2005), deals with the decision follow-up explicitly.

The next chapter will discuss the concepts and systems from the CSCW literature, with focus on meetingware.

Table 2-3: Comparative framework of DSS and decision support features

	Lotus	SharePoint	Groove	EXSYS	Autonomy	Entopia	WfMS related	BSCW	Crystal Ball	Proclarity	Frontline
Decision related support											
Decision Representation											
Explicit representation of decision		x					x				
Implicit representation of decision	x		x						x		x
Textual decision representation		x	x				x				
Graphic decision representation											
Location where decisions are made		x									
Decision Making Process Phases											
Problem Recognition	x	x	x	x	x	x	x	x	x	x	x
Problem Definition	x	x	x	x	x	x			x	x	x
Alternative Generation				x		x			x		x
Model Development			x	x					x	x	x
Alternative Analyses			x	x	x					x	x
Choice			x	x							
Implementation							x				
Metaphor Support											
Project representation	x	x	x			x	x	x			
Task or Action representation	x	x	x					x			
Decision Follow-up											
Provide means for decision follow-up							x				
Provide means for task follow-up			x								
Knowledge Management	x	x	x	x	x	x			x		
Advanced Visualization (e.g. Networks, 3D)					x	x			x	x	x
Search options	x	x	x	x	x	x	x	x	x	x	x
Web interface	x	x	x	x	x	x	x	x	x	x	x
Meeting support											
- agenda management		x	x								
- minutes management		x	x					x			
- post-meeting management											
Document Management support											
- allows documents	x	x	x		x	x		x			
- versioning control	x	x	x		x	x		x			
Support for multimedia	x	x	x		x			x			
Integration with other tools											
- integration with calendars	x	x	x					x			
- integration with to-do lists	x	x	x								
- integration with e-mails	x	x	x	x		x		x			
- integration with instant messenger	x	x	x								
- extensible or API available	x	x	x	x		x		x			
Group Support											
Awareness mechanisms	x	x	x		x		x	x			
Synchronous support	x	x	x								
Asynchronous support	x	x	x	x	x	x	x	x	x		
Support for Group or Org.Memory	x	x	x		x	x	x	x			

3 Meeting and related concepts

This chapter contains a review of the literature regarding: the CSCW research area and its relevant concepts, meeting support systems (meetingware) and related systems.

In this work, since we are interested in decisions made in meetings and how we can track them after the meetings are finished, we concentrate the list of related work on this class of systems. Once more, our idea is to provide the highlights of the state of the art in this field and not a complete list of available tools and research initiatives.

It is important to notice, though, that different from the DSS tools listed in the previous chapter, many of these meetingware are research initiatives. Only some of them became products and are still in use.

3.1 Meeting support systems and concepts

The first generation of computer technology was focused on improving the productivity of individuals and automating routine tasks. However, many problems organizations faced, and still face, were not routine. They could not be solved by the rugged individualist because no one person had all the experience, all the insight, all the information, or all the inspiration to accomplish such a task alone. And so teams form (Nunamaker et al., 1995).

At the beginning of the 80s, the field of **Computer Supported Collaborative Work** (CSCW) was created, as a multidisciplinary field that studies and looks for new technologies to communicate and process information. Its main goal is to achieve efficient communication, collaboration, and coordination of group activity (Ellis, et.al. 1991). CSCW started as an effort by technologists to learn from economists, social psychologists, anthropologists, organizational theorists, educators and anyone else who can shed light on group activity (Grudin, 1994).

Still in the CSCW community, another important concept is the concept of **Groupware,** which is a generic name and refers to specialized computer tools that are designed for the use of collaborative work groups. Groupware is any technology specifically used to make groups more productive. Typically, these groups are project-oriented teams that have to carry-out related tasks and share deadlines. Groupware can involve software, hardware, services and/or group process support (Johansen, 1988). Another definition of this term suggests that these are computer-based systems that support groups of people engaged in a common task (or goal) and that provide an interface to a shared environment (Ellis, et.al. 1991).

Examples of groupware are: communication tools (e.g. email, videoconferencing, chat) or coordination tools (e.g. project management systems, task management systems, meetingware, group calendar and workflows) or collaboration tools (e.g. electronic brainstorming and collaborative editors).

Another important concept created by the Information System and Management communities in the 80's, see (Bui & Jarke, 1986) and (De Sanctis & Gallupe, 1987), is of **Group Decision Support Systems** (GDSS). These are interactive, computer-based systems that facilitate solution of unstructured problems by a set of decision-makers working together as a group. It aids groups, especially groups of managers, in analyzing problem situations and in performing group decision making tasks. GDSS can also be seen as hybrids DSS, since most of them have various features which can be differently categorized in the framework of Power (Power, 2004a): data-driven, model-driven, knowledge-driven, document-driven and communication-driven DSS. Expanding the definition a little more, since GDSS are not only used in strictly "decision-making" processes, but also to support meetings, for example, GDSS can also be seen as one class of Groupware. To avoid misunderstandings between Information Systems and CSCW communities, another acronym was created by dropping off the "D" from GDSS, staying only "GSS" – Group Support Systems. But as one can observe in this work, these terms overlap.

One type of groupware or GDSS of our interest in this research is the **Electronic Meeting System** (EMS), which are systems to support meeting activities, either face-to-face, or distributed, synchronously or asynchronously. EMS, also known as "**Meetingware**" is a set of communication tools that allows the exchange of information among participants in a meeting (Rodriguez & Favela, 2001). This type of system aims at improving the possibilities of interaction with supporting commentaries, prioritization, voting, brainstorming, and documentation.

Meetingware were created, ideally, to be used in "electronic rooms", i.e. rooms equipped with a set of interconnected computers controlled by a central server, printers, and audio-visual supports as electronic boards, and video controllers (e.g., cameras and projectors), (Nunamaker et al., 1991).

A considerable amount of time in collaborative work groups is used in meetings; thus the importance of an EMS that reduces the time of meetings and improves the presentation and exchange of information is evident (Rodriguez & Favela, 2001).

Besides being difficult, meetings are expensive. For all their difficulty, meetings are still essential; for all their expense, they are not likely to go away. Meetings will remain absolutely essential and valuable, as they have been for thousands of years (Nunamaker et al., 1995). According to the same author meetings tend to have poor quality or to produce poor outcome, or even, give the sensation of waste of time because of a list of factors. Among them are: domination of individuals, fear of speaking, misunderstandings, lack of focus, premature decisions, missing of information, distractions, wrong participants, ignored alternatives, poor planning, hidden agendas, conflict, inadequate resources, poorly defined goals, groupthink, and lack of consensus, just to cite a few.

EMS are tools to improve productivity by reducing the attention costs for the communication, thought, and information access processes for groups engaged in concerted work. Participants use the technology to support both distributed and face-to-face work. EMS technology had its beginnings in the first computer-aided software engineering (CASE) project in the middle 1960s. This project, which won the Warnier Prize in Computer Science as the first instance of CASE technology, was the crucial point for the early work of Prof. Nunamaker and his team at the University of Arizona.

In the late 1970s they constructed the first EMS so that analysts could capture natural-language discussions of system requirements from many users simultaneously, so that all points of view could be considered, and users could negotiate their differences before information systems were built. The first prototype, called *Plexsys* was developed in the early 80s and was considered a success. Users could say what they needed to say, discuss ideas and identify agreements or disagreements that needed further discussion. With this system, users could reduce the meeting time considerably and organize better the requirements. With this experience, it turned out that actually the system could be useful for any group of people expending cognitive effort to achieve a goal, and not only for Software Engineering. The first improvements proposed for this system focused on the face-to-face situations, but later further research was done in the direction of supporting distributed meetings.

Still in the 80s, two other different initiatives took place. One at the University of Minnesota, where researchers developed SAMM (Software-Aided Meeting Manager), a Unix-based system, as an integrated suite of tools intended to support meeting processes such as issue identification, brainstorming, voting, and agenda management (Dickson et al., 1989). The second initiative took place at the University of Michigan, where smaller EMS tools were developed (DeSanctis & Galuppe, 1987), for the Macintosh platform, that integrated behavioral science, group process theory, and adaptive structuration theory. The group of Prof. Nunamaker, at the University of Arizona, started up a new company to commercialize EMS, the Ventana Corporation, at the beginning of the 90's. Ventana's main product is called GroupSystems. In the list of tools listed below we included some more detailed information about this system.

EMS can help moderate and allow for various modi of interaction. Different from a meeting without technological support, no one using an EMS in an electronic meeting room need to loose the chance to speak up and provide ideas. Everyone can type into his or her own computer simultaneously. EMS can dramatically increase the number of people who can participate in a meeting. Some studies, suggested in (Nunamaker et al., 1995) showed that the productivity of meetings decreases when the number of people increases. EMS allows for dozens of people to work together effectively.

This group realized a selection of "lessons learned" that show us the benefits and the limitations of this set of tools. These lessons were learned over a year of observations involving more than 100 groups and dozens of meetings at IBM, and Boeing Corporation in the USA (Nunamaker et. al, 1995), (Nunamaker et al., 1996).

A later representative reference was developed in the GMD research institute, in Germany, for supporting face-to-face meetings (Streitz et al., 1994). They developed a system called DOLPHIN that includes a large, interactive electronic white board and individual workstations for meeting participants. The design of DOLPHIN was based on observational studies of editorial board meetings where an electronic newspaper was planned and created. Using DOLPHIN, board members could create and share informally freehand drawings or handwritten scribbles, and structured information such as hypermedia documents.

People who have never used meeting facilitation systems are often skeptical about their value. They point to the importance of social dynamics, face-to-face discussions, and nonverbal communication in meetings, apparently absent in anonymous typed interaction. Advocates of meeting facilitation systems have ready responses. First, people still talk to one another in a facilitated meeting; they use computers only in support of specific tasks, such as brainstorming. Second, these systems have fared well in some controlled experiments and field studies (Grudin & Poltrock, 1997).

Nevertheless, it is interesting to observe how meetings are taking place nowadays, when many academic and business people attend meetings with their own laptops. The main and not irrelevant difference is that these laptops may or may not be connected over a network, or over the Internet. Somehow, the "old" scenario proposed by Prof. Nunamaker and GroupSystems is a reality in our days, but with a different configuration. What is missing is a solution that would be easy and light to install and run in ad-hoc meeting situations.

The research of meetingware systems has evolved to other directions that include not only the topics discussed in the last decades, but also new topics regarding the organization setting and complexities of our present days, including new types of meetings (e.g. virtual meetings). Artificial Intelligence, for instance, is used in automatic mechanisms for extracting data from recorded meetings or to enable that roles previously realized by human-beings could be automatized; Information System and Business related areas contribute with more specialized tools to support smaller and specific processes; while Human Computer Interaction and CSCW areas contribute with understanding of group dynamics, or better interaction and collaborative possibilities. A more recent area that is also touching the grounds of meetings is the Ubiquitous Computing area, since meetings are multimedia or multimodal experiences, this research area is evaluating how to record, what to record, how to do automatic recognition of gesture, body movement, speech, etc., in order to make meetings more productive and useful.

Consequently, the state-of-the-art of meetingware systems is again too vast to be entirely defined in this work. The next section discusses some of the main concepts which are part of the meeting processes and are at the same time relevant for this research. Later, we present a group of tools showing a glance of the state-of-the-art in the field, starting from the early systems, like GroupSystems, until more modern approaches, like MeetingRoom (MeetingRoom, 2005).

3.1.1 Calendar

Calendar and scheduling products often serve as personal information management systems while helping teams coordinate their work. Individual users are supported by personal calendars, action item lists, contacts lists, and other features. Group Calendar Systems as defined in (Palen, 1999) are systems of on-line calendars that can be shared across a network. Individual keep their own calendar on-line, and allow various degrees of access to other users. Collaboration is supported, by simple sharing or viewing of other people's calendars, or by sending special meeting invitations through the Calendar system.

In the context of pre-meetings activities, calendars can be very useful in the activity of group time scheduling and resource planning (e.g. rooms, equipment). Coordination is supported by group calendars, meeting reminders, and especially by scheduling functions that aid in searching the calendars of multiple users to find convenient times for meetings and schedule resources such as meeting rooms. Integration with email can facilitate the invitation process (Grudin & Poltrock, 1997).

Calendars are tools for quickly seeing if people in a group are free or busy at a certain time (see Figure 3-1). Today's organizations have a need for human coordination. In most cases, this coordination requires meetings, which lead to the inevitable and often extreme difficult process of schedule coordination.

Too often, team leaders, project managers, executive assistants and product managers spend far too much time scheduling meetings. There's the frustration of telephone tag, email overload, and going back and forth trying to coordinate schedules only to have someone reschedule at the last minute. This is valuable time that could be better spent on other vital business issues (Meeting Maker, 2005).

Group Calendar offers, like any other automated tool, benefits and drawbacks. Starting from the drawbacks, these tools have to consider not only the technological aspect or the schedule definition, but also different cultural, social and work habits. Moreover, conflicts among users availability and the allowance for seeing private calendar and business calendar can impact the usage of these tools. Group Calendar Systems appear simple in function, but complex in execution (Palen, 1999).

Nonetheless, the coordination of groups nowadays are done even oversea, like for example the case of distributed groups, doing outsourcing and off shoring of system development, where different parts of the teams are located in different cities, regions, countries and even hemispheres. These complex distributed situations require more coordination than the old settings of teams who were located in the same building, and were easier coordinated and reached.

Despite meeting scheduling being an active research area for more than a decade, in fact, scheduling features in commercial products went unused for many years due to the lack of a "critical mass" of use in most environments—too many people found paper calendars more convenient (Grudin, 1988; 1994). Calendar applications have matured, sporting better interfaces, a range of individual-support features, and email integration. Users and technical infrastructures have also matured, leading to widespread use of scheduling in some environments (Grudin & Palen, 1995).

A limitation, very common in the past, which seems to be converging to a set of applications, is integration. While a member of a group uses application A for managing his time schedule, other members use B, or even C, and once an appointment is generated in application A, there is no easy way to transfer or synchronize it with the other members. Besides software integration, we are also facing the hardware integration requirements, when business and academic users are working mostly with more than a computer, e.g. a desktop, a notebook, a mobile phone, a PDA, etc.

On the other hand, we observe that group calendars are already part of many organizational settings and are part of highly used products (see previous chapter), like Lotus Products, Groove, SharePoint, among many others.

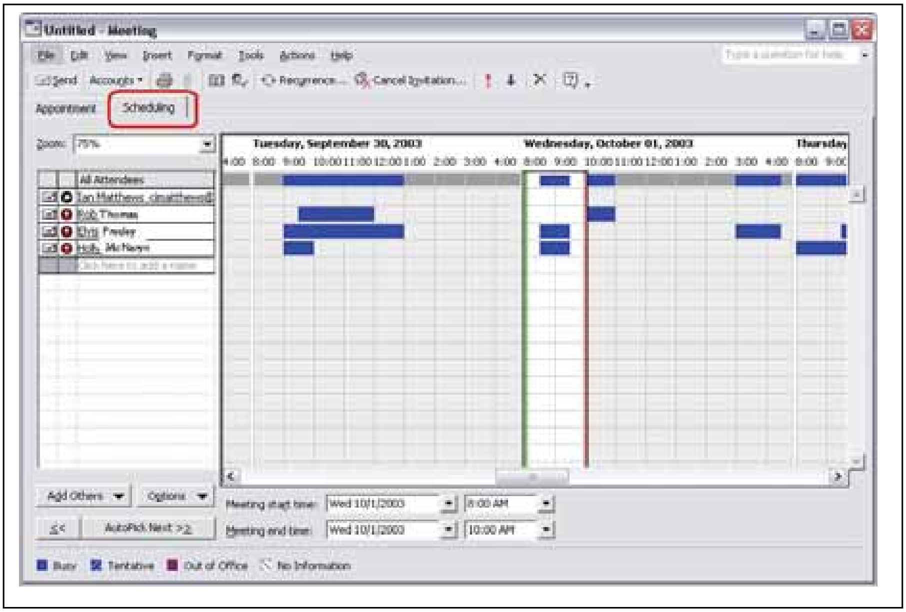


Figure 3-1: Example of a Group Calendar from Microsoft Outlook (Microsoft, 2005)

Most importantly, a Group Calendar tool must offer a list of features to improve and/or allow users for managing their time schedule better than their paper calendar notebooks. Among these features are: automatically posted to personal calendar, possibility to attach notes to appointments, enable negotiation (e.g. acceptance and rejection of meeting proposals), link appointments to contacts and projects, differentiate appointments (e.g. priority, categories), automatic detection of conflicts, automatic search of available time frames and resources, fast navigation of entire calendar year, configurable filtering functions (e.g. accounts, team, resources), easy management of private and cyclic appointments, configurable access controls for each appointment, configurable template notification system via email or short message, scheduling of external participants (e.g. organizational partners), configurable list of special events per user, dynamic print views for day, week and month and access to standard formats like iCalendar or XML data exportation.

3.1.2 Agenda

The term “*agenda*” (plural of “*agendum”*) *derives* from the Latin word “*agere*”, meaning to act. According to the Webster Online Dictionary (Webster, 2005), agenda is a list or outline of things to be considered or done, or an underlying often ideological plan or program. Specifically in a meeting context, an agenda is a *list of things to be dealt with* at the meeting.

Some authors see the pre-meeting activities as a planning phase, during which agenda items, logistics, schedule and pre-work are carried out (Szerdy & McCall, 1997), (Nunamaker et al., 1995), (Borges et al., 1999). Practitioner heuristics additionally suggest including information about the meeting's purpose; start time, breaks, and end time; agenda items by topics, presenter, time allotment, and action needed; who should attend; and what those attending should bring (e.g. (Murninghan, 1981) – cited in Niederman & Volkema, 1996).

One of the main advantages of having an agenda is the ability to bring an issue to the attention of a group or team, in contrast to the sequence of activities undertaken by a group toward performing a particular task (Niederman & Volkema, 1996), and providing a preview of the topics addressed so that individuals may be adequately prepared for a meeting. Most agendas are a list of topics without specification regarding how the group will address them.

Traditional formal meetings involve the contribution of agenda items by participants to attempt to fairly and completely cover a number of issues during a meeting (Raikundalia, 1998).

Besides, agenda serves an important role of guiding the decision-making process, playing a critical role in determining the meeting success or failure (Antunes et al., 1999). In (DeSanctis & Dickson, 1987) a GDSS is discussed which shows a good example of a system that provides an agenda as found in a decision-based meeting. A number of steps are carried out as indicated in the agenda, such as defining the problem involved, defining alternatives and rating alternatives. These steps are carried out in the order provided in the agenda so that to arrive at a final decision. The system also supported flexibility in agenda usage by providing three types of agenda:

- standard: having fixed content (highly structured meeting)
- open: allowing a facilitator to tailor the standard agenda (medium structure)
- an agenda where the standard agenda is tailorable by participants (low structure)

Successful GDSS meetings require expertise and experience in planning meetings and building agendas. Nierderman and Volkema (Niederman & Volkema, 1996) citing several references, support the idea that improving meetings generally stresses the value of composing and distributing an agenda prior to or at the start of a meeting. In a study done by the authors, they concluded that the impact of using agenda was highly rated, indicating some consensus that there is an impact of agenda on meeting outcomes. But it was not clear the impact on meeting outcomes. Satisfaction with process and time of meeting were also highly rated. But consequently, one can associatively consider that if the meetings are better, they might produce better results, or outcomes, so, in a longer run, directly or indirectly agendas have effects on meeting outcomes.

Last, but not least, (Niederman & Volkema, 1996) indicates that distribution of agenda prior to the meeting assures that all group members have the opportunity to be informed about topics to be covered. In spite of this practical advice, almost no research has presented evidence that these heuristics generate better meeting results.

3.1.3 Meeting and minutes

According to the Merrien-Webster dictionary (Webster, 2005), the concept of "meeting" has several meanings. For example, (i) to come together with especially at a particular time or place, or (ii) to come into the presence of, or (iii) to enter into conference, argument, or personal dealings with. All of them are valid in the context of this research. But more important to our discussion are the meetings where decisions are made.

There are many different types of meetings: general staff meetings, planning meetings, project reviews, transfer of information meetings, and decision-making meetings, project team meetings, brainstorming sessions, presentations and document reviews (Yankelovich et al., 2004). They can be categorized by size, frequency of occurrence, composition, motivation, and decision process used (Jay, 1976). Meetings are a particularly difficult area for establishing causal relationships because they can vary much among themselves in terms of goals/purposes, membership, size, group processes, leadership, and physical setting.

Especially in the decision making perspective, (DeSanctis & Gallupe, 1987) suggested three types of tasks that can be categorized according to what the group must accomplish during the course of a meeting. Major group goals in decision-related meetings include:

- Generating ideas and actions:
    - Planning tasks require generation of action-oriented plans. Creativity tasks require generation of novel ideas.
- Choosing alternatives:
    - Intellective tasks require generation of correct alternative. Preference tasks require selection of an alternative for which there is no objective criterion of correctness.
- Negotiation solutions:
    - Cognitive conflict tasks involve resolution of conflicting viewpoints and mixed-motive tasks involve resolution of conflicting motives or interests.

Meetings are ubiquitous in projects; they foster effective teamwork and appear necessary for human connection. It is important for a team to have a forum to share concerns about project issues or progress on work that is currently underway (Garcia et al., 2004).

Despite their importance and frequency, meeting participants often feel time is wasted. Low group participation, free riders, a bad decision-making process, and failure to hold a group's attention are some symptoms of a bad meeting (Garcia et al., 2004).

Following the literature on DSS and GSS, six variables were identified as important group meeting outcomes in the organizational context. These are:

- Quality of meeting deliverables (decisions, plans, reports, etc)
- Participant satisfaction with meeting outcomes
- Participant satisfaction with meeting process
- Amount of group consensus
- Efficiency (time it takes to run a meeting)
- Probability of positive long-term organizational impacts

Various methods for running meetings have improved meeting efficiency (Garcia et al., 2004). These methods vary according to the procedure followed, but they all emphasize the need for a formal agenda that should be pre-disclosed to the group and strictly followed during the meeting.

Documenting meetings can be an important part of organizational activities. Meeting minutes, a.k.a. protocol, or list of actions, or list of decisions made and issues discussed in meetings, constitute a portion of the organizational memory. Right after a meeting is it often useful to look at the notes to review and act on decisions (Chiu et al. 2001).

(Chiu et al. 2001) did a survey to determine the preferences for accessing meeting minutes. A key question was whether people like to have meeting minutes delivered as email or to have the minutes on the Web for browsing, i.e. the push/pull question. Thirteen people were interviewed and the answers were: 5 preferred email, 2 preferred the Web, 5 preferred both and 1 said it would not matter. This showed the requirement of having both: email and the Web possibilities. Besides that a summarized list of requirement was provided:

- Provide both email delivery and Web access
- Wireless laptop for taking meeting notes
- An application for taking notes
- Timestamp note items for correlating with media
- Support revision right after a meeting
- Smart link generation using contextual metadata

Revision is a crucial step in the meeting minutes process that gives an opportunity for the filling in of missed items, checking details and fixing typos. We consider that the revision of meeting minutes a highly interactive activity that can be supported by the principles from the CSCW area, either synchronously, with collaborative editing, or asynchronously, through versioning control or even negotiation support.

The time to access minutes is also important to be discussed. (Chiu et al. 2001) identified three important timeframes:

- **During the meeting (real time)**
  - Questions and discussions referring to a slide show
  - Instant replay of a remark or event with recorded audio or video
- **Right after the meeting (minutes to days)**
  - Review details, action items
  - Skim email minutes if missed the meeting
- **After an indefinite duration (weeks to months)**
  - Recall plans and accomplishments over past year
  - Organizational memory

More recently with the advent of distributed meetings, other problems appeared. In (Yankelovich et al, 2004), based on an internal study realized at Sun Microsystems, the authors compiled a list of the top problems that people have in distributed meetings. Among them were audio problems (e.g. some people could not hear, too much extraneous noise), behavior problems (e.g. speakers did not check others for understanding, meeting was not well facilitated), and technical problems (e.g.

difficulty to identify who was speaking, necessary documents were not available). Ease of use was perhaps the strongest theme to emerge from the interviews.

In a more recent approach (Schultz et al., 2001) other aspects of meetings are discussed. For instance, the ability to detect opinions, statements, and emotions in meetings recorded. These are seldom recorded as part of meeting minutes, yet they play a crucial role in determining the meaning and importance of what was said. Queries such as "Was John angry when he was discussing Brazil?" or "Show me John's opinions about Brazil" would become possible.

Meeting outcome documents (e.g. meeting minutes) are not formalized in the relevant literature. Each person, group or organization follows its own way of documenting meetings. From the list of tools we have investigated, except for the suggestion of "itemizing" the meeting content into issues, or topics, no more formal approaches were found. Curiously, in Microsoft website (url: http://office.microsoft.com/en-us/templates**),** concerning the Office package, there is a number of meeting templates available to be downloaded, suggesting some structures for minutes.

3.1.4 Anonymity

The possibility of supporting anonymity in group communication is a characteristic vastly discussed in the area of CSCW, for example (McLeod, 1997) and (Davenport, 2002). This is a feature, characteristic of support tools that allow users to preserve their identities, so that they can feel more comfortable to suggest ideas, without being directly critized or measured. Anonymity has helped shy users to participate more active and has also provided a possibility of suggestion/ideas/critics analysis per se, instead of relating its value to the person who said or wrote certain information.

In decision follow-up support tools, this characteristic should also be considered, although the knowledge about the participants, ideas and problems can be made explicit easier if the names of the people involved are clear and known. Nevertheless, some groups may feel uncomfortable to have a tool "observing" or "measuring" every action, task or process, under her/his responsibility. In these cases, anonymity could be considered an important feature.

3.1.5 Roles

In the context of meeting, either before, during or after they take place, the concept of "role" turn to be of great importance. Role is a function, a position assigned to a particular person, thing, or group of people. Or according to (Webster, 2005) a socially expected behavior pattern usually determined by an individual's status in a particular society, or even a function or part performed especially in a particular operation or process. Or as defined in (Belbin, 2005) a team role is a tendency to behave, contribute and interrelate with others in a particular way. Roles are elements more stable than activities that compose a process (Araujo, 2000). Roles are defined according to the tasks or activities that take place in an organizational process.

In the context of meetings (Borges et al., 1999), (Nunamaker & Briggs, 1991; Nunamaker et al., 1995) stressed the importance of group leadership. People react differently to new technologies and some participants need stimulus to adopt a positive attitude towards the supporting system, and that can be exactly the situation of introducing automated tools to support meeting processes. Group leadership, explicitly represented or not, is very relevant in meeting contexts regarding the acceptance or not of technological support.

Facilitation is one of the several third party processes studied in organizational behavior (Lewicki et al, 1992), which involve the following activities:

- mediation (support interactions among disputants)
- arbitration (resolve disputes while giving equal opportunities)
- inquisition (stronger than arbitration)
- process consultation (facilitate problem solving)

Or additionally, concerning the decision making process, group facilitation is a process in which a person who is acceptable to all members of the group intervenes to help improving the way it identifies and solves problems and makes decisions (Schwarz, 1994).

The most discussed and investigated role in the meetings scenario is the role of a facilitator, i.e. the person who carries out the facilitation tasks of organizing pre-meetings, planning decision-making processes and supervising the technology usage during the meetings. Individuals within an organization or for other firms, whose job is planning and executing meetings would be meeting facilitators. However, there are additionally large numbers of individuals who work for example, as trainers/teachers within an organization or as organization development specialists who spend a significant percentage of their time facilitating meetings. Moreover, there are other individuals who assist organization in a variety of projects who as an outgrowth of other tasks must lead meetings to accomplish their objectives. Finally, there are also department or division heads and other managers who in the normal course of fulfilling their obligations conduct meetings.

An EMS can make usage of these concepts and possibilities or not, but certainly investigations about the different impacts of having roles is worth being done.

3.2 Meeting and Decision Follow Up

As discussed in the previous chapter, from our perspectives, technology can help us improve:

- The formalization of a decision implementation, using management systems, allowing planning, tracking and getting warning signals about problems, providing control over a decision implementation;
- The use of technology in organizing, storing and making information and knowledge generated during a decision implementation available, contributing to improve the quality of future decisions;

However, it is difficult to formalize and follow decisions, since they can be made in different moments in organizations, like meetings or during informal communications.

Hence, to discuss better the decision follow-up approach proposed in this research, let us review the concepts regarding electronic communication, which is one of the means we aim at using for providing decision follow-up automatically. These concepts can be part of meeting settings, but also analyzed independently. Here, we focus our discussion on the meeting and related activities.

3.2.1 Communication

Communication, in a broad sense, is an activity present in all phases of the decision making process. Communication can be realized in different ways (synchronously, asynchronously, with open identification or based on anonymity), using different channels (text, voice and images) and supported by tools (telephone, fax, email, chat, discussion forums, videoconference, voice conference, etc).

The different formats or channels, as well as the use of tools, can increase the reachability, the broadness and the opportunities for interaction, since participants of a communication process do not need to be available at the same time and at the same place.

In decision follow-up, communication seems to be one of the most important aspects to be observed. (Aggarwal, 2001) emphasizes this idea suggesting that the new generation of DSS will have to focus on decision, decision flow and decision dependency, and communication is a very important aspect to be supported by these tools. Actually, formal (e.g. letters) or informal (e.g. conversation in a cafeteria) communication is used to transmit information, knowledge, to make new decisions, to externalize problems, and successful stories.

Since we aim at discussing how computers can support decision follow-ups, especially based on the information available in communication processes, our analysis will be restricted to the possibilities supported by computers. In the following sub-sections we analyze some important aspects involved in communication based on computers.

3.2.1.1 Chat or Instant Messengers

Chat and Instant Messengers (IM) are technologies that allow group synchronous interaction. Each member can post her/his ideas, either through textual messages or currently, with “emoticons” (icons which represent emotions), and these messages are seen by those who are sharing the communication channel, or simply by those who are “on-line”.

The instant messaging market is growing rapidly, among both consumers and business users, according to a study done by Radicati, a technology market research group, finished in 2004 (Radicati, 2005). The study foresses especially strong growth for IM Management products, which provide corporations with the ability to log, manage, and archive IM according to external government regulations, and/or internal corporate policies. This market is projected to grow at an average rate of 40% per year over the next four years, growing from $32 millions today to $121 millions in 2008.

In the situation of decision follow-up, chat and IM tools can be of great value to help rapid communication exchanges. Either to get quickly some needed information or even to solve problems. Another important use of chat and IM tools nowadays is their use as a communication channel with experts. Through chats and IM tools, distributed people consult colleagues considered experts in certain fields in a fast way.

From a second perspective one can argue, why not to use the telephone instead? One of the problems with the telephone is the cost involved. Chat or IM tools are based on Internet access. Either you have it or you do not pay anything extra for using these tools, once you have an Internet connection. The second point is accessibility. Normally, while talking on the phone, except phones with headphones, the user occupies one of her/his hands, while in a chat tool the user is interacting only with the computer, changing between different windows or tasks. The third argument is that the information exchanged in a chat or IM tool can be used electronically, to document the conversation and to set up any data in the system. One user can send electronic data via chat or IM, but not via a normal phone. And although mobile phones are able to receive electronic data, the costs are still too high to compensate for its usage.

3.2.1.2 Videoconference

Videoconference is a technology to support synchronous communication among different people, normally used for distributed groups. Like in chat or IM, each participant can pose her/his idea through voice and gestures (images) and all the other users connected (on-line) can see the interaction. Since videoconference can vary in robustness and price, it is not necessary for all members of the conference to use video. Maybe because of the lack of availability of cameras or costs, some members use only voice, while others use the full capabilities of this technology.

The storage of conversation made through videoconference demand a lot of hardware space. Although this is one of the greatest tools invented to support distributed synchronous communication, most of the time, after the tool is shut down, the history of the communication is lost.

In the context of decision follow-up video conferencing can be very useful in supporting people communication for solving problems, exchange ideas or to provide follow-up itself. But the costs of videoconference tool do not allow many groups and organizations to base their communication process on this technology. Besides that, videoconference tools, different from emails, chat and IM tools demand users to be online the whole time of the conversation. While these other tools allow for some level of asynchronicity.

3.2.1.3 Discussion forums

Discussion forums or mailing lists work similar to emails, since they are asynchronous message based. However, different from emails, in discussion forums and mailing lists, every member of the discussion or list is able to see all messages addressed to the discussion or list of users.

There is a slight difference between these two concepts. Mailing list is usually an automated system that allows people to send *e-mail* to one address, whereupon their message is copied and sent to all of the other subscribers to the mail list. In this way, people who have many different kinds of e-mail access can participate in discussions together. While a discussion forum (also known by various other names such as discussion board, discussion group, message board, and online forum) is a general term for any online "bulletin board" where you can leave and expect to see responses to messages you have left. Or you can just read the board.

Discussion forums can be used to support communication and decision follow-up, but they suffer from the lack of flexibility. A user does not have flexibility to organize the messages the way s/he wants. Normally, this is done automatically by the system according to the “subject” field of the messages.

3.2.1.4 E-mail

E-mail is the one of most successful computer supported communication tool available nowadays, specifically oriented to asynchronous situations. According to a study done by Radicati Group (Radicati, 2005), 651 millions people around the world now use email regularly. This figure is expected to grow steadily over the next four years, reaching 850 millions of users by the end of 2008.

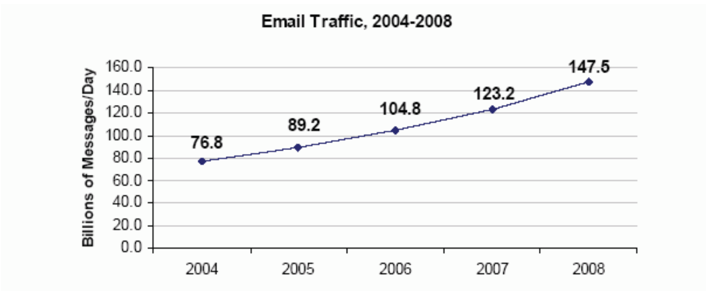


Figure 3-2: Email Traffic comparison between 2004-2008 (Radicati, 2005)

Email has not changed its basic principles since its invention several decades ago by ARPANET. It is still a mean of sending and receiving messages to and from people, with a subject, a body (normally the text) and some related information attached to it (e.g. attached documents, pictures, video, etc). Other two concepts used by email systems are: “CC” (carbon copy), which represents indirectly related people or even “BCC” (blind carbon copy), which represents hidden receivers, i.e., those who receive the message will not know who was included in the BCC field, although the opposite is not true.

In the scenario of decision follow-ups emails are already being used. Many users rely on emails to get up to date information about projects, tasks, responsibilities and track of on going processes. (Gwizdka, 2004), (Whittaker & Sidner, 1996) are examples of studies that tried to understand how users organize their emails and the information contained in them, like to-dos, calendar events, to-reads, etc. In the late 80s, (Mackay, 1988) described how email supports a variety of time and task management activities.

Email has great advantages over other tools. It is simple, not costly (except the storage costs, which are becoming very high), it is already accepted as a communication tool in many organizations and it supports asynchronous communication.

On the other hand, e-mails are not easily structured. Some available tools provide basic mechanisms to organize them in folders, sub-groups, by sender/receiver, incoming and outcoming messages, but the context of the messages and even the content are not analyzed by these mechanisms. Thus, if the user writes a "reply" to another user, using an old email as a basis, but writes about a totally different subject, these structure mechanisms will not recognize the difference.

An important step in the evolution of email was to provide a capability for creating additional fields. The Information Lens (Malone et al., 1986a) demonstrated how these fields, combined with agent technology, could help users process and handle their mail. Today many groupware products, including email systems, contain tools for constructing such agents, and improved human-computer interfaces that make them more usable. Some tools work in that direction offering some limited filters that can be built by the users, e.g. e-mails coming from sender "maria" should be moved to folder "very important messages", etc. But still the structure is almost 100% dependent on an artificial organization mainly done by the user.

In the case of decision follow-up, emails can be very useful. Many project leaders, mainly of distributed projects, rely on email to catch up with information. The only problem, as we mentioned before, is that the information is there, but the content analysis of them has to be made by users. If a user really wants to structure a conversation, not only based on the "subject" field of the email messages, the user would have to read each message again and create "artificial" structures to store them in a meaningful way.

Although the principal purpose of email is communication among people, its structure, reliability, and universality have encouraged its use as a mean of delivering messages between processes and people or among processes. In this way, email supports coordination as well as communication. For example, many Lotus Notes applications, workflow management products, and calendar systems use email to alert a person of events or of tasks to be performed. Some workflow management systems use email as the mechanism for routing and presenting work to users (Grudin & Poltrock, 1997).

One of the most successful email applications in our current days is Microsoft Outlook. We would like to discuss some of its features, since it has such a representative participation in the world market share (in 2002, according to the Fortune magazine, MS Exchange and Outlook together had more than 40% of the market share). A recent study from Radicati shows that in 2005 they are the leader in the market.

There are two big sets of features that are worth detailing concerning Outlook: integration and collaboration.

First of all, Outlook communicates in a smooth way with Microsoft products, which can be seen as advantage or disadvantage technically speaking, but which shows the compliance with the business taking place all over the globe. Outlook 2003 (see Figure 3-3) provides an integrated solution for managing and organizing not only e-mail messages, but also schedules, tasks, notes, contacts, and other information. It also includes possibilities to manage communication through the use of filters, rules, search facilities and different ways of viewing messages (e.g. folders, or various message properties).

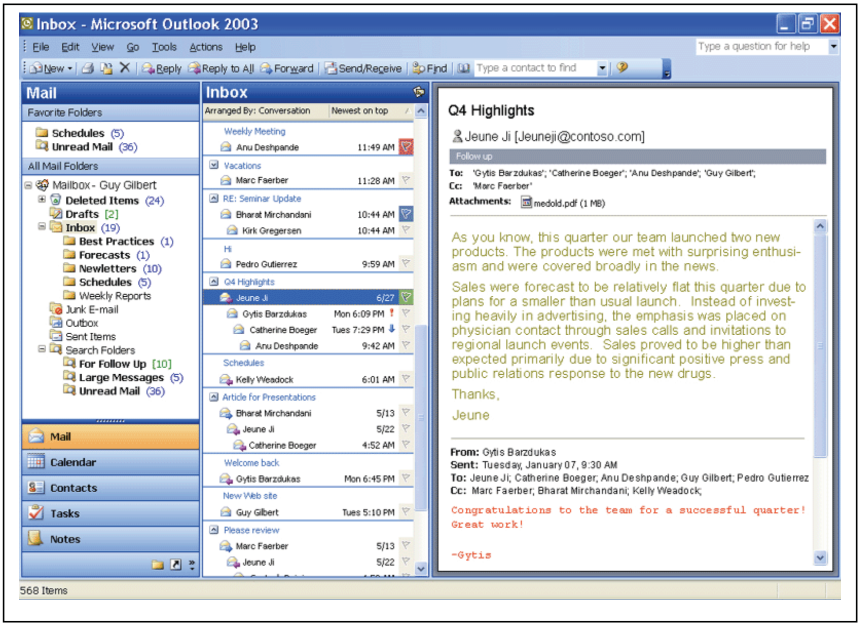


Figure 3-3: A screenshot of Microsoft Outlook 2003 (Microsoft, 2005)

From the collaborative perspective, the system is integrated with Instant Messenger, and allows for supporting control distribution of sensitive work, preventing recipients from forwarding, copying, or printing important e-mail messages by using information rights management (IRM) functionality. Together with Microsoft SharePoint Server, users can save e-mail message attachments to shared workspaces where other team members can get the latest versions, check files in or out, or even save task lists, related files, links, and member lists.

A recent study from a Software Magazine from January 2005 (url: http://stuff.techwhack.com/archives/category/web-stuff/e-mail-client/) showed an interesting evolution of Microsoft Outlook, in response to the market pressure done by Google (Google, 2005) email client, Gmail. According to this study, Microsoft is feeling the pinch of Google's Webmail service GMail. The latest news from Microsoft is that they would now be offering a subscription service, which would tightly bring together Outlook and MSN Hotmail (the webmail of Microsoft), which is used in more than 220 countries and territories, with about 190 millions of users. These two products will be integrated with each other for customers looking for a combined subscription based module. This new offer has been labeled 'Microsoft Office Outlook Live". The interesting part of this deal is that users do not need to own Outlook for this package. Microsoft is offering the excellent Personal Information Manager (PIM) software on subscription bases. The package includes Office Outlook 2003 along with 2GB of online storage combined with features like spam and virus protection. Users will also be able to send mails as big as 20MB through this deal. Outlook will hence function as a front-end of MSN Hotmail and users will be able to use it not only for mail purposes but also for the calendar and contact manager. These changes will be synchronized with the online servers so that users can get access to their mail through the webmail interface from anywhere.

Two other initiatives promise to reactivate the "market dispute" in 2005, besides the Gmail from Google: (i) Mozilla has released its new version of email client, named "Thunderbird" 1.0, aiming, mainly, at improving the serious problems users are facing to deal with spam and virus and (ii) the Opensource community is working on a platform for PIM, named "Chandler", which is intended for use in everyday information and communication tasks, such as composing and reading email, managing an appointment calendar and keeping a contact list.

Figure 3-4: An annotation icon example (Kerr & Wilcox, 2004)

Another interesting recent research initiative worth commenting is the ReMail (Reinventing Email), from IBM (Kerr & Wilcox, 2004). The main goal of this project was to reduce problems faced when one manages his/her own email, particularly keeping track of information, feeling pressure to respond quickly, and feeling overwhelmed by mail volume. This prototype includes, besides the regular email functionality, integration with chat tools (instant messenger from Lotus—Lotus Same Time), calendar, and some new concepts: annotations, threads, collections, and different ways of visualizing emails.

An "annotation" (Figure 3-4) is a digital note equivalent of a Post-It note (from 3M). It allows users to mark a message with a small colored icon of their choice that could also contain a text, flagging a message for later reference.

Because people spend a great time searching for related messages, they implemented the "threads", which work similar to a conversation. Thread map is an interactive visualization which represents messages with a reply-to relationship. In their evaluation, threads were seen as a useful feature, for finding information and for collecting related documents. Chronological visualization was seen as an important feature as well. A new variation of visualization for threads was included in a later phase of the project and was named "correspondent map" (Figure 3-5). This feature was added to help users view messages from people they corresponded with most frequently. This map groups messages within a collection by sender, domain and number of messages sent.

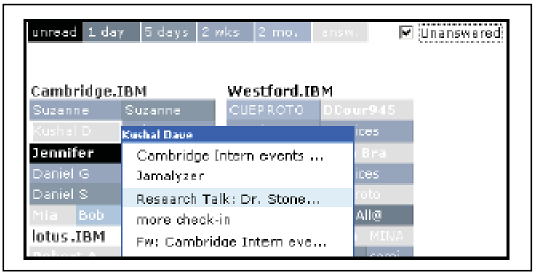


Figure 3-5: Correspondent Map (Kerr & Wilcox, 2004)

The "collections" is a feature that blends standard folder view with dynamic rule-based views. Users are allowed to move message in and out of a collection, and a message can belong to multiple collections. An interesting capability of a collection is to be "in-sight" and "out-of-sight", meaning respectively that a message is seen in the Inbox or not.

A very important remark done by the authors in this reference is that, actually, the biggest problems users deal with in their most recent study was efficiently handling activities in which email played an important role, but often involved other items as well. This remark is relevant for our research exactly because we agree with that and, moreover, we aim at using email as a tool to automatically create and allow follow-ups of decisions. One of our hypothesis is that the "content is basically there", but the structure, some new management features and the visualization requirements are not.

Another recent relevant initiative is the proposal of a "Meta-Mail" tool (Dyson, 2004). This author mainly reinforces what we tried to explain in this section: the great importance of email in organizations, and its role, which is not anymore only a medium for asynchronous exchange of simple messages, but a basis for many other dimensions, like decision management, task management, among others.

The author defends the idea that process-management tools must be smoothly integrated with email and other ad hoc communication functions. For her, Meta-mail's process elements will be user-definable, inspectable, reusable and above all personal. Behind these ideas, a feature for automatically generating metadata is mandatory. The author finally discusses the impact of effective search engines in emails and other data sources. For instance, Google is making lots of advertisement about its great capacity of searching more than folders in the concept of Gmail. On the other hand, as asserted by (Dyson, 2004) text search can catch topics (or nouns, what is something's about), but it cannot catch the implicit transactions (or verbs, such as commitments, deadlines and decisions).

Finally we would like to cite two other relevant research approaches concerning emails and task management, one from Jacek Gwizdka, at the University of Toronto and the other from Victoria Belloti and others at Palo Alto Research Center.

In (Gwizdka, 2002b) the "reinvention" of the Inbox is proposed, as a solution for users who use email as a time and task management tool. Inboxes are often used to keep those messages referring to the future that cannot be dealt with upon their arrival. These messages are used as reminders about both email tasks and non-email task and events. According to the author, information from these messages are not transferred out of email, presumably because of the high cost of doing so. This high cost is often due to the lack of integration of email with other software applications or media. So, his approach suggest creating mechanisms to support the management of pending tasks in email. The author discusses one prototype for dealing with these problem: a 2D interface that is used to display temporal information along with the priorities of pending tasks. Tasks are represented by dots on a 2D plane with task attributes, time and subject, shown on the x-axis and y-axis. In another article (Gwizdka, 2002a) discusses some positive results with his approach, regarding variables like efficiency of finding information, although most of the variables evaluated are highly dependent on the users cognitive ability.

In Palo Alto Research Center another approach was investigated. In (Bellotti et al, 2003) they discuss a new interface for Microsoft Outlook, called "Taskmaster", which aimed at supporting task management in emails. Based on this problem, they identified seven specific problems:

1. Keeping track of lots of concurrent actions: One's own to-dos and to-dos one expects from others.
2. Marking things as important or outstanding amongst the less important items.
3. Managing activity extending over time or keeping track of threads of activity and discussions.
4. Managing deadlines and reminders, which may be associated with particular messages or other content.
5. Collating related items (e.g., an extended thread or responses to a survey) and associated files and links.
6. Application switching and window management.
7. Most important, getting a task oriented overview, at a glance, rather than scrolling around inspecting folders.

In Taskmaster users can manage their emails in a different way. One of the main interesting concepts discussed and investigated is the "thrask", i.e., a mix of "task" and "thread". A thrask supports semi-automatic collections of these items. A remarkable example implemented in thrask is to allow incoming and outcoming messages to be viewed together, which helps with problem number 1 listed above. Users can rename thrasks according to more understanding names, can fine-tune the contents of a thrask by adding items and thrasks to other thrask or by moving items out. Technically unrelated thrasks can be combined.

Both initiatives are interesting because they are related to our approach with providing decision follow-up, via emails and tasks. The next section discusses more specifically the task management state-of-the-art.

3.2.2 Tasks

Another concept which is relevant to our research and it is related to the proposal defined in the next chapter is the concept of "task". Task can be defined as a mission, an assignment, and obligation, a logic unit of work, a process that cannot be subdivided any further, i.e. an atomic process. The difference between a process and a task is: a process is constructed from tasks and conditions or order of tasks.

Tasks affect all levels of an organization (operational, managerial, strategic), they are related to „deadlines“, production, decisions, projects, implementation, finalization of activities, reports, etc. Working in groups is a reality therefore, we need to coordinate people‘s work and most of this work is organized in tasks.

We could start our discussion concerning tasks related to the pre-meeting activities, but since this is not the objective of our research, we will concentrate our analysis on the context of meetings. Tasks are normally defined when users elaborate meeting minutes. Tasks, also called, actions, contains the most relevant information regarding the steps for the post-meeting activities, i.e. what needs to be done.

However, it is important to observe that the definition of what a task is can be very subjective. Normally, a task is atomic for the person defining or ordering it, but for the person carrying it out is often a nonatomic one. This judgment is very subjective.

Our main aim behind this discussion is to show that meeting support tools and task management, together, can enable users to lookup at the results of tasks and get a better view of the follow-up of decisions made in meetings.

The link between decisions and tasks should start during the meeting minutes documentation process and keep on going during the execution and change of each task status by each assigned user. At any moment, a decision maker or a decision responsible should be able to request a decision follow-up and the system would provide a summarized view, based on the tasks related to that decision.

One of the most interesting aspects of this integration is to use the opportunities of daily activities of task management, common to many users, to provide a follow-up for decisions, not creating an extra tool or an extra effort from the user’s perspective.

We define as “**task management**” the activities that deal with the generation, scheduling, and chronicling of project activities. Following the idea of one business guru, Peter Drucker: “*Working smarter not harder requires technology that automates and integrates tasks, so more can be done in less time with fewer mistakes.”*, cited in (Reekes, 2002).

Task management involves four basic mechanisms structure: sequence (one after the other), selection (OR), parallelization (AND) and iteration. Another way of looking for task management support is based on negotiation models of “requester” and “contractor”, where the first (e.g. a project coordinator) proposes a task to be done, i.e. assign, request, and the contractor (e.g., system developer) accepts/commits, or rejects, or forwards the request to another person. This process can be cyclic until someone commits and executes a task or cancel its execution. Figure 3-6 illustrates a basic example of Microsoft Project task management working together with an email client.

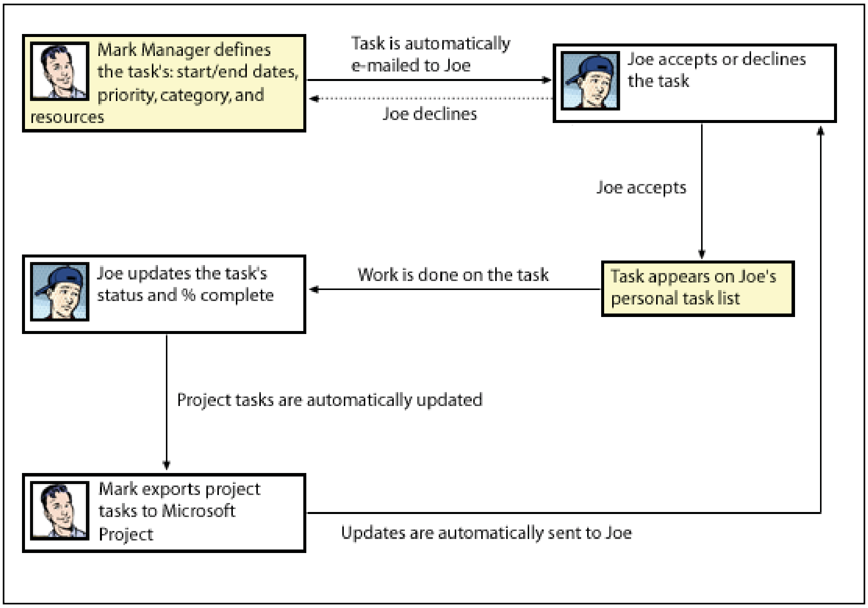


Figure 3-6: Microsoft Project and Email supporting task management (Microsoft, 2005)

In this section, similar to the Email section presented before, we do not aim at discussing the whole state-of-the-art of these tools, but to provide a glance and a context of some important concepts relevant to our research, which have been already discussed in other research or project.

In our investigation, we are considering tasks in the context of meetings and projects. Tasks are defined during meetings, or as a consequence of meetings, which belong to projects. We could also discuss individual task management problems, support or theories, but this is not part of our research scope.

In Project Management, tasks are the units of work and are defined along the life-cycle of the project. Tasks are part of planning and executing macro activities in projects. Normally, a project planning starts with the elaboration of a work breakdown structure (Wilson & Sifer, 1988) that formulates a hierarchical diagram, which aim at representing the work activities and its relationships or dependencies. Another common practice is to start with such a diagram on a high level of abstraction and little by little construct smaller refinements, until reaching atomic tasks, assigned to individuals. All this work aims at defining better the amount of time and resources a project might need to succeed.

A much disseminated project management tool is the Microsoft Project. This system is integrated to MS Outlook, and allows various project control mechanisms, like control of time required, schedules, tasks, resources (people, material, and equipment), project objectives, constraints, scope, a plan builder, track and management of activities and different project graphics visualization, e.g. Gantt charts.

Other interesting task management approaches were discussed by (Fialho, 1998) and (Kreifelts et al., 1993).

For instance, (Kreifelts et al, 1993) describe a tool for managing distributed work: the Task Manager. This tool provides users with organization possibilities, e.g. create, modify and refine cooperative tasks, monitor their progress, share documents and services, and exchange messages during task performance. The Task Manager organizes distributed work in tasks which have a person responsible, a deadline, other participants, the material necessary for task performance, and possibly subtasks. The primarily user interface of the Task Manager is a hierarchically structured to-do list which displays all tasks a user is involved in and which allows direct access to the relevant information attached to a task. This work gave the basics for the collaborative system previously discussed: the BSCW system. Besides task management, the system evolved to a document management system, with lots of features to support collaborative work. And actually, nowadays, task management is not the focus of the product anymore. Anyhow, this initiative shows us that one need is intrinsically present in the other, i.e., task management is part of collaborative work.

In (Fialho, 1998) "Tasker" is proposed. This system aims at supporting project management, through task management, based on speech act theory of Searle (Searle, 1969). For the author, once a task is defined, a conversation for action and a conversation for clarification can take place among project members. Conversations occur in various scenarios described in (Fialho, 1998). For example, in the situation when a team member detects that a task must be executed, which is still not assigned; or a conversation between a requester and a performer regarded an assigned task, or even a situation where a task is pending because another task must be accomplished first. In this latter case a conversation can take place to clarify the process, the dependencies and conflicts that might appear.

In the next chapters, we will clarify our task management approach, in the context of meetings, and the negotiation model behind it, which is also based on Speech Act theory, as in the Tasker. The next section will discuss a set of meeting support tools.

3.3 State-of-the-art of meetingware

3.3.1 SISCO

SISCO (Borges et al., 1999), an asynchronous support meeting preparation (ASMP) tool, aims at assisting the preparation for a decision making meeting. The basis of SISCO includes improvements on communication among meeting participants and organizers, but also support for group interfaces, group memory, and group communication. The authors state that in most cases decisions are too complex to be made over an asynchronous distributed system and thus, they assume that they will be made during the meeting itself.

The SISCO system divides the pre-meeting phase into four groups of activities: preparation, setting-up, development and closing-up.

The first service of SISCO is the support to create a pre-meeting, which corresponds to the creation of the structure that will contain the group memory. In this context, the coordinator, i.e. the person responsible for the main activities of the pre-meeting's definition and development, organizes the setup of the activities, which takes place as follows:

- It starts with the creation of a pre-meeting, where the coordinator defines which people will take part in the discussion process and request the system to prepare their initial access.
- The coordinator assigns himself and each user one of the four discussion roles: coordinator, facilitator, contributor and observer.
- He creates and modifies pre-meeting restrictions regarding pre-decisions and constraints.
- He creates and modifies pre-agenda items, and a corresponding facilitator and discussion deadline for each of them.
- He organizes a comprehensive report at the end of the pre-meeting for use in the face-to-face meeting.
- Finally, monitoring the development of the pre-meeting.

Figure 3-7: Discussion Hierarchy of SISCO (Borges et al., 1999)

Then, after the setup preparation several discussions can take place where contributions are created. Figure 3-7 shows an example. The authors emphasize the importance of structuring the discussions, since it allows easy retrieval, idea evolution tracking and preparation for decisions. The communication via the discussion database is formal, structured, persistent, and public (to the authorized participants).

The discussion model adopted by SISCO is an extension of the IBIS argumentation model. It has the same basic elements: issue, position and argument but includes remark and task. The data model also includes information about participants, the discussion agenda items with their objectives and pre-defined constraints. The participants are classified according to the role they have in the pre-meeting. Besides the coordinator, a participant can be a facilitator, a contributor or an observer of an item in the discussion. Facilitator is the person responsible for supervising the pre-meeting discussion of a particular agenda item. Contributors produce the actual discussion. For each agenda item, there is a group of participants who are invited to be contributors for that item.

An interesting concept developed in SISCO is the “participameter”. It is a tool to measure the quantitative level of participation in a discussion. It differentiates the participation according to reading, elements contributed, the frequency of connections, the number of tasks for which each person has been a candidate, and the numbers of tasks achieved.

The closing-up activity takes place when the coordinator judges that the discussion is mature enough to proceed to a meeting. This decision may cover the entire agenda or a subset of items. After that, the system informs the involved people about the structure and logistics of the meeting that is going to take place.

Besides the support for the four main group activities, SISCO includes ways to quickly find discussion elements by full-text search, navigation by author, following a discussion thread, and accessing unread material. The displayed information shows the relation it has with other discussion elements.

This is a system which shares some of our ideas, i.e. the idea of separating meeting in phases and dealing with each phase separetely, but at the same time complementary. However, SISCO is very focused on the pre-meeting phase only, leaving open the possibilities for the other two phases of the meeting cycle.

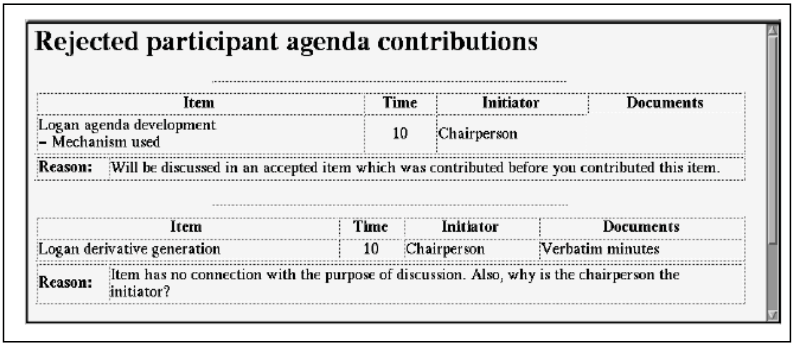
3.3.2 Logan

Logan is a Web Electronic Meeting Document Manager (WEMDM) which provides a web-based automated process for asynchronous, collaborative development of a textual meeting agenda (Rainkundalia, 1998). The process includes high meeting participant involvement through participant contributions to the agenda. And the construction of the final agenda is automated via *secretarius* moderation.

Logan supports asynchronous, distributed, formal meeting collaboration using documents in conjunction with a tool for carrying out discussions. It provides analysis of meeting transcripts (logs) to generate dynamically derived documents (derivatives) for use in a meeting as meeting’s guide usage, helping in giving direction and guidance to meetings.

The creation of a Logan agenda involves participants to a high degree. The agenda is developed collaboratively from contributions made voluntarily by invited participants before the meeting. A contribution is a set of details suggested by a participant as a discussion item. These details include:

- description of the item: the name of the item
- initiator: the participant to commence the discussion of the item
- time: estimate of number of minutes required to discuss the item
- documents: the documents to use during discussion of the item



Rejected participant agenda contributions

Item	Time	Initiator	Documents
Logan agenda development – Mechanism used	10	Chairperson	
Reason:	Will be discussed in an accepted item which was contributed before you contributed this item.		

Item	Time	Initiator	Documents
Logan derivative generation	10	Chairperson	Verbatim minutes
Reason:	Item has no connection with the purpose of discussion. Also, why is the chairperson the initiator?		

Figure 3-8: Rejected participant agenda contributions (Rainkundalia, 1998)

Then, participants have access to the agenda, possibly including their contributions, and then submitting them to the secretarius (the participant assigned administrative meeting activities). The secretarius is also the responsible for determining which contributions are relevant and appropriate for an specific meeting. The system allows discriminating "accepted" items from "rejected" items, together with a field containing the "reason" why the item is being rejected. When an item is rejected, it does not mean it cannot take part of the final agenda. In this case, the participant who has suggested the item can suggest it again, after reworking it or improving the argument why it should remain. A final date for contributions is determined by the secretary. The final agenda may result from numerous cycles of contributions and collaboration amongst participants and the secretarius.

This is a system which focuses only on the preparation of agendas. And although it advances in this area, the other two meeting phases (i.e. the meeting itself and the post-meeting phases) are not discussed or supported by this tool.

3.3.3 Agenda Planning Technique

In (Garcia et al., 2004) the authors describe an agenda planning technique with a built-in incentive mechanism, based on the VCG (Vickrey-Clarke-Groves) method from game theory, which aims at helping project managers (PM) in the domain of engineering construction industry to create a more effective agenda for their meetings.

Specifically in the scenario of Engineers, where projects are complex in nature, and it is difficult for PM to perceive the dependencies between all tasks of every discipline, the management of meeting agenda can allow for quality improvements.

The tool proposed works in four steps:

- **Form candidate agenda:** prior to the meeting, the PM bring together all proposed topics to be discussed in the meeting, and proposes a candidate agenda. Group members are then supposed to vote for the relevant topics.
- **Follow voted agenda:** during the meeting the groups discuss only the items that received high plurality of votes. In the meanwhile or after the meeting, the PM organizes the discussed topics followed by a list of follow-up tasks and responsibilities.

- **Solicit additional agenda items:** the meeting leader asks other participants, at the beginning of the meeting, if he or she wants to raise any other issue that was previously discarded. The PM makes annotation about the discussion and the contributions.
- **Last call:** after questioning all participants individually, the leader opens the meeting to any further topics proposed by any participant. A "fee" is associated to each participant who raised an issue and to those who collaborate with him/her.

The idea is to use this payoff "fee" to align the individual with the group preferences. Along the time, participants can develop a good or a bad reputation based on this behavior of raising issues which are not relevant for the group. From the backstage structure, the PM has flexibility to choose anonymity for topics, discussions, etc. Besides, individuals can contribute asynchronously.

This system shows that even after several years of research on meetings and related concepts, there are still open problems to be solved. This reference is from 2004, and once more, it is relevant because it combines game theory and CSCW principles, like anonymity, in the meeting preparation phase, but it is limited because the support provided does not include the other meeting phases.

3.3.4 Facilitate.com

Facilitate.com (Facilitate.com, 2005) is a system that provides a virtual conference center for collaboration at any time, at any place. The system provides support for three different types of meeting:

- **Same time, same place:** offers a shared electronic flip chart (Figure 3-9), tools for consensus building, solution identification, action planning (Figure 3-10), anonymity, flexible voting tool and virtual meeting tool.
- **Same time, different place:** virtual conference center – where any number of asynchronous or synchronous meetings and surveys take place simultaneously. Participants log into their assigned electronic meeting room to discuss topics, generate ideas and prioritize solutions. While adding their own ideas, participants view and respond to contributions from the whole group creating a dynamic interchange of information and perspectives. Prioritization and categorization tools to consolidate ideas into solutions. Flexible voting tool for fast decision-making and consensus building and a complete meeting documentation is available at the end of the meeting.

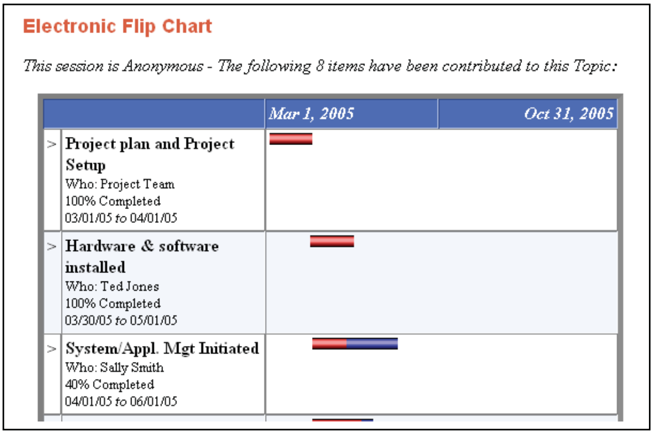


Figure 3-9: Example of the Electronic Flip Chart (Facilitate.com, 2005)

- **Different time, different place:** the virtual conference center allows project leaders for setting up an agenda of topics for discussion in a remote location. The team leader solicits input from participants, organizes the input, distributes a voting ballot, and publishes the results and requests input for the next steps. Action plans keep everyone on track. Prior to meeting face to face, the team leaders can set up a web meeting to solicit input from participants, identify the high priority items and construct a focused agenda with much of the initial information exchange completed. During the on-site meeting time is spent making critical decisions and cementing agreement on appropriate actions.

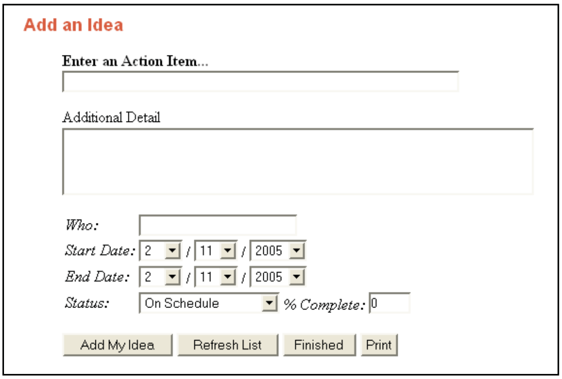


Figure 3-10: Example of Action List Generation (Facilitate.com, 2005)

Facilitate.com has an explicit component to support "Action Planning". After a meeting is considered ended, action plans can be built, with roles, responsibilities and timelines associated. Once developed, the users can update the status of the action plan on-line until it has been completed. Besides that, there is a support tool to create and execute on-line surveys and chat.

Different from SISCO, Logan or the Agenda Planning initiatives, Facilitate.com focus on the meeting phase and on a post-meeting phase with a light approach. This tool seems to be a simple solution to support some meeting situations and a first support for decision follow-up. What is mainly missing in Facilitate.com is integration to other tools.

3.3.5 GroupSystems

GroupSystems was created by a group at the University of Arizona and IBM (Nunamaker et al., 1991; 1995; 1996). The system achieved results in the industry that almost no other research initiative has achieved so far in the area of meetingware.

GroupSystems includes various smaller tools to support the meeting cycle. One of its main tools is the Agenda (see Figure 3-11). In this environment, many activities can take place in order to prepare an agenda for a future meeting. As part of this tool, users can define the items to be discussed, the restrictions and the objectives of certain discussions together with respective deadlines in order to be ready when the meeting really takes place. Additionally, the agenda allows for inviting participants, but only the meeting organizer can carry out this task. The concept of "role" is present in the system where there is at least one leader and participants' distinction.

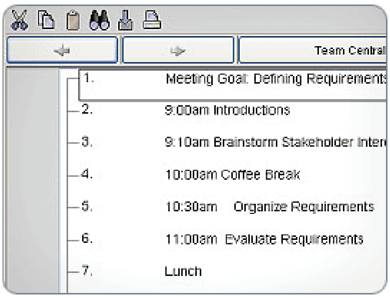


Figure 3-11: Setting up an Agenda (cf. http://www.ventana.com)

The system supports anonymity for the generation of ideas by the participants and other tools are used in a sequence such as brainstorm, whiteboard, and handouts in order to allow communication among the participants. The system also provides support for reuse of ideas through a keyword based search. The items are organized in folders (or meetings), or by topic. Each folder can contain its own discussions, handouts and whiteboards separately. A group outliner feature helps organize the ideas generated hierarchically. GroupSystems supports synchronous and asynchronous interaction and web-based support in the recent versions (url: http://www.ventana.com).

An interesting feature of this system is its historical data support, allowing for organizational memory. Ideas, discussions and all elements generated by the groups can be recovered following a specific order so that new participants can understand the evolution of the discussions. And, moreover, after finishing a meeting, the whole documentation can be used as the basis for the meeting minutes, which can be sent to meeting participants in three different formats: HTML, MS Word, and MS Excel.

3.3.6 Cybozu

Cybozu Share360 (Cybozu, 2005) is a web-based groupware composed by a set of tools that provide users with agenda, to-do lists, address book, project management, and several other features that assist collaborative work.

The Address Book tool is quite extensive and allow organizational views, personal views, web mails, and synchronization with products like Microsoft Outlook and with handhelds, e.g. with Pocket PC and Palm OS.

The Calendar tool is very rich and contains the standards features like switch from yearly/monthly/weekly/daily views of a personal schedule, view month-by-month calendars for years to come and different formats for times/dates, but also group schedules can be created and managed. The input of appointments can be done in users personal calendars, but can be classified as public or private. It also allows for schedule recurring times for individual appointments or group meetings.

Specifically about meetings, users can attach documents to meetings, such as agendas, maps, or meeting notes and find available time of others that allows viewing the calendars of all facilities and attendees while scheduling a meeting to help avoid conflicts. Meeting invitation with acceptance and decline options are also present.

An interesting and distinguishable feature of Cybozu is the Room management module. With this tool, users can manage resources, view all facilities' schedules at a glance to see what's happening in each room, view facilities from a weekly, daily or monthly view, review and reserve resources online for coordinated scheduling, and track various equipments like white boards, monitors, TVs, VCRs, speaker phones, parking spots, as well as conference rooms.

The Cabinet is Cybozu name for a centralized shared workpace repository, with support folders, sub-folders structures, notifications according to document changes, logs, notes, among other smaller features.

The Broadcasting tool allow for sending memos to different groups of people, reuse memos, search memos by keywords, users, etc. This tool also provides support for threaded-discussions.

The Notes tool offers besides the normal textual support, the possibility of sending or receiving quick messages with an easy, one-click form. This includes a distinction between private messages, receive messages via a pop-up electronic note on users home page, and receive messages via regular email or on mobile phone and pager account.

The Project Management tool includes options for creating projects, adding organized tasks under each project, adding jobs and tasks in an orderly multi-level hierarchy, save completed projects for future reference or reuse, attaching lockable document to any project or task, starting threaded discussions in any project or task, setting up private projects accessible only by the project manager and associated individuals responsible for a task and adding and update a project note on the top page of each project in order to quickly disseminate important information for the project.

From the follow-up perspective, although the system does not represent "decisions" explicitly, some track possibilities are available:

- View multiple project summaries on one page for a broad overview of all projects.
- View the progress of each project and identify potential problem areas upfront using the color-coded identifiers.
- View project status reports in the table view, adding necessary updates every day for each project. Instantly see issues/goals and project status details from everyone in charge of a task.
- Print Gantt charts.

The Project Management tool includes two other tools, one called "Time-sheet control", which includes features for tracking and visualizing time sheets in various ways and a Task Management tool that includes definition and management of tasks, with different categories, users, integration with calendar events, and different task status.

Finally, the Web-forms tool is a simple and flexible workflow solution. It allows the creation of flexible forms for various types of control. Users can define new Web-forms and create routing possibilities according to their needs. Web-forms contains some pre-defined forms for multiple approval based routings and several possibilities for management actions, like checking for the form's status, possibility of receiving alerts via email, or adding comments to each form with specific instructions, notes, etc. Users have also the possibility of viewing complete forms routes, filtered by status, category or form type.

One of the good features of Cybozu is its broad set of features, which are complementary. In the scenario of decision making processes, these tools, together, can offer means of information awareness that might have a positive impact on project management. But the explicit view on decisions is not supported by the system, and so, a clear understanding of decision implementation and follow-up is not possible.

3.3.7 RoamWare

In (Wiberg, 2001) the author presents a mobile physical/virtual meeting support system intended to support knowledge management (KM) in mobile situations, named RoamWare. One of the objectives of the project was to support the transformation from socialization to externalization, so that informal meetings could be made explicit and used in knowledge processes. The Roamware system is designed to support people who are moving around and participate in different mobile meetings. The system is intended to be used in the background while walking around, collecting data about other co-located persons who also have the system running. The idea is that the collected information will be used in later setting when the users are geographically dispersed from each other.

One of the motivations of this project is related to the fact that sometimes people cannot physically participate in a meeting. They might, for example, be too far away. Hence, RoamWare supports participation in these meetings since the physical meetings are made "physical/virtual" through a synchronous mirroring of the physical meeting to the web. An interesting concept cited in Wiberg proposes that physical/virtual should be seen as possibilities of "joining together distinct virtual and physical spaces by creating a transparent boundary between them".

This system differs from ordinary meeting support systems because it supports co-located meetings and dispersed computer-mediated meetings; it is mobile and thus supports meetings taking place where many meetings are actually happening (e.g. while walking between buildings, in hallways).

The RoamWare system consists of three subsystems:

RoamList: the key functionality of the RoamList is the automatic generation of an interaction history of participants within proximity during face-to-face meetings.

RoamWeb: the list of meetings collected by the RoamList devices is automatically published on the web as personal interaction histories. The RoamWeb system reflects the individual interaction histories built up through physical meetings onto the web as a virtual representation of the meetings held.

Figure 3-12: Examples of RoamWare screenshots running on a PDA (Wiberg, 2001)

RoamLines: this system provides the group members with visualizations of different threads of interaction among members of the mobile group. RoamLines supports querying the system about topics such as who has been to which meeting? Who was absent? Who have discussed a specific topic?

The system is a great innovative way of supporting meetings and awareness about meetings in remote sites. But as the author mentions, after two evaluations, the users complained not to be able to see all histories created by the different devices. In particular to be able to follow different threads of interaction according to different topics, different persons, different places (e.g. meeting room) and finally to be able to get notification, when a certain person has made contact with another or entered a specific place. This idea could be used in the context of decision follow-up as a support tool, integrated to a solution which supports meetings.

3.3.8 Meeting Central

Meeting Central (Yankelovich et al., 2004) is focused on improving the effectiveness of distributed meetings, meaning different people in different locations using various forms of communication.

Based on an internal study realized at Sun Microsystems, the authors compiled a list of the top problems that people have in distributed meetings. Among them were audio problems (e.g. some people could not hear, too much extraneous noise), behavior problems (e.g. speakers did not check others for understanding, meeting was not well facilitated), and technical problems (e.g. difficulty to identify who was speaking, necessary documents not available). One of the main goals in the project was to restore a small number of visual cues in order to help facilitators do their job more effectively.

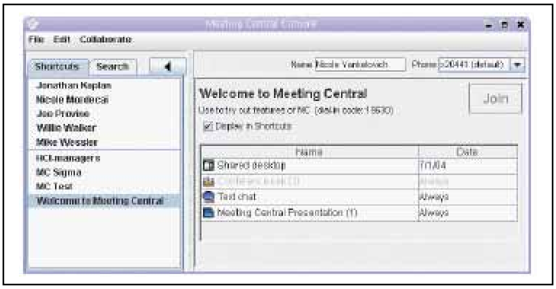


Figure 3-13: Meeting Central Console (Yankelovich et al., 2004)

With these in mind they designed and implemented a set of tools to support distributed meetings. The tools are the following:

- A Meeting Central Console: it provides a mechanism for searching and organizing contacts, and meetings. Telephone based support as part of the Console.
- Facilitator: an application which provides awareness about participants online, who is speaking, support for indicating the wish to speak next, emotional expressions, etc.
- PDF Viewer: this allows for individual views and navigation and, at the same time, a collaborative view of shared files.
- Shared Desktop: an application for sharing web pages and sharing other applications if needed.

Early evaluation of this work shows favorable comments about the whole concept and integration of supporting tools, but since this work is still in progress, the authors suggest some improvements to be done in the next versions, like joint web browsing (collaborative browser), and free-hand drawing.

This work is included in our state-of-the-art because its support on a class of meetings which is currently very common: distributed meetings. But there is not reference in this work about decisions and decision follow-ups.

3.3.9 LiteMinutes

LiteMinutes (Chiu et al, 2001) is a system that uses both the Web and email for creating, revising, distributing, and accessing multimedia information captured in a meeting. Supported media include text notes taken on laptops, slide images captured from presentations, and video recorded by cameras in the room. The main objective of this project was to improve a traditional problem of catching everything in a meeting.

According to the authors, multimedia minutes provide a rich record of what took place in a meeting. Video picks up details that are difficult to catch, captures nonverbal activity, and show context. Slides contain images, and meaningful information. Altogether, and linked, they can be used to playback interesting points of a meeting (see example of meeting minutes in Figure 3-14). In this project, the linkage is done via a parsing program that analyses the time stamped (contextual metadata) contained in the notes taken, and compare them with the video and slides files recorded during the meeting.

In LiteMinutes, notes are taking during the meetings via a Java Applet. Videos and slides are recorded and presented during the meetings. At the end of the meeting, the scribe, decides to whom send the minutes, e.g. to all participants and to the server, or if only to the application server for further revisions before sharing it will all people. Once in the server, the minutes are parsed and analyzed, i.e. the text and timestamps, together with video and slides files and linked according to their respective context. The parser also transforms the notes taken and links into an HTML format email, which is sent as minutes to the email addresses filled by the scribe. A copy is kept on the Web collection for summarized and weekly reports' views of all meetings.

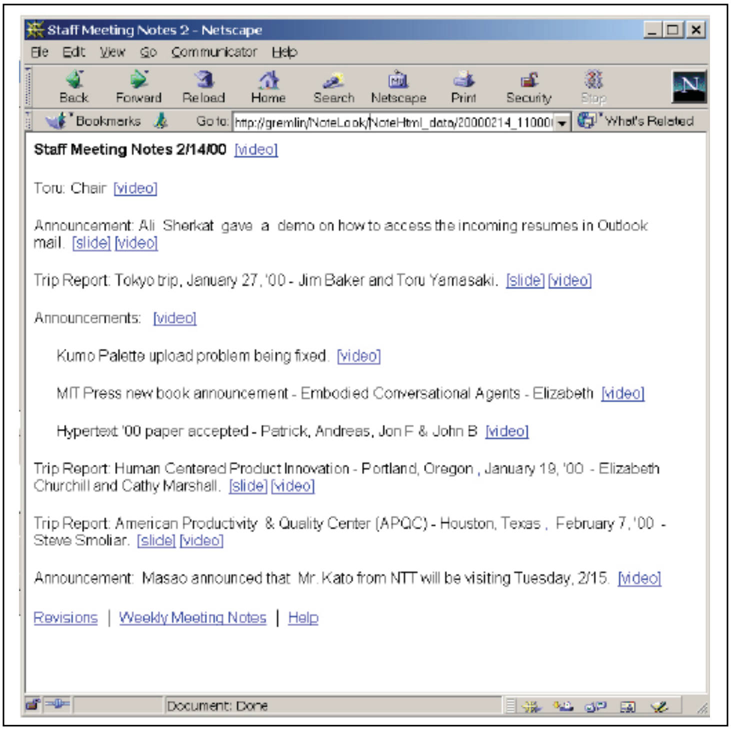


Figure 3-14: Meeting Minutes shown in a Web browser, with links to other media (Chiu et al, 2001)

After the minutes are received, the revision process can take place. The revision can be done in the users' email clients, since the format is HTML. Since the original email minutes contain internally timestamps to the meeting it belongs too, the reviewed HTML emails are also sent back automatically to the server for historical purposes and up to date views.

This system belongs to the most recent initiatives regarding meetings in the relevant literature, and involves multimedia and automated recognition of image and speech. Although it does not discuss the decision making process and the decision follow-up support, the idea of using tags or timestamps in the meeting minutes could be used to document decisions in the same way. Nevertheless, this is not discussed in this project.

3.3.10 EasyMeeting

Another recent and interesting approach was proposed by Chen in (Chen et al., 2004). This project emphasized efforts on using pervasive computing in meeting rooms. The author named it EasyMeeting. The system makes uses of FIPA agent technology, Semantic Web ontologies, logic reasoning, and security and privacy policies.

A smart meeting room system is a distributed system that consists of communities of intelligent agents, services, devices, and sensors that share a common goal. The goal is to provide relevant services and information to the meeting participants (e.g., speakers, audiences, and organizers) based on their situation conditions (or contexts). Some typical smart meeting room applications include automatic capturing and indexing free-hand sketches on a whiteboard, tracking location, providing assistance to attendees, rescheduling, etc.

The authors emphasize the importance of the “context” in the realization of a smart meeting room system. Besides location context, considered important in other works cited in the paper, the EasyMeeting considers context ontology, context reasoning and users’ related contextual information.

The implementation of the system has a dedicated module, named CoBrA—Context Broker Architecture—which is responsible for context management. The system is composed of the following modules: speech recognition service, presentation service, lightening control service, music service, greeting service, and display service.

Although this initiative does not focus on the research topics discussed in this document, we have considered important to cite it in order to show that improvements on meetingware system and improving meeting spaces with the use of technology is an actual concern.

3.3.11 Meeting Room

Meetings are a daily occurrence in any work environment. For an absent individual, there are many questions to answer about a meeting: who was involved, what was planned, what topics were covered, are some of them. At the Interactive Systems Laboratories at Carnegie Mellon University – USA (Meeting Room, 2005), they are developing a system that automatically allows individuals to browse a meeting. The “Meeting Browser” system answers many of the questions an absent individual may ask. It records meetings, automatically transcribes and summarizes them, and allows a user to search the meeting, or a set of meetings, for a particular speaker, topic or idea.

The Meeting Browser encompasses speaker identification, automatic speech recognition, language model adaptation, dialogue analysis and automatic summarization. This project holds potential for a variety of applications in academia, research and industry.

By automatically transcribing a meeting, it can provide these answers instantly:

- A user may search the transcription for keywords and topics.
- Emotion recognition provides clues as to the meeting's mood.
- Summarization offers a quick overview.
- Highlighted action items provide assignments.
- Video playback makes the user feel as if he missed nothing.

The project aims at supporting realistic meeting scenarios, with support for speech recognition problems, analysis of retrieval performance and addition of non-keyword based features, the generation of readable summaries and a practical user interface.

As already identified in previous works meeting recognition is a very challenging task. The difficulty results basically from three reasons: First, the conversational style - meetings consists of uninterrupted continuous recordings with multiple speakers talking in a conversational style. Second, the lack of training data - meeting data is highly specialized depending on the topic and participants, therefore large databases can not be provided on demand. As a consequence their research has focused on the question on how to build systems for new tasks and languages using limit amounts of training data. Third, the degraded recording conditions: to minimize interference a clip-on lapel microphone was chosen instead of a close-talking headset. However, this comes at the cost of significant channel cross-talk. Quite often one can hear multiple speakers on a single channel. For the purpose of building a speech recognition engine on the meeting task, they combined a limited set of meeting data with English speech and text data from various sources, namely Wall Street Journal (WSJ), English Spontaneous Scheduling Task (ESST), Broadcast News (BN), Crossfire and Newshour TV news shows.

Among all possible meeting scenarios, they especially targeted two different types: research group meetings and discussion-type TV news shows, where a host and several guests hold a discussion of current events.

To assist efficient reviewing and browsing meetings, recognizer output is fed to an automatic summarizer based on Maximal Marginal Relevance (MMR) criteria, and then streamed into the meeting browser system. The meeting browser interface can display meeting transcriptions, time-aligned to corresponding audio and video data. The user can choose to search, browse, or annotate the meeting.

Other than offline browsing, they are also developing an online meeting room demo, where realtime (or close to realtime) speech recognition, speaker identification, people tracking, people identification, face/gaze tracking, etc. are put together to make a live meeting scenario, so that they know the number of participants, who they are, who's talking (to whom), etc. Their hope is that by extracting and fusing additional cues they can better capture/understand the meeting dynamics and structural information.

This initiative, similar to Easy Meeting, does not focus on the research topics discussed in this document, but it shows the current interest on improving meeting support.

3.3.12 Multimodal Meeting Manager

Multimodal Meeting Manager, M4 project (M4, 2005), started on 1 March 2002, and is supported by the EU IST Programme. It is part of CPA-2: the Cross Programme Action on Multimodal and Multisensorial Dialogue Modes.

The M4 project is concerned with the construction of a demonstration system to enable structuring, browsing and querying of an archive of automatically analyzed meetings. The archived meetings will have taken place in a room equipped with multimodal sensors.

For each meeting, audio, video, textual, and (possibly) interaction information will be available. Audio information will come from close talking and distant microphones, as well as binaural recordings. Video information will come from multiple cameras. While the video and audio information will form several streams of data generated during the meeting, the textual information---the agenda, discussion papers, and text of slides---will be pre-generated and can be used to guide the automatic structuring of the meeting. The interaction stream consists of any information that can help in analyzing events within the meeting, for example, mouse tracking from a PC-based presentation or laser pointing information.

Their main objectives with this system are:

- Development of a "smart" meeting room, collection and annotation of a multimodal meetings database.
- Analysis and processing of the audio and video streams;
- Robust conversational speech recognition, to produce a word-level transcription;
- Recognition of gestures and actions;
- Multimodal identification of intent and emotion;
- Multimodal person identification;
- Source localization and tracking.

Although the technologies addressed here are not perfectly developed, they are established firmly enough to warrant their use in combination. Integration of multiple sensory inputs is a challenging problem at the early stages of investigation. Integration and structuring using the output of the various recognizers and analyses; Specification of a flexible intelligent information management framework; Models for the integration of multimodal streams, including statistical models for asynchronous multiple streams, multimodal syntax and multisource decoding; Summarization of a meeting, or a meeting segment; this could take various forms such as a textual or a set of video key frames; Multimodal information extraction and cross-lingual retrieval/browsing across the archive. Construction of a demonstrator system for browsing and accessing information from an archive of processed meetings.

The project is illustrated via the following scenario: Consider a meeting (of 3-15 people), following an agenda. Offline, the meeting is segmented into agenda items. The browser follows the structure of the agenda. Clicking on an agenda item brings up a set of ways to view that topic. These might include: a textual summary; a diagrammatic discussion flow indicating which participants were involved; a set of audio or video key frames to give the essence of the discussion. A more advanced browser might be able to identify what actions were agreed as the outcome of the discussion on that topic. It would also be possible to query the rest of the archive either by example from that segment, or through an explicit query.

Figure 3-15: M4 prototype screenshot of the system in use (M4, 2005)

The existence of the textual "side information" enables the application of some useful constraints. As a minimum there will be a prespecified agenda for the meeting that provides a basic structure in terms of a topic sequence. There may also be discussion papers or the text of slides associated with an item, which may be used to adapt the language model of the speech recognizer, or to act as query expansion information for the retrieval system. In the case where slides or other visual aids are used, the interaction stream (typically containing “pointing" information) may be used as a further constraint, giving information about a sub-topic sequence, for example.

This initiative, similar to Easy Meeting, and to Meeting Room, does not focus on the research topics discussed in this document, but it shows the current interest on improving meeting support, i.e. the focus on speech recognition.

3.4 Conclusion

Through the management of meetings, either in the pre-meeting, during the meeting or after the meeting, different alternatives were discussed and listed above. Some of them with richer support, as the case of Cybozu or GroupSystems, and others with more specific contributions, such as SISCO or the Agenda Planning Technique. Table 3-1 shows a summarized version of features supported by the systems presented.

Many interesting ideas are present in these cited references. SISCO offers a very rich pre-meeting support, and an interesting asynchronous support for discussions based on the IBIS model. LOGAN is interesting because olf its holistic approach, which supports pre, meeting and post activities. Besides, the feature of providing a chain of meetings and support for dealing with discarded topics can be also very useful for organizational memory and even, indirectly, to decision follow-up support. The same is applied to GroupSystems and Cybozu, which are more comprehensive systems that indirectly offer the basis for organizational memory, but no direct support for decision follow-up. Both are more focused on the meeting phase, while we are more interested in the post-meeting phase. The last group of systems deals with multimodality perspectives of meetings. MeetingCentral and LiteMinutes are more specific approaches with very light designs. LiteMinute introduces the context-aware support for meetings, while MeetingCentral tries to promote the possibilities for collaboration in distributed meetings. And the last systems discussed are proposing another level of support, for helping capture meeting content with multimedia and automatically generate outcomes based on that.

Only a few of them deal with decision follow-up possibilities. Most do not represent explicitly decisions made, except for the tasks or actions presented in a few of them (see gray cells in Table 3-1).

The criteria used in this table were developed by the author of this document based on the literature cited in this chapter and in previous one, with emphasis on the aspects which are relevant to our interest, i.e. decision representation, decision implementation and follow-ups mechanisms, in the context of the meeting cycle. From our understanding, the problem of decision follow-up, even reducing it to the context of only decisions made in meetings, is so complex and broad, that only one tool is not able to support all the nuances. As we mentioned previously in this document an integrated set of tools to support the whole cycle of the decision making process seems to be the best approach. However, specific tools to support each part must be proposed and integrated. The next chapter presents our proposal to deal with this problem.

Table 3-1: Comparative framework of MeetingWare

	SISCO	Logan	Agenda Plan	Facilitate	GroupSystems	Cybozu	RoomWare	MeetingCentral	LiteMinutes	EasyMeeting	MeetingRoom	M4
Other Support												
Web interface	x	x	x	x	x	X	x	x	x	x	x	x
Meeting support												
- agenda management (create, edit, etc)	x	x	x	x	x	X	x	x	x	x	x	x
- minutes management (create, edit, etc)		x		x	x	x			x	x	x	x
- allows group contribution (agenda, minutes)	x	x		x	x	x			x			
- analyse and justification of discarded topics		x			x							
- support for offline agenda preparation												x
- support for offline minutes preparation												x
- organization of topics/issues (agenda, minutes)	x	x	x	x	x	x	x		x	x	x	x
- organizational data (local, data, invited people)	x	x	x	x	x	x	x		x	x	x	x
- task/action representation (agenda, minutes)			x	x		x			x		x	x
- supports structured discussion (e.g. IBIS)	x				x	x	x					
- support brainstorming					x	x						
- support voting				x	x	x						
- repres. of the decision-making process												
- search support	x				x	x					x	x
- repres. of roles (facilitator, coordinator, etc)	x	x	x	x	x	x		x	x	x	x	
- invite/notify participants	x	x	x	x	x	x					x	x
Support virtual meetings				x			x			x	x	x
Mobile content delivery						x	x					
Provide means for decision follow-up		x				x	x					
Provide means for task follow-up		x				x						
Document Management support												
- allows documents		x			x	x		x	x		x	x
- support for different Media (video, audio)								x	x		x	x
Integration with other tools												
- integration with calendars for appointments						x		x				
- integration with to-do lists						x						
- integration with e-mails						x		x				
- integration with instant messenger						x						
- extensible or API available												
Group Support												
Awareness mechanisms	x	x				x	x	x	x	x	x	x
Synchronous support			x	x	x	x	x	x	x	x	x	x
Asynchronous support	x	x	x	x	x	x	x	x	x	x	x	x
Support for Group or Organization Memory	x	x			x				x			

4 Conceptual Design for Decision Follow Up

This chapter contains the theoretical basis and the conceptual design of our proposal. In order to provide more contexts to our description, we will first recall the main ideas that guide this research. The scope of the thesis includes the creation of automated mechanisms to allow for decision follow up, especially of decisions made in meetings. These mechanisms are related to the cycle of meeting preparation, of meetings themselves, and of the post-meeting activities. Besides proposing a structure for this cycle, we also integrate tasks and emails as the two main means to provide information about decision implementation and the respective follow up.

4.1 Proposal Context

Considering that meetings are one of the most common practices where decisions are made (Simon, 1976), (Stefik et al., 1987), (Nunamaker & Briggs, 1991) and that after them, decisions are implemented, our approach discusses the theoretical and the technological aspects of a complete meeting cycle and the decision follow up related to this scenario.

There has been much emphasis on decision support systems and group decision support systems focusing on improving the decision making process, as we discussed in chapter 2, but little attention has been paid to the implementation stage that follows a decision meeting. As a consequence to this fact, decisions are wrongly implemented or not implemented at all. Often, decisions that are implemented without the necessary follow up, may generate outcomes different from those planned at the time of the decision. Besides that, cultural barriers and the lack of appropriate tools induce just informal links. As a result, important decisions are not properly or timely implemented (Borges, Pino and Valle, 2002).

Decision meetings are not isolated events. They are part of a continuous cycle of premeeting, meeting and post-meeting activities (Oppenheimer, 1987, as cited by Bostrom, 1989). The meeting itself is the most visible part of this cycle, but the other components are always present. Making pre-meeting and post-meeting activities explicit may be the first step to enhance the whole cycle and thus, to obtain better decisions as a final result.

All the three phases can be considered equally important, since they deal with different aspects of a decision (Figure 4-1). However, only few tools have being proposed to support pre-meetings and post-meeting phases, as we discussed in chapter 3.

The premeeting phase involves the organization of a meeting: the creation of an agenda, the identification of people to be invited, the negotiations about who should participate in the meeting and their roles, the preparation of each participant for the meeting, the inclusion of background information, the discovery of material to support new proposals, the identification of new decisions to be made, and the review of pending ones (Borges et al., 1999), (Valle et al., 2002).

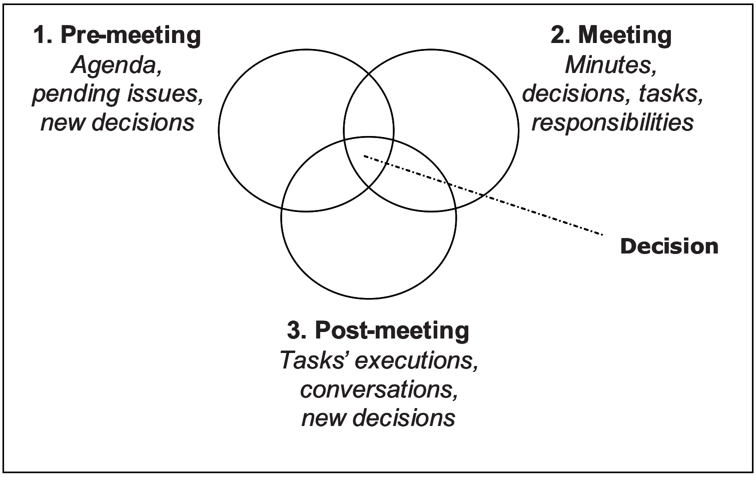


Figure 4-1: Decision role in the Meeting cycle

Meetings can be of different types, categorized by size, frequency of occurrence, composition, motivation, and decision process used (Hayne, 1999). During the meeting phase, where decisions are made, other demands are raised, different from the previous phase. For example, the recovery of historical data related to the present problems, the recovery of lessons-learned to help improve the decisions to be made, the definition of who is going to be responsible for each task; these interactions among participants may conclude with one or more decisions to be implemented afterwards.

The post-meeting phase is when the implementation of the decisions is executed. This stage contains activities to be carried out by people not necessarily present in the meetings. Its activities include dissemination, monitoring implementation of the decisions and clarification of ambiguous decision details. In this phase new demands appear like the use of past successful steps and lessons-learned about past mistakes. However, we believe that this phase also involves knowledge that can be useful in future instances of the cycle. Experiences of the implementation phase can provide indicators for new meetings, and stories about what was implemented with or without success (Valle et al., 2002). And last, but not least, new decisions may appear during this phase. These decisions may or may not be related to other decisions made, but they should be observed and checked as those related to meetings.

Looking at decision follow up in a post-meeting phase we can see different possibilities of support tools. If the organization uses a project management tool, this is the main tool where the information generated by project members is stored. From this tool, data related to decision follow up could be provided. For example, information about late deliverables, or about new decisions that have to be made, or that reallocation of resource must be made because of a certain delay in a project part, etc. If the organization makes use of a workflow, the processes instances, tasks and activities would be the elements to be looked after in order to provide decision follow up. While if the organization stores information in a document management system, documents and folders would be the main resources of information for decision follow up. The most complex case would be if the organization uses more than one tool. Again, decision follow up mechanisms would have to trace data in different formats and tools.

But there is a common denominator in this scenario, for almost any modern organization, which is one of our beliefs to provide decision follow up: the electronic communication. As pointed at the end of chapter 3, communication is one of the richest components of the decision making process. It is through communication channels that people exchange knowledge and opinions, raise problems and solve them, as well.

E-mails are one of the most successful electronic communication tools, with a stable utilization across organizations (Levitt & Mahowald, 2002). In (Wired News, 2002), the study shows that for power users, typically better educated and higher earners (managers), it is necessary to spend about 2 hours or more daily on e-mail, often beyond four, not only because of their direct participation in the communication process, but also as a person who receives copies of others' emails to be aware about on-going projects they are involved in.

Levitt and Mahowald (Levitt & Mahowald, 2002) suggests that e-mail is currently and will likely remain the dominant collaboration tool for the foreseeable future. The email or groupware "inbox" is where most of the collaborative communications, contacts, and other content are located. Any collaborative tool that is looking to build a customer base, even if someday it will surpass email in usage, will need to start by acting as an extension to email.

We believe that formalizing decisions and the tasks related to them in an electronic format should be the first step to provide decision follow up, and the second step should be the analysis of its relation to the electronic conversation that occurs in the same context, i.e. a project, in our example.

Thus, this proposal aims at building a set of mechanisms to provide decision follow up looking at task management, a more direct element for means of coordination, and at asynchronous communication, e-mails in particular.

4.2 Requirements and barriers

Based on the review of the relevant literature, for e.g. (Stefik et al., 1987), (Kraemer & King, 1988), (Nunamaker & Briggs, 1991), (Borges et al., 1999), (Hayne, 1999), (Simone & Divitini, 1999), listed in chapters 2 and 3, and on a set of 6 interviews with project managers of European and German information technology related projects (more details are available in chapter 6 and in Annex II), **we have identified the main requirements for a solution that aims at supporting decision follow up of decisions made in meetings,** see Table 4-1. Observe that this list can be further detailed, in more granular requirements. Our proposal does not include support for all these requirements. Those we considered relevant to our objectives are marked with a "✓" symbol. The criteria used to select these requirements, and not the others, or all of them, were based on reasoning what could be feasible in a determined period of time, with the resources available, and still respecting a degree of completeness to verify the ideas proposed in our research. For example, one requirement we could not consider in this thesis concerns the management and recognition of multimedia information. This requirement per se, in the context of decision follow-up, can be seen as a possibility for future work, with particularities which concern only multimedia.

Table 4-1: General Requirements List

Context	Req.	Selected Req.	Statement
Decision Representation			
	R1	✓	Decisions made should be explicitly represented
	R2	✓	Decisions should be highlighted in meeting minutes
	R3	✓	Users should be able to classify critical decision, critical impacts
	R4	✓	It should provide control points
	R5	✓	It should provide information about responsibility
	R6	✓	Decision's responsibility should be associated to one person, or to a group of people
	R7	✓	It should allow for attaching a document to a decision
Meeting Minutes			
	R8	✓	It should provide template for meeting minutes
	R9	✓	It should provide a formalization of meeting outcomes
	R10	✓	It should provide means to relate meeting outcomes to electronic communication related to decisions made
	R11	✓	It cannot be too structured
	R12		It needs to be more flexible than a form
	R13	✓	It should provide flexibility on receiving or visualizing information about decision follow up (e.g. a web-page with an overall view)
	R14		It should be result oriented and deliverable oriented
	R15	✓	It should provide different views on elements involved
	R16		It should support the storage of slides, and connect them to people (in meetings)
	R17	✓	It should enable several versions of the same meeting minutes
	R18	✓	It should be connected to "to-do" lists

Table 4-1 (cont.): General List of Requirements; the symbol "✓" represents the selected requirements included in our proposal

Context	Req.	Selected Req.	Statement
Follow up			
	R19		It should avoid follow up via email, since there is already an overload of emails
	R20		The follow up mechanism should be triggered by a button
	R21	✓	It should make use of graphical elements, like icons representing concepts and status
	R22	✓	It should provide "summaries" with what is doing well, or still open, and things done
	R23	✓	A life-cycle of a decision should be provided
	R24		It should provide reports via email if wished
	R25	✓	It should provide summarized report when wished
	R26	✓	It should provide different means of visualization when wished
Media Support			
	R27		It should support different media types
	R28		It should allow for media exchange
	R29		It should allow for video and annotation
Other Requirements			
	R30	✓	The system must not be an extra tool only to provide decision follow ups, but mechanisms that run, either with few or no interference of users, which are linked to communication tools users already make use of
	R31		It should be integrated to MS Project
	R32		It should represent decisions in Gantt-charts and Pert-charts
	R33		It should be as flexible as paper and pencil
	R34		The electronic version of the tool should be available in mobile devices
	R35		It should provide extra features: annotations, highlights, free-drawing, copying, pasting, dragging, and fast

The requirements not included in our proposal, although relevant, would involve concepts, and consequent codification that would bring too many distinct viewpoints to our research. Therefore, some of them needed to be discarded.

Besides the requirements listed above, a list of **barriers** was also elicited, which are also considered in the design of our proposal.

- Managers (or leaders) do not have structured information about decisions made in meetings. Meetings take place, decisions are made, activities after the decisions are made, but there is no automated possibility to ask for the "current status" of a certain decision made in a meeting or for "past experiences";
- Meeting documentation is not timely or accurate;
- Meeting notes are rarely the actual words of the participants;

- Meeting notes are late;
- Everyone takes their own notes, so meeting minutes are inconsistent;
- Most of the time, no action plans are developed;
- Decisions are made, but their follow-up occurs in various forms, depending on the group/organization and on the available infrastructure. But basically, communication is one of the richest elements about decision follow-ups. Documents also, intrinsically, contain decision follow-up elements;
- There is no automated mechanisms for follow up;
- Not everyone in a meeting is going to participate in the implementation of the decisions (make them reality). There is a communication problem between these two groups of people “meeting participants” and “decision implementers”;
- Post-meeting is an unstructured phase in organizations (different from meetings or premeetings) and it can be differently supported in different organizations;

4.3 Concepts

4.3.1 Relevant Theories

Since our requirements involve concepts of coordination (e.g. the meeting life-cycle, decisions’ traceability, task management), and we aim at considering how communication impacts it, we have looked at theories that were more related to these two areas: coordination and communication.

Our conceptual model contains aspects from two theories: the Coordination Theory and the Language Action perspective. Both were discussed and integrated in the implementation of the prototype and used as a basis for the evaluation analysis.

Other theories could be considered relevant and useful as well, but we decided for these two because they offer a direct map to the elements we must deal with in our system. Moreover, both are theories which were used in the fields of CSCW and Information Systems with many positive results. Some of them will be cited in the next sections.

4.3.2 Coordination Theory

Many disciplines discuss the concepts and usage of principles of coordination. Among them are Organization Theory, Economics, Linguistics, and Psychology. From the Computer Science perspective, especially in the area of CSCW, the term “Coordination theory” was proposed by Thomas Malone and Kevin Crowston, in the beginning of the 90’s, and aimed at discussing how the concept of coordination impact and should guide the development of cooperative work tools. One of their main motivations for this “theory proposal” was the wish to understand how the widespread of information technology change the ways people work together.

For Malone and Crowston (Malone & Crowston, 1994) **coordination can be seen as a process of managing dependencies among activities**, and as a counter intuitive example we can say that if there is no interdependence, there is nothing to coordinate. Moreover, further progress should be possible by characterizing different kinds of dependencies and identifying the coordination processes that can be used to manage the activities.

4.3.2.1 Used Concepts

Coordination, according to these authors, involves four main elements: goals, actors, activities and dependencies (Malone & Crowston, 1990). However, their main emphasis is on the concept of "dependency". Based on it, the analysis of different kinds of dependencies, the coordination processes associated to each of them, and how they impact the design of information technology, can be done. They propose a list of coordination processes for managing dependencies and how coordination processes could be used to manage them (Malone & Crowston, 1994).

In particular to our thesis, since we are more focused on CSCW, we will list only those that are relevant to our scope (Table 4-2; cf. for 1st and 2nd column (Malone & Crowston, 1990); 3rd column added by this author). In the first column the coordination processes proposed in the Coordination Theory are listed, the second column shows a list of groupware which were based in these types of processes (note: some include support to more than one coordination process, and use concepts of other theories, e.g. Coordinator with "Language-Action Perspective"), and the **third column** lists the techniques for coordination associated to each process, which are included in our proposal.

Table 4-2: Coordination processes, examples of systems and relevant techniques

Coordination Process	Groupware Examples	Techniques relevant in our proposal
Managing shared resources (including task assignments)	Coordinator [1] Information Lens [2] CHAOS [9]	Allocating actors' time to the tasks to perform Tracking
Managing producer/consumer relationships (including prerequisites constraints and usability)	Polymer [3]	Standardization (usability) Notification Tracking
Managing simultaneity constraints	Meeting scheduling tools [4]	Scheduling Meeting scheduling Synchronization
Managing task/subtask relationship	Polymer [3]	Goal selection Task decomposition Negotiation
Group-decision making	gIBIS [5] Sibyl [6] Electronic Meeting Rooms [7]	Alternatives: by manager, by consensus Task decomposition Tracking
Communication	Electronic Mail Computer Conferencing Information Lens [2] Electronic Meeting Rooms [7] Milan [8] CHAOS [9]	Asynchronous (Email only) Standards for communication (conventions) Means to create common knowledge

Reference list for Table 4-2:

[1] – Winograd & Flores, 1986;
[2] – Malone et. al, 1986a;
[6] – Lee, 1990;
[7] – DeSanctis & Gallupe, 1987;

[3] – Croft & Lefkowitz, 1988;
[4] – Beard et al. 1990;
[5] – Conklin & Begeman, 1988;
[8] – Michelis & Grasso, 1994;
[9] – Simone & Divitini, 1999;

Using the principles of Coordination Theory does not mean that we encompass all possibilities of coordination processes and techniques. Many other processes and techniques, not listed in Table 4-2, can be identified and used in other scenarios, which are different from ours. Moreover, our proposal, although it includes various aspects of coordination, in general, takes into account techniques that are more relevant to solve the problems and issues of our particular scenario. Additionally, priorities on which level each of these techniques will be implemented or not, are part of a natural restriction that we need to include in the proposal as well.

4.3.3 Theories of Language

The work of J.L. Austin in (1962) was the basis for many research areas related to the study of language, and in our specific case, the use of language in CSCW and IS. The area also received contributions from Ludwig Wittgestein, a philosopher, and a phenomenologist, Adolf Reinach at the beginning of the 19th century. Especially in the community of IS, a student of Austin, John Searle, is very much associated with this theory (Searle, 1969).

According to Searle "speaking a language is performing speech acts (…) in accordance with certain rules for the use of linguistics elements", and the minimal unit of an utterance is not a word or a sentence, but a speech act (Schoop, 2001). He classified speech acts in locutionary and illocutionary acts. An illocutionary act is a complete speech act: uttering words, predicating and referring in a particular mode, e.g. stating, requesting, promising. He proposed five classes of utterances according to the illocutionary point: assertives, directives, commissives, expressives, and declaratives.

The use of speech related theories in IS and in CSCW has been investigated over the last decades, mainly after the contribution from Winograd & Flores, (Winograd and Flores, 1986), named Language-Action Perspective, and the work of Jürgen Habermas, named "Theory of Communicative Action" (Habermas, 1984). Many initiatives used different aspects of speech act theory in IS and in groupware: Coordinator (Winograd & Flores, 1986), SAMPO (Auramäki et al., 1992), Cosmos (Bowers & Churcher, 1989), Action Workflow (Mora et al. 1992), Milan (Michelis & Grasso, 1994), CHAOS (Simone & Divitini, 1999), Negoisst (Schoop et al., 2003) are some of them.

4.3.3.1 Used Concepts

The Language Action Perspective (LAP) argues that human beings are linguistic beings and act through language. LAP emphasizes what people do while communicating, how language is used to create a common reality for all communication partners, and how their activities are coordinated through language.

Language, according to LAP, is seen as action, following the basis of speech-act theory, emphasizing the act of language rather than its representational role, based on descriptions only.

Two research questions are present in this perspective: (i) “how does an utterance have meaning?” and (ii) “what kinds of actions do we perform in language?”. Language, according to the LAP definition, is not only used for exchanging information as in documents, statements, but also to perform actions, e.g. promises, orders, declarations (Schoop, 2001).

There are a number of basic **assumptions** underlying LAP:

- The basic unit of communication is a speech act
- Natural language sentences correspond to the performance of speech acts
- The meaning of sentences can be revealed by specifying the speech acts that have been performed
- Speech acts obey socially determined rules
- Cooperative work is coordinated by the performance of language actions which are speech acts

The theory is composed by four basic **elements** (Mora et al., 1992):

- **Language acts:** classified according to a speech-act
- **Conversations**: which are coherent sequences of language acts with a regular structure of expectations and completions
- **Time tokens:** for completions in conversations
- **Explicit mutually-visible representations of acts:** conversations, and times, as a way of facilitating communication in an organization

In our proposal we aim at investigating the use of language and conversations in the scenario of meeting life-cycle and mainly, in the decision implementation, i.e. in a coordination scenario where cooperative work must also take place. We want to analyse how asynchronous communication in the context of a project, where decisions are being implemented, influence the course of the implementation. The concept of “conversation”, proposed in (Winograd & Flores, 1986) is, then, applied in our proposal.

The Theory of Communicative Action (Habermas, 1984) uses Searle’s theory as a basis, but for Habermans, the illocutionary force establishes the mode of communication between the *speaker* and the *hearer*. The propositional content establishes the relations between the utterance and the outside world.

Habermas distinguished between three-world concepts (Cecez-Kecmanovic & Janson, 1999):

- o <u>objective world (external reality):</u> “the totality of what is the case” – which true propositions are false. An actor may attempt to change an existing state of affairs and can succeed or fail in doing so.
- o <u>social world (normative reality):</u> “normative context that lays down which interactions belong to legitimate interpersonal relations”. The social world embodies moral practical knowledge in the form of norms, rules and values.
- o <u>subjective world (internal reality)</u>: “as the totality of subjective experiences to which the actor has privileged access”. By uttering experiential

sentences, desires and feelings, an actor makes his subjective world known to the listeners who may trust or distrust the actors' sincerity and truthfulness.

According to the reference, Habermas suggests that the objective and social worlds are external to an actor; the subjective is not. These three-world concept forms a reference system for actors in social interactions. The three worlds and the rational relations of actors to these worlds form a framework within which the classes of social actions are defined (Cecez-Kecmanovic & Janson, 1999).

In our coordination scenario, which relies on group work, this distinction of the three worlds suggested by Habermas is relevant. We support these ideas through the structure of the whole meeting cycle that guarantees an association with the objective world; the social world is represented through the roles and through the standards the groups define in a project context, for instance through conventions on the way they communicate; and the subjective world is represented by the options a user has to express herself via asynchronous communication. One of our main claims is the reinforcement of the subjective world, which is probably, on one hand, the most difficult one to be represented in a system, but which, on the other hand, deals with natural way users work.

Habermas also proposed three types of actions (Cecez-Kecmanovic & Janson, 1999), (Schoop, 2001):

- **Instrumental actions:** actions that have effect on events and state of the world. Actors relate to the objective world in order to achieve their goals by manipulating and controlling objects in their environment (material, financial). By using technical rules, actors calculate alternative means and select the one that maximizes their chances of achieving the desired goal.

- **Strategic actions**: based on rational rules, and try to influence a rational opponent. Actor pursues a goal by influencing the behavior of other actors (opponents) following rules of rational choice. Action's success is measured by efficiency in achieving the desired intervention in this environment.

- **Communicative actions:** motivation is a rationally grounded consensus between the communication partners, who pursue goals by achieving shared understanding and coordinating their plans of action. Actors come to understanding with one another by negotiating definitions of a situation, argumentation and cooperative interpretation of events, goals, values and norms, and by sharing their supportive experiences, desires and feelings. Communicative action is successful to the degree that agreement is cooperatively achieved and that individual plans of action are coordinated. Communicative action exemplifies the concept of communicative rationality inherent in human speech, which denotes a communicative practice characterized by actions and obligation to give reasons for or against validity claims raised, to challenge, accept or reject claims of others on the based of the better argument. Participants engaged in communicative action are in a process of continuously disputing and resolving validity claims while temporarily agreeing on something.

Habermas also proposed two **additional concepts** to classify social actions:

- **An orientation to success:** where actors are interested in achieving a desired state of affairs in the objective world.
- **An orientation to reaching understanding:** where actors seek a communicatively reached agreement by relating to the objective, social and subjective worlds. Actors in this case, have their own goals, not only one shared goal, but harmonize their plans/goals of action based on normative agreement and a common definition of the situation.

He also distinguishes the reactions from the **hearer,** who can understand the utterance, say yes or no/accept or reject, or accept and commits. And from the **speaker** perspective, he distinguishes the validity claims as follows:

- Utterance is **comprehensible**, and if not, the speaker must explain it.
- Utterance is **true**, and if not, the speaker must justify it, supply more info.
- Utterance is **truthful**, and if not, the communication can stop, or evolve for a new trustful situation.
- Utterance is **appropriate** to normative context, values or standards. This is suggested to be solved by looking at norms, standards, literature, common experiences.

The principles of the Theory of Communicative Action and LAP can be mapped to some parts of our proposal. Some concepts are applied internally in the coordination structure, e.g. in the negotiation for task execution, and some are applied to external elements, e.g. the relation between emails and decisions, which will be detailed later in this chapter. Since we are proposing that, together, coordination, cooperation integrated to communication, can bring other levels of understanding (i.e. about decisions being implemented), we predict that assignments, negotiations and clarification will be necessary to be supported, and we decide to support that through communication elements. Thus, our proposal touches the grounds of the three different worlds of Habermans theory (objective, social and subjective), in order to achieve goals of people who belong to groups (e.g., a project team). Our proposal also uses the basis of conversation from LAP.

4.4 Conceptual Model

This section describes our conceptual model. First we will define which coordination elements compose the model and how they are related to each other (Figure 4-2).

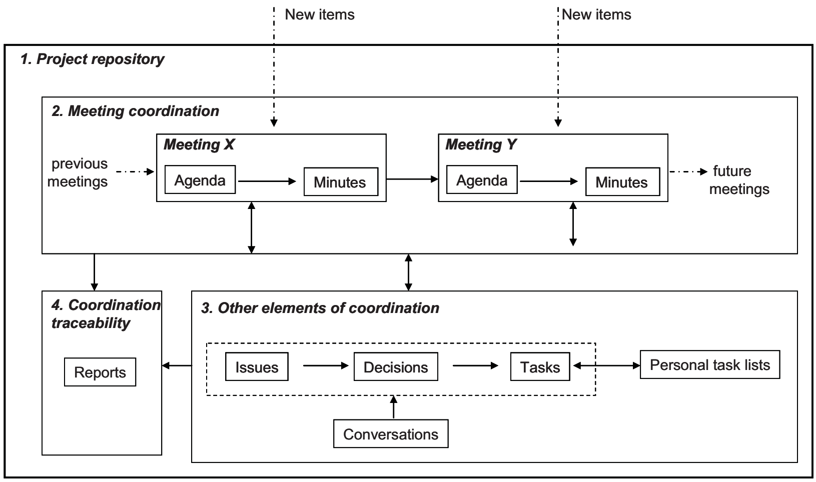


Figure 4-2: Coordination Conceptual Model

The model is composed of four different coordination elements, which are related to each other:

1. The **project repository** is the component that includes all the others. For example, the different meetings cycles, the traceability of decisions, the management of tasks and the reports used for traceability, all of them, take place in the context of a project. A "project" is the metaphor used for organizing all the coordinated elements, including decisions.
2. The **meeting coordination** component deals with two phases of the meeting cycle: (i) the pre-meeting—represented through the meeting agendas and (ii) the meeting itself—represented through the meeting minutes. Agendas provide the basis for Minutes, represented by the arrows between each of them. And additionally, agendas and minutes items can be originated from three different sources: (a) the outside world, representing new items, i.e. topics that are entering the project context for the first time, (b) from previous meetings documentation, i.e. allowing links between different meetings and (c) from the project repository element 3, i.e. issues, decisions, tasks and conversations, which are the basis for the decision follow-up idea. This flexibility allows for links between all these elements, which are normally done informally when project people are organizing meetings without using an approach like the one proposed in this research.
3. The **other elements of coordination** represent the concepts of issues, decisions, tasks and conversations. Issues represent the highest concept, which can be related to several decisions. A decision can be, or not, related to an issue, and can have several tasks associated to it. A task is dependent on a decision and can have a project representation as well as a personal task representation. These three elements together form the core concepts that provide the means for decision

follow-ups. Each feedback given to any task, at the personal level, automatically updates the associated project task, in a cascade effect, giving the basis for a decision follow-up. The same idea is applied for a conversation. Conversations can refer to any of these elements at the project level. The task feedback and the conversations are part of the post-meeting phase in the meeting cycle.

4. The **coordination traceability** element represents the possibilities of visualizing the relationships between the elements discussed in 1, 2 and 3, altogether, through the use of text and graphic reports. The idea here is to obtain in an automatic view the understanding of the status of a decision made in a meeting. Additionally, other levels of report are included to allow for various levels of traceability in the context of a project.

In the next sessions we discuss each element in detail.

4.4.1 Project

A project, in our proposal, is the concept that represents the higher level of coordination. It works as an "umbrella", where all other things take place (i.e. meetings, conversations, decision implementation). A project can represent a big group of people working for a couple of years with several goals to be achieved, taking the European research projects as an example, or a small group of people who needs to coordinate and collaborate with each other to achieve one specific goal.

Although there is a margin for flexibility in the interpretation of what a project is in different contexts, our proposal considers some baselines. A project must have a name and a description associated to it, a list of project members that can change over time, a start and an end date.

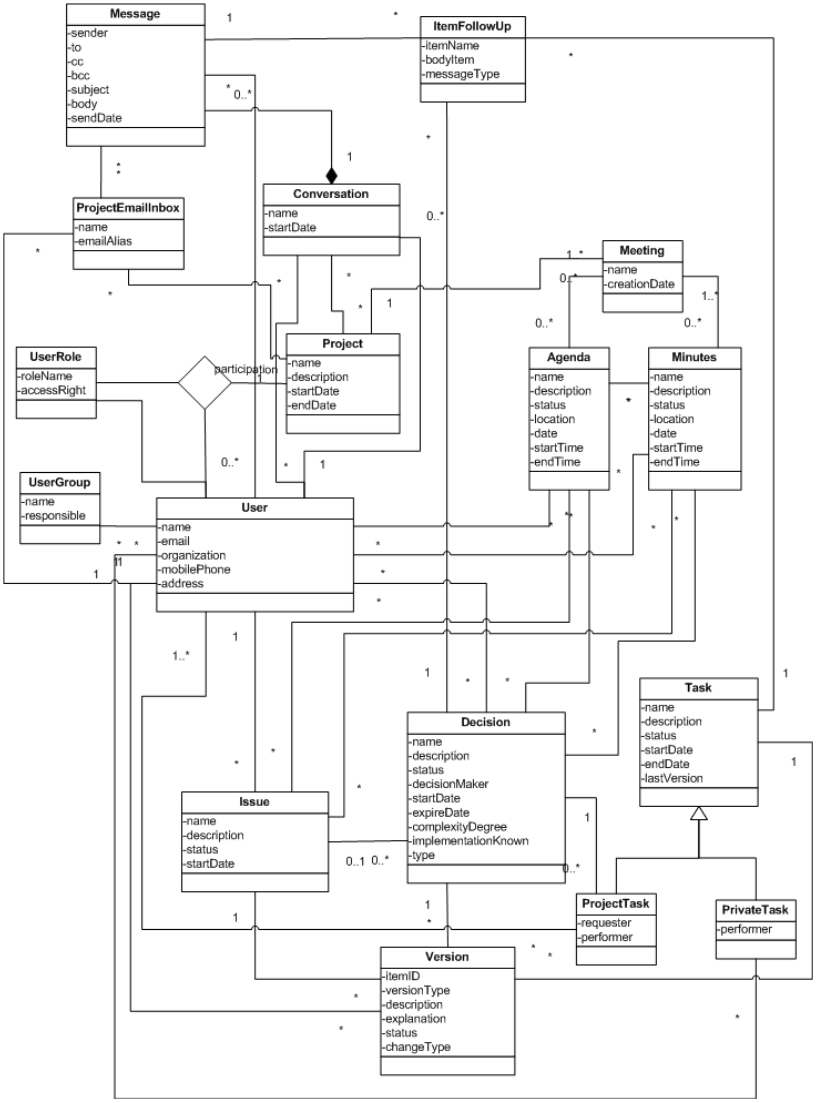


Figure 4-3: Class Diagram of the overall Project Concept

Our concept of "Project" works as the main repository, where the other elements are created, stored and interrelated. The holistic view of all these elements together is one of the contributions of our proposal, with a highlight to decisions made. Figure 4-3 shows all classes involved in our proposal. In the next sections we present detailed discussion about each relevant part. The diagrams follow the definition of the UML (Rumbaugh et al., 1999).

4.4.1.1 Group support and access rights

Since our proposal deals with coordination in projects, realization and documentation of meetings, and the execution of tasks, there are a number of users who can make use of our system, and who can benefit from its assistance, although our main focus is on the role of a "project manager". In our conceptual model we support the concepts of users, group of users and user roles (Figure 4-4). They are used in various parts of the system for different purposes.

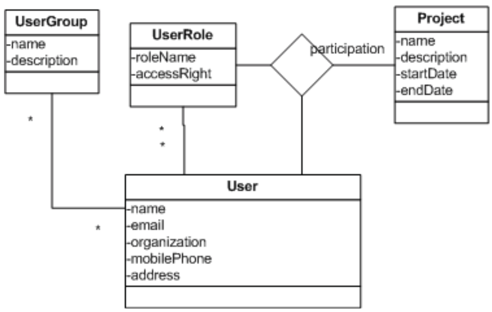


Figure 4-4: Class Diagram of User Management in the Project Context

The first group of users is composed by the project members, including the project manager. They are able to access every element under the project hierarchy, except if the project manager determines areas of restricted access, for instance, for a selected group of reviewers.

In the particular management of agenda and meeting minutes, the access rights takes place in a special way. At the time of creation, only the user who is writing down the information can see the element being created (agenda or minutes). And only after this user publishes the agenda or the minutes, the users selected as invited meeting participants are able to access and change the stored data with their contributions. The other project members are only able to see the existence of data, but they are unable to change it.

In the particular case of task assignment and execution, task performers might not be project members, and even though, be able to receive task assignments.

In the context of issues and decisions, owners and assigned people (e.g. a sub-group of project leaders) can create and change these coordination elements from each other. Other project members can only see the existence, but not change elements of other project members.

4.4.1.2 Awareness

Awareness can be defined as an “understanding of the activities of others, which provides a context for your own activity” (Dourish & Bellotti, 1992). Awareness involves the understanding about the activities and contributions already executed by related people, the current status as well as future situations.

There are different types of awareness proposed in the literature. Group awareness (Begole et al., 1999), workspace awareness (Gutwin et al., 1996) and ubiquitous awareness (Gross, 2001) are some of them. And they are more or less adequate depending on the context, and objectives they are used for.

As pointed in (Borges et al., 1999), awareness mechanisms are essential for group support systems in order to transform irregular interactions of group members into a consistent and perceptive performance over time. Awareness mechanisms are important to keep members up-to-date with important events and therefore to contribute to more conscious acting on their part.

Along the concepts discussed in the next sections, we include a couple of basic awareness mechanisms (e.g. colors, icons and notification) to support the follow up of decisions to those who are involved or interested in obtaining information about decisions made.

4.4.2 Issues, Decisions and Tasks

These three elements together form one of the innovative concepts of our proposal. Their existence aim at modeling the steps involved in the solution of problems in the context of projects. One issue can involve several decisions, which, in order to be accomplished, may require a set of tasks to be executed. Issues and decisions can also be created independently, and have no consequent elements, but project tasks are always associated to decisions in our model. However, if they are explicitly connected, what happens in one level is reflected in the other, i.e. their management follows a cascade effect. For example, if an issue is cancelled for any reason, all decisions and tasks, which are related to it, are automatically cancelled.

They are managed in different repositories, under the project hierarchy, and they can also be created in the context of meetings. They are accessible in both contexts (i.e. from their own repositories and from the meetings repositories).

Since meetings are only one possibility where issues are discussed, decisions are made and tasks are assigned, we decided to have these three elements also represented outside of meetings to allow for a more realistic approach.

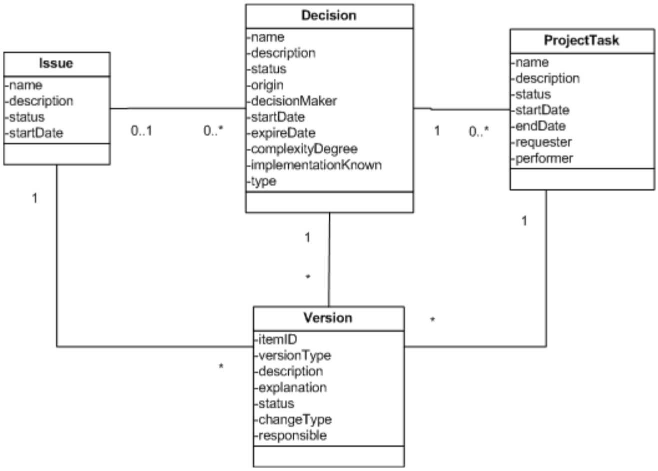


Figure 4-5: Class Diagram of Issues, Decisions and Tasks

The relationship among these three elements allows for automatic coordination mechanisms, and enables the storage of historical data of problem solving scenarios.

The awareness about a status of an issue, of a decision or of a task, can be done in the context of each of them, i.e. the user of our system should have the possibility of looking only at one these elements separately, or seeing it in the context of the others, if there is an explicit relationship between them.

4.4.2.1 Issue

Issue, according to the Merrien-Webster dictionary (Webster, 2005) is: (i) matter that is in dispute between two or more parties or (ii) a vital or unsettled matter (iii) the point at which an unsettled matter is ready for a decision.

“Issues” have been a matter for many discussions and research initiatives in the field of CSCW and IS, through the use of Issue-Based Information Systems (IBIS). The work of (Kunz & Rittel, 1970), (Conklin & Begeman, 1988) and (Borges et al., 1999) are some examples. These concepts are meant to support coordination and planning of decision processes. IBIS guides the identification, structuring, and settling of issues raised by problem-solving groups, and provides information pertinent to the discourse. Systems based on the IBIS model normally contain elements like issues, questions of fact, positions, arguments, in order to model a problem (Kunz & Rittel, 1970). The main idea behind the IBIS model is to represent the rationale in a decision making process.

In our conceptual design, project issues are all relevant topics discussed or to be discussed, which may or may not have decisions associated to them, and are worth being documented. Issues exist independent of other project elements, but inside of a Project context, as part of meetings, either in the agendas, or in the meeting minutes.

An Issue has a name, a description, a start date, a creator, and allow for associations with documents and conversations. An issue has its own life cycle (Figure 4-6):

- Open - where an issue is just created
- Innactive and active - where decisions and tasks are created for an issue
- Suspended - where an issue is postponed (can be resumed - back to decided/open)
- Closed - the issue does not need actions to be done
- Cancelled - where the issue is cancelled
- Completed - where the issue is accomplished
- Archived – used for historical data

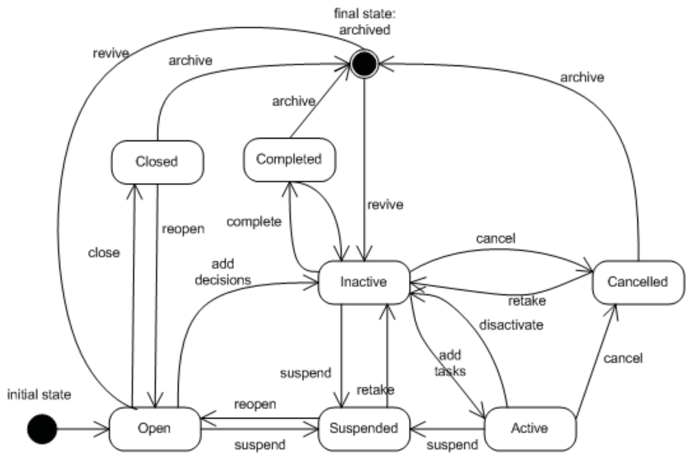


Figure 4-6: Issue State Diagram

Issues, like decisions and tasks, in our conceptual design can be traced, so that project members can accompany their development and interfere at any moment where a problem is encountered, aiming at improving the coordination activities.

4.4.2.2 Decisions

The concept of Decision is one of the most important one in this proposal. Different from many other initiatives cited in chapters 2 and 3, we aim at making decisions explicitly represented and stored in our system.

Recalling the definition mentioned in chapter 2, a decision is typically made to achieve a *goal*, to solve a *problem*, or to implement a *plan* (Balasubramanian et al., 1999). Decision is a report of a conclusion, and in our case, it can represent the conclusion of a group. Project decisions are all decisions made, which may have tasks associated to them or not. A decision can be created, and exist independently of any project element (except the project itself), or be part of meetings, either in agendas, or in minutes.

After a decision is created, tasks, conversations can be associated to it. According to the task management results, or to the influence that conversations have on it, follow ups are provided. The trace of a decision will then contain information on how decisions evolve, following the natural evolution of the tasks and conversations.

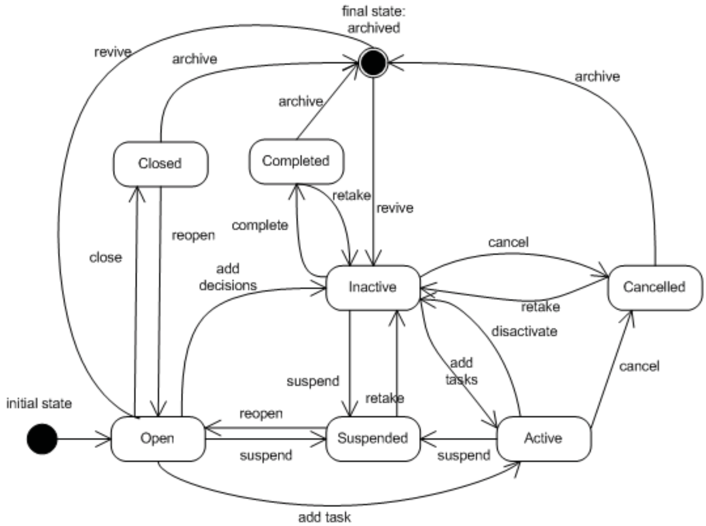


Figure 4-7: Decision State Diagram

A decision has its own life cycle (Figure 4-7):

- Open – represents the time when a decision is created.
- Innactive and active – represents the time when tasks are added to a decision.
- Suspended – represents the situation when a decision is postponed for a while.
- Closed – represents the situation when no actions are required.
- Cancelled – represents the situation when a decision is cancelled.
- Completed – represents the time when a decision is accomplished.
- Archived – represents status of historical data.

The only difference between an issue and a decision state diagram is in the activity of adding tasks. In the issue, a decision must be added first (transition from open to inactive in the issue diagram), and then a task can be added (transition from inactive to active), while in the decision contexts, tasks can be directly added (transition from open to active).

A decision may represent the statement of one individual (e.g. from the project manager) or the consensus of a group of people. In this thesis, we do not support the phase when a decision consensus is made, i.e. the decision making activities themselves. Our automated support starts at the moment an individual or a group have come up to a conclusion about a topic, made a decision and will document it. Our support also allows for storing decisions to be made in the project context, but once more, no support for simulations, brainstorming, voting, or similar mechanisms are part of this thesis.

A decision has a name, a description, responsible part (one or many), a start and an expire date, and may have tasks, documents and conversations associated to it. Milestones or critical points are also associated to a decision, in order to facilitate the coordination and the review processes.

Besides that, decisions can be classified (optionally) by several dimensions:

- **Complexity degree:** complex, medium or simple. Where complex should map decisions with high risk, involving many tasks or critical tasks, and many group interactions. Medium should map decisions which combine complex and simple tasks and group interactions. Whilst simple should map only ordinary decisions, that might not need to be traced.
- **Predictable implementation:** yes or no. This information can work as a "flag", i.e. a sign to help managers deal with uncertanties. "Yes" should map situations where previous experience indicate that an implementation is already know, while "No" refers to situations that project members are facing for the first time.
- **Decision type:** new (e.g. new product), provocative (e.g. new idea in a project), recurrent, i.e. similar to previous decisions made.

The combination of these attributes indicates decisions that must be traced, or not. They aim at helping the coordination to focus on the most important items in the project, based on the critical level of implementation.

4.4.2.3 Tasks

A task is a mission, an assignment, an obligation. Inside of computer science it is common to define a task as a logic unit of work, or a process that cannot be subdivided any further, i.e. an atomic process.

In our proposal, a task follows the idea of "unit of work" and contains a name, a description, a start date, and end date, a requester and a performer, and it allows for associations with documents and conversations.

A task has its own life cycle (Figure 4-8):

- Created – refers to the status when a task is new in the system, contains a performer assigned to it, but it was not accepted yet.
- Accepted – refers to the status when the performer accepts to execute the task.
- Confirmed – refers to the status when the performer confirms the acceptance.
- Declined – refers to the status when a performer does not accept it, providing a reason why, and the task is "sent" back to the requester in order to either start a negotiation process or to forward the task to a different performer.
- Reproposed – refers to the status when the requester decides to repropose the same task to the same performer. We support the idea that in this new proposal some parameters must have changed in order to provide the means for an acceptance.
- Cancelled – refers to the status when a task became unnecessary and is removed from the list of active tasks. This action must be done by the requester.

- WithDrawn – refers to the status of suspending a task.
- Completed – refers to the status when a task is finished. A report should be provided in this case, stating relevant information about the task accomplishment.
- Archived – represents status of historical data.

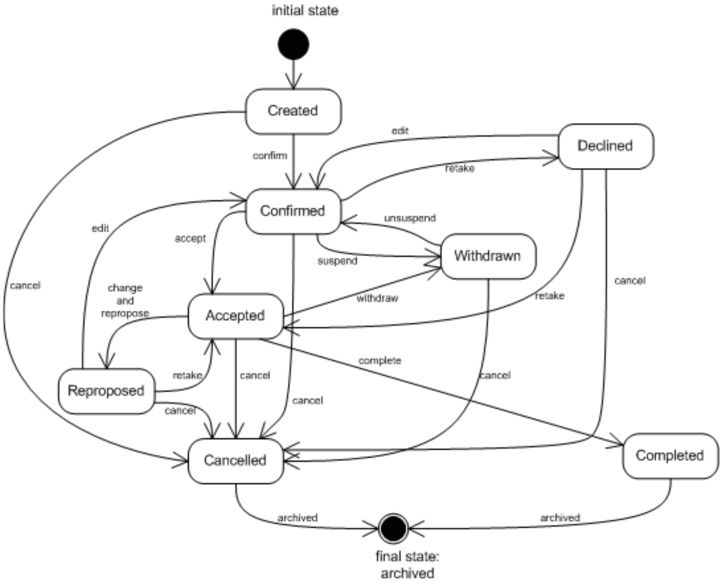


Figure 4-8: Task State Diagram

Besides these formal states that a task can have in its life-cycle, two other states, which are dynamically interpreted are the following:

- Edited – refers to a task that has been changed. For each change (e.g. description, end date) a justification can be provided by the person who is executing the change. This information aims at providing an option for documenting changes.

- Late – refers to a task that should be completed by a date and it is not. In other words, the end date has arrived, and the status is still not set to complete. This concept is very used in the reports for awareness purposes.

Some of these operations are realized by the performer and others by the requester (Table 4-3).

Table 4-3: List of actions for the negotiation actors

Requester	Performer
Confirm task	Accept task
Edit task	Decline task
Cancel task	Complete task
Withdraw task	Update task
Unsuspend task	Resume task
Archive task	Repropose task
	Retake task

4.4.2.3.1 Task Management

Project tasks are always associated to decisions. A project task can be created after a decision is made, in the context of meetings or not, but under the project structure.

In our concept there are two ways of creating tasks:
- **Case 1 - context of meeting minutes:** at the project perspective, a meeting minute is prepared by an annotator who writes down the output of each "issue", of each "decision" made in a meeting. This person classifies if there is a task or a set of tasks associated to the respective decision, using the opportunity of group interaction to assign task end dates, descriptions and candidate to be the performers. And this is exactly the situation when the most interesting scenario occurs, because this relation (i.e. one issue, one/many decisions and one/many tasks) would create the opportunity for "follow-up" of "decisions" looking at the "results of the tasks" associated to them.

- **Case 2 - personal task list:** at the individual perspective, when a user creates simple private task in her task list. There is no "decision" follow up in this case, but the reports (e.g. visualizations) and actions (e.g. stating that a task is complete) created for "case 1" should be allowed in this "case 2" scenario as well.

In both cases, there are **two main actors (roles)** involved in the task management: the **requester** (i.e. the person who creates a task and assigns it to a person) and the **performer** (i.e. the person who implements the action in the task, who executes the task itself). In "**Case 1**" the requester and the performer are probably different people (e.g. a project manager and a programmer respectively), whilst in "**Case 2**" the requester and the performer are probably the same person.

For the "Case 1" scenario, the visibility of task assignment follows the visibility of the minutes, i.e. while the annotator is elaborating the meeting minutes, no assignments are finalized, no one is able to see the minutes "draft", only the owner of the minutes. More details will follow below when we discuss the meeting phase. But once a minute is finalized, i.e. considered ready to be shared, then all tasks that were defined as outputs of "decisions" are automatically sent to each "responsible task-list" and to a general "project task-list". After tasks are created, they can be managed in the two contexts as well.

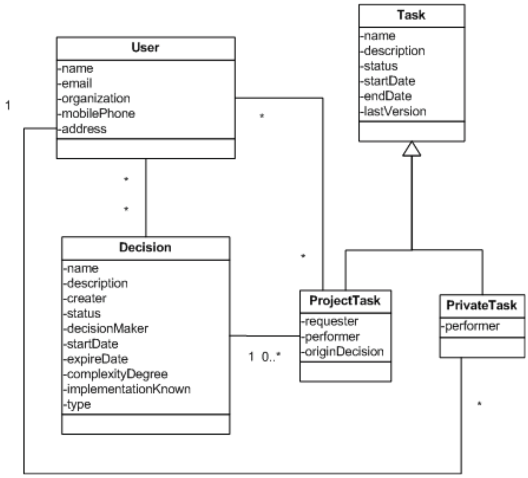


Figure 4-9: Class Diagram of Tasks in the Project Context and Private Tasks

In the **Project Task List**, users have the possibility of checking the status of tasks through different visualization reports:

- User visualizes all tasks by each meeting, by different performer
- User visualizes tasks status. The late and pending status are shown with an icon or coloured text as a mean for awareness.

In the **Personal Task List**, the following operations and options are supported:

- User creates a task
- User accepts a task assignment
- User rejects a task assignment
- User accepts a task change
- User rejects a task change
- User proposes a change (e.g. description, date, responsible)
- User sets task status to complete
- User sets task status to withdrawn (suspending it) and mandatorily state the reason why

Here some basic filters are provided to help users distinguish private tasks from project tasks, and tasks from different projects.

Besides the visualization and evolution of a task, changes on data might also be necessary. For the task requester, she will have the possibility of changing any task in the **project task list.** For the task performer she will have the possibility of changing any task in her own **personal task list.** For **any other user**, they will be able only to "read" the summary of the meeting minutes and see which tasks are included, or check for the project task list, without having the permission to change any data. Changes are indicated in the "edited" status of a task.

4.4.2.3.2 Task Negotiation

In order to accomplish a task execution in many occasions, a negotiation between the performer and the requester is necessary. For conceptualizing our elementary negotiation support mechanism in the task management context, we use the basis of the study of Habermas for the means of negotiation.

As mentioned in (Schoop, 2004), electronic negotiations have focused on automation of processes. But we agree and support the statement of this author when she says that a communication perspective on electronic negotiations need to be supported in order to achieve the aim of enabling even complex negotiations electronically.

In our scenario the main goal is to provide means for coordination, and to enable project team members to follow the steps of decisions made. Since tasks are one of the means for that, the second is messages (detailed in the following sections), it is very important that tasks also evolve, so that decisions can be finally implemented, i.e. the main objective when a decision is made.

Once a task is created, the system sends automatically a task to the performer personal task list. However, in many situations, the performer might not be in accordance to what was assigned to her, or not able to commit to it. Examples of these situations vary from simple changes in the input parameters (e.g. description of what has to be done) to a negotiation for a new end date, or even to forward a task to another person because the original performer is unable to execute it.

According to the literature on negotiation (Schoop, 2004), our approach is following the principles of:

- *Integrative negotiation* – since we aim at supporting the scenario when both parts of the negotiation process can be "satisfied" – also known as "win-win" approach.
- *Prescriptive negotiation* – since we aim at supporting negotiators achieve good results by providing means for clarification on how to do it.
- *Mix between norm-based and goal-based negotiation* (Weigand et al., 2003) – since we aim at supporting negotiations of tasks which are: based on commitments from negotiation steps (meetings) and expressed by a higher goal, the decision associate to it.
- *Asynchronous* – since we aim at supporting the negotiation via asynchronous exchange of information and communication.

Using a different terminology for task status, discussed above, our task management mechanism supports the seven messages types based on Searle's classification of speech act: offer, request, counteroffer, accept, reject, question, clarification. The message type is automatically set to the task's status, according to the users' reaction on the tasks assigned.

4.4.3 Conversations

What we could observe in our interviews, added to our own experience in European research projects, as well as in the review of the literature (chapter 3), and is the great importance of electronic communication for distributed groups working on a project. Even sounding obvious what we have just said, it is a fact that most of the time, there is a lack of linkage between electronic communication elements and other project elements. Naturally, with the advances of Artificial Intelligence and Information Retrieval nowadays, many links can be automatically generated. Nevertheless, there is still space for improvement and discussions about the role that this linkage has.

According to (Cecez-Kecmanovic & Janson, 1999) engaging in communicative action, individuals have a nontrivial possibility to reach a shared understanding of the situation and thereby achieve their own goals by coordinating their plans to action. We share the opinion of this author when she states that “a particular benefit of the theory of communicative action perspective is its sensitivity to the taken-for-granted assumptions, hidden agendas, interests and power relations which are all distorting communication”.

Our idea is to make messages an integrated concept in the project context aiming at improving the process quality (i.e. the development of a project in our terminology). This link aims at providing a richer view of decisions history with conversations associated to them. We say conversation (as a set of related messages), because a single message, or a single speech act, does not necessarily contain the richness of group communication. Anyhow, more than being interested in discussing the content of messages, we want to investigate the role and consequences asynchronous messages have in decision implementations.

The concept of message is part of many items in our proposal. For example, at the phase where meeting agendas are prepared, at the revision processes of meeting minutes, but mainly, at the post-meeting phase when decisions are being implemented and tasks are being executed.

Some steps taken during project development can be seen as “speech-acts” when they influence and change the status of decisions, issues, tasks, agendas and minutes, according to the LAP approach. Especially in distributed projects, the scenario we consider the most appropriate to justify our proposal, conversations takes place in various moments, and it normally involves groups of people.

4.4.3.1 Project Email Inbox

A project email inbox is a repository of all shared project messages, sent by project members. Messages that are of restrictive access should not be sent to the project email inbox.

Project members can send messages to other members using their email clients, which must enable messages without context (ordinary emails) and messages with context, i.e. messages referring to project elements (e.g. decisions and tasks) that represent our interest.

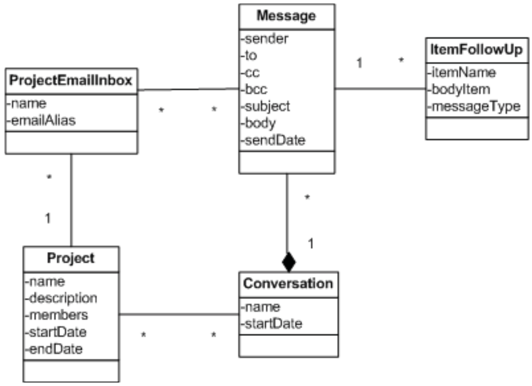


Figure 4-10: Class Diagram of Project Inbox, Messages and Conversations in the Project Context

Messages with context are semi-structured messages with fields containing the name of the project, and optionally, references to decisions and tasks. For each element selected (i.e. a decision, a task), a specific text is written. This semi-structure, allows for easier message parsing by automated mechanisms, without disturbing the usability of the simple task of sending an email message. For the user she is "just" sending a message with some more attributes, which do not require extra typing, but simple selections. The mechanism that supports the semi-structured message gets the contextual data from the project repository and shows it in an integrated way in the email client.

The Project Inbox represents the communication repository, and together with the other elements discussed below, allows for various levels of understanding the decision implementation process.

4.4.3.2 Conversation Types

Following the principles of LAP (Winograd & Flores, 1986), we identified two types of conversation to be integrated in our proposal:

- Conversation for **action,** used in the contexts of:
    - Requesting collaboration.
    - Requesting task executions.
    - Negotiating (i.e. in the task management, and in the meeting organization).
    - Follow up reports (see next sections), in order to solve a problem identified during decision implementation.
- Conversation for **clarification,** used in the contexts of:
    - Clarifying topics related to decision implementation, and task execution, providing opinions, arguments and contra-arguments.
    - Informing project participants about something of the group's interest, like sharing information.

Table 4-4: Relation between conversation and message types and location

Conversation Type	Message Type	Where it occurs
For action	Request task execution	Project repository
For action	Negotiate task	Project repository
For action	Reaction on follow up	Project repository
For action	Request collaboration	Project repository and Email client
For clarification	Request/Explanation for clarification	Project repository and Email client
For clarification	Sharing information	Project repository and Email client

The messages listed below occur implicitly in the workflow process, i.e. they are triggered by automatic mechanisms which are part of the project activities. Except for the notification mechanism (e.g. about the creation of an agenda), all other interactions are done without the support of a specific e-mail client, they are embedded in the project activities supported in our proposal.

1. Message of type "Request task execution" encompasses the first task assignment process that occurs after the meeting minutes are shared with project members, i.e. when tasks assignments are sent to performers' personal task lists.
2. Message of type "Negotiate task" is part of the task negotiation process and occurs between tasks requesters and performers until consensus is found.
3. Message of type "Reaction on follow up" aims at providing a support for the coordination perspective, i.e. when a project manager sees a problem in one of the reports for traceability, he can immediately react sending a message, from the context, about the problem observed, to the responsible parts.

The other three types of message take place in a combination of the project repository and an e-mail client.

1. Message of type "Request collaboration" encompasses the messages sent by the time of elaboration of an agenda proposal, or meeting minutes, where high levels of group collaboration takes place. Based on these messages, project members can actively react proposing changes on agendas and meetings outcomes.
2. Messages of type "Request/Explanation for clarification" are messages sent by normal email clients with copies to the project repository. These are **not** ordinary email messages, since they contain contextual information, like the name of the project, and optionally, the selected item(s) related to the request/explanation. In one single message, multiple requests or comments can be included. These messages follow a pre-defined semi-structure format. The integration of contextual information is meant to be a semi-automatic mechanism. The system offers the sender options for improving the contextualization automatically, for example, listing the name of the projects where the user is involved, or the list of on-going decisions, and the user must select those items to which she wants to make a request or provide an explanation. The use of semi-structured messages was also proposed in some systems in CSCW, e.g. (Malone et al., 1986b), (Simone & Divitini, 1999), (Michelis & Grasso, 1994) and was also mentioned by Winograd and Flores in the LAP context. Semi-structured messages contain drawbacks on the usability level, but offer an enourmous range of possibilities for automatic processing. Thus, a balance level of structure and flexibility should be

supported in order to motivate users to provide the minimum contextual information. That is also one of our goals.

3. Messages of type "Sharing information" are messages used for purely informing project participants about something (e.g. a conference, an article in the newspaper, a simple announcement). These messages are useful in implicitly related conversations. While the other types are explicitly related conversations.

4.4.4 Meeting Cycle Support

The meeting cycle support contains three phases, supported with different features. The mechanisms proposed aim at enabling documentation of meetings' preparation and realization, coordination of meeting outcomes and integration with the other modules previously discussed. Figure 4-11 presents the main classes involved in this context.

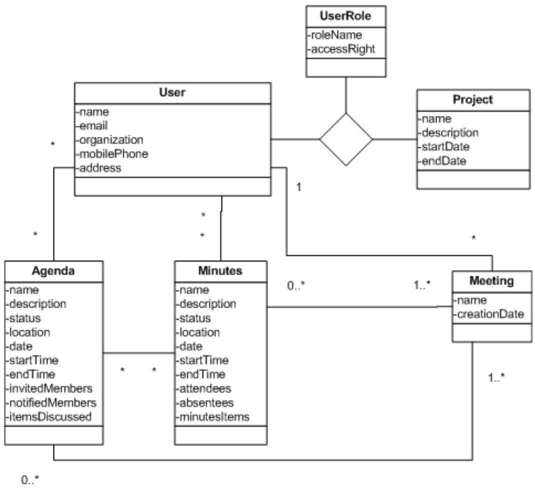


Figure 4-11: Class Diagram of Meetings in the Project Context

4.4.4.1 Pre-Meeting Support

In the meeting cycle this is the phase where new meetings are organized, agendas are proposed and the corresponding people are invited and notified.

The organization of a meeting should be trigerred by one person (e.g. the project coordinator), but it allows for the collaboration, i.e. revisions, additions, changes, from different project members, asynchronously.

Agendas take place in meeting contexts and have three sets of data:

- Metadata: a name, a description, location, date, starts and end time.
- List of people involved: here we have two groups, those who are invited to take part in the meeting, and those who are only notified.
- List of topics to be discussed: ideally, each topic should be associated (optionally) to a range of time dedicated to it, and a list of documents, which can represent presentations, explanations, arguments for the discussions.

Once an agenda is proposed, the group of people invited for the meeting is notified and can work on it. Appointments for the proposed meetings are part of the notification process and users only confirm it or propose a new date/time. For each item changed in the agenda, a new version is created in order to allow for historical documentation.

The most interesting part of our proposal, in what concerns "agendas", is the possibility for the users to create not only new topics to be discussed, by also to reuse what is already in the repository. Either from previous agendas, meeting minutes, but mainly, from the issues, decisions and tasks' repositories, thus enabling for continuous coordination.

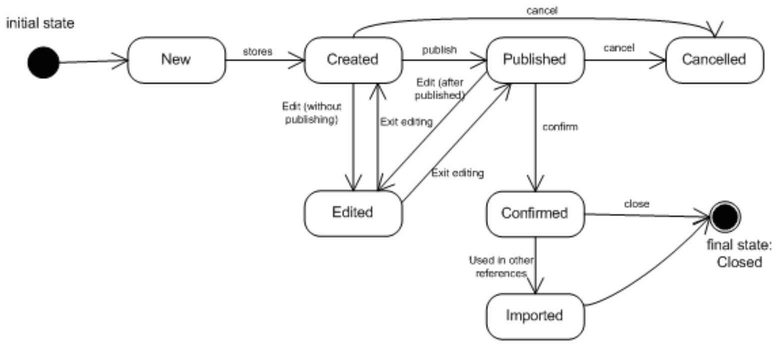


Figure 4-12: State Diagram for Agendas and Minutes

Agendas have their own life-cycle composed of the following steps (Figure 4-12):

- New – viewable by creater only, it can be deleted without notifiying other project members.
- Created – viewable by the created and by a group of selected coordinators
- Published – all project members can see it.
- Edited – withdrawn from public sight, except the creater and a selected group
- Confirmed – there is no possibility of editing anymore, and the issues are published.
- Cancelled – when there is a need to stop the planning of a meeting, and the agenda must be cancelled.
- Imported – status used to set that an agenda is used in another agenda or minutes, via the "import" operation.
- Closed – the agenda can no longer be imported.

4.4.4.2 Meeting Support

In the meeting cycle this is the phase that represents the time when meetings are being documented, i.e. when the meeting minutes are elaborated. The documentation process can take place during the meeting and after the meeting through the review process.

The organization of a meeting minutes should be under the responsibility of one person, but its finalization should be done by the collaboration of project members.

For collaboration here we mean revisions, additions, and changes. The collaboration is done asynchronously.

Meeting Minutes are managed in meeting contexts and have three sets of data:

- Metadata: a name, a description, location, date, starts and end time.
- List of people involved: here we have two groups, those who attend the meeting, and those who are absent.
- List of topics discussed: for each item, there is a possibility of defining the item type (if a comment only, or if an issue, etc), a discussion (optional) and a response (optionally used for required actions). For each decision made, a list of decision makers and the tasks associated to it are documented. By this time, the responsible for the minutes can provide more details to the tasks, like descriptions, start and end date and assign tasks to respective task performers.

Once a minute is proposed, the group of people who attended the meeting is notified and can work on it, making the necessary changes, adding details not perceived by the meeting scriber. For each item changed in the minutes, a new version is created in order to allow for historical documentation.

The most interesting parts of our proposal, in what concerns “minutes”, are the following:

- The possibility of explicitly represent what a decision is and associate them to tasks.
- The reuse of what is already in the repository, either from previous agendas, meeting minutes, but mainly, from issues, decisions and tasks, allowing cycles between meetings and continuous coordination support.
- The automatic generation of tasks in each personal member task inbox, which are assigned to tasks.
- The automatic inclusion of issues, decisions, and tasks in their respective repositories, keeping track that those items were created or re-used in that specific meeting minutes. Thus, allowing for later traceability options.

Minutes have their own life-cycle composed of the same steps as the agenda creation (Figure 4-12).

- New – viewable by the creator only, it can be deleted without notifiying other project members.
- Created – viewable by the creator and by a selected group
- Published – all project members can see it.
- Edited – withdrawn from public sight, except by the creator and a selected group
- Confirmed – there is no possibility of editing anymore, issues, decisions are published and tasks and assigned.
- Cancelled – when there is a need to stop the planning of a meeting, and the minutes must be cancelled.
- Imported – status used to set that the minutes is used in another agenda or minutes, via the “import” operation.
- Closed – the minutes can no longer be imported.

4.4.4.3 Post-Meeting Support

In the meeting cycle this is the phase that represents the time when meetings are over, documented and published, i.e. the consensus about what should be in the minutes is achieved and the tasks start to take place.

Two sets of mechanisms for this phase are the task management and the conversations integration, previously discussed in this chapter.

Apart from these, there is a very important aspect of the "post-meeting" phase that is the follow up of decisions and tasks. Here, we propose a group of reports in order to provide means for this traceability. We discuss the details of each report in the next section.

4.4.5 Reports

One of the innovatives concepts we are proposing in this thesis is the explicit management of decisions made, through the task and conversation integration to decisions made in meetings. However, all the concepts discussed in this chapter are "only" the basis to first document decisions, and second enable follow ups of decisions made. The main goal, though, is to reach a better understanding of the decisions under implementation. For this specific goal, we suggest the use of various reports which will support the users of our system to trace decisions and tasks.

This concept was influenced by some aspects from the area of "requirements traceability", from the Software Engineering. Requirements traceability is defined as the ability to describe and follow the life of a requirement, in both a forward and backward direction (i.e., from its origins, through its development and specification, to its subsequent deployment and use, and through periods of ongoing refinement and iteration in any of these phases) (Gotel, 1995). It can be achieved by using one or more of the following techniques: *cross referencing* (i.e. linking elements throughout the project documentation, like tagging, numbering, or indexing of requirements), *specialized templates and integration or transformation documents* (i.e used to store links between documents created in different phases of development) and *restructuring* (i.e. the documentation is restructured in terms of an underlying network or graph to keep track of requirements changes, like assumption-based truth maintenance networks, chaining mechanisms, and propagation).

Similarly, what we aim at achieving with our reports is a mean to trace "decisions", "tasks" and "conversations" associated to them, enabling cross referencing links, and structuring the meeting process in order to obtain a network of related elements.

4.4.5.1 Report on Decisions

- Objective: provide in a consolidated view, the understanding of the status of a decision (or a group of decisions) and easily identify where the problems are, through the explicit statement of tasks' statuses. It also allows for reactions on the problematic items, with the use of contextualized messaging.
- Report content: these reports can optionally (through the use of filters) show information about the overall status of decisions, their related tasks (with performers and status), conversations that took place in the context and responsible parts. The reports also allow for user reaction, i.e. once a

problem is identified in the decision implementation, the report viewer can send a message to the responsible part, using the context (e.g. the task name and status) as input and request that some actions must be done.

4.4.5.2 Report on Decision Life Cycle

- Objective: provide in one consolidated view, the understanding of the life-cycle of a decision (or many decisions) and easily identify how the changes have occurred.
- Report content: this report shows the complete life-cycle of a decision until its current status. Starting from the moment the decision was created, when tasks were assigned and changes that occurred.

4.4.5.3 Report on Meetings

- Objective: provide in one consolidated view, the understanding of the status of tasks related to a meeting.
- Report content: this report shows the list of tasks and respective status, conversations and documents related to each.

4.4.5.4 Report on Task

- Objective: provide in a consolidated view, the understanding of the status of a task (or a group of tasks) and easily identify where the problems are. It also allows for reactions on the problematic items, with the use of contextualized messaging.
- Report content: this report can optionally (through the use of filters) show information about the overall status of tasks, performers, status and conversations that took place in the task context.

4.4.5.5 Report on Task Life-Cycle

- Objective: provide in one consolidated view, the understanding of the life-cycle of a task (or many tasks) and easily identify how the changes have occurred.
- Report content: this report shows the complete life-cycle of a task until its current status. Starting from the moment the task was created and logging the changes that occurred.

Besides these reports, users can have different visualization filters for their private personal tasks: organization by creation and end-date, requester, and project.

4.5 Conclusion

In this chapter, we have discussed the main concepts involved in our proposal for improving the problem of decision representation, and follow up. Many of these concepts are part of other group support tools, like task management systems, or agenda organization. However, besides proposing some innovative concepts, like the explicit representation of decisions in meetings with a naturally used method of creating follow up possibilities via task and semi-structured messages, one of our main goals is to provide a holistic solution to the meeting cycle with decisions represented. Thus, these concepts can be implemented in different ways. Project management tools, meetingware or document management tools are possible candidated for that. In the next chapter we discuss the details of the implementation of our prototype.

5 Implementation of the COORDE Prototype

This chapter describes how the concepts and requirements discussed in chapter 4 were implemented as a set of coordination components in the COORDE (Coordination of Decisions) prototype, which aim at improving the issues discussed in chapters 1, 2 and 3. Part of the BSCW – *Basic Support for Cooperative Work* (Appelt & Koch, 1998) – concepts were implemented in the context of a master thesis concerning meeting follow up and task management (Koh, 2005), which was realized under the supervision of the author of this dissertation and of one of her supervisors.

5.1 COORDE's Components

The *COORDE prototype* is composed of an extension of a document management, the BSCW system and an extension of Microsoft Outlook (Microsoft, 2005). The BSCW extension package is named *Coordination Package* and is composed of a set of modules: issue, decision, task, and meeting management modules.

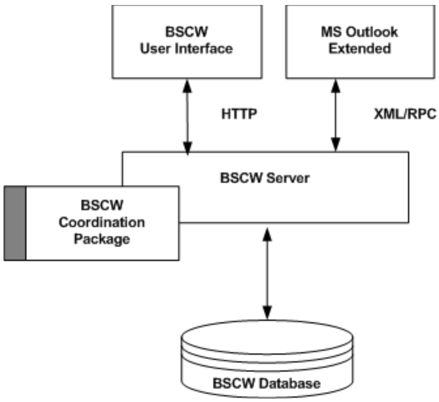


Figure 5-1: COORDE Architecture – showing the relationship between the different components in the system

5.1.1 Used Technology

The main technology used in our implementation was the BSCW system and Microsoft Outlook. Additionally, some libraries and extensions to these systems were integrated to the solution, to support features for the visualization of reports and better interactivity.

5.1.1.1 Standard BSCW

BSCW was developed at the GMD research institute, which was merged to the Fraunhofer Institut – FIT, in Germany. Nowadays BSCW is a product commercialized by a spin-off company Orbiteam (Orbiteam, 2005), in Bonn, and users of any part of the world can freely access the system via a public server on: http://bscw.fit.fraunhofer.de.

BSCW is a web-based groupware focused on the use of shared workspaces, which allows for flexibility and reachability of distributed group work. It contains a number of features that enable communication, cooperation and coordination, following the classical definition of (Ellis et al., 1991), (Olson & Olson, 1997). And although most of its features support cooperation and coordination, the standard BSCW (version 4.2) offers very little support to the concepts discussed in our research (e.g. meeting agendas, minutes, decisions). Therefore, we have chosen to use the BSCW as a basis for our solution, so that we could benefit from many of its standard group oriented funcionality and we have extended it to support the extra coordination features which are part of the scope of this thesis.

The main metaphor used in BSCW is of a “shared workspace”, also known as “folder”. A shared workspace acts as a repository for documents and other types of data that can be accessed by the members of that workspace. Shared workspaces can contain *sub-shared* workspaces, enabling flexibility, and at the same time, structure for managing documents.

BSCW support group membership associated with shared workspaces. These groups are built by the users according to what (e.g. project, ordinary files, personal repository) they want to manage and share. Each user of a BSCW server has a private space from where she can access her workspaces and private folders. If the user wants to share information with others she can ’invite’ them to the respective folder, thus turning it into a shared workspace (Appelt & Koch, 1998).

The standard version of BSCW provides a set of features which enable flexible cooperation support (Appelt & Koch, 1998):

- *Authentication:* Users of the BSCW system must authenticate using a simple name and password scheme.
- *Unrestricted document types:* BSCW makes no restrictions on what type of document is stored in a shared workspace (such as text documents, graphics or audio information) but even provides additional objects such as URL links and threaded discussions.
- *Meta-information on objects:* Users may attach additional information on objects in a description field. Furthermore, BSCW provides detailed information about stored objects such as size, MIME type and ownership. There is a set of meta attributes suggested by the system according to the element type (e.g. if URL, or if a text document), but users have the flexibility of extending the metadata information associated to them.
- *Version management:* Documents in a shared workspace can be put under version control, particularly useful for joint document production.
- *Access rights:* Flexible access rights on a per object basis can be used to specify in great detail who is allowed to do what in a shared workspace.
- *Document conversion:* Documents stored in a workspace can be converted to an appropriate format which is supported by the user’s computer (e.g. from MS Word to HTML).
- *Search facilities:* Shared workspaces can be searched for objects by different characteristics (e.g. type or name of author).
- *Event history:* BSCW provides limited awareness of other users’ activities by means of an object based event history. Once a document has been created, read or modified a corresponding event is generated, stored with the object and presented to the user the next time he logs into the system.
- *Other features*: folders that enable upload of emails, calendar and group calendar, synchronization with folders in the hard disk, communities’ support, among others.

BSCW is based on standard Web technology. The server consists of a set of CGI scripts that can be used to extend any unmodified Web server with the described shared workspace functionality (Figure 5-2). The system is written in Python (Python, 2005) an interpreted, object-oriented scripting language for which public domain interpreters are available for the commonly used server platforms (such as Unix or Windows). The only prerequisites for the use of the BSCW system are an Internet connection, a Web browser and an email address to register with the BSCW server (BSCW Administrator Manual, 2005).

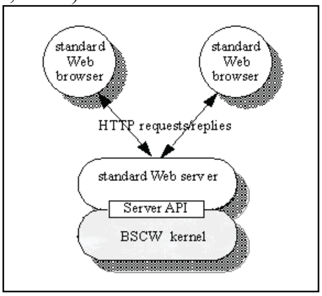


Figure 5-2: the BSCW Architecture (BSCW Administrator Manual, 2005)

The BSCW system also offers two alternatives for Application Programming Interface (API's), one in Python and one in Java (Java Sun, 2005). These APIs allow that BSCW and other systems communicate and exchange data, as is the case of our own implementation with Microsoft Outlook.

Three reasons justify the choice for BSCW. First the code of the BSCW is open for academic purposes, different from the majority of software available in the market. Second, the author of this thesis has worked for more than 3 years at the research institute (Fraunhofer FIT), where the BSCW was and is still developed. And third, the standard BSCW offers many features that are relevant for group support in our scenario—e.g. users and group management, and folders structure.

5.1.1.2 BSCW Extension: The Coordination Package

The coordination package was developed following the principles of "packages" in BSCW, which can be added or removed at run-time. The implementation of the package is organized just like the kernel. It consists of class definition modules (cl_*.py), utility modules (bs_*.py), operation handlers (op_*.py), interface template files (*.xhtml, *.html), icons (*.gif), style sheets (*.css) and configuration modules (bs_*config*.py). The package is stored in a separate directory: packages/Coordination.

Additionally, one of the reports was implemented with a Java Applet (Java Sun, 2005), on top of BSCW, to enable better interactivity and data visualization.

5.1.1.3 BSCW API and extensions

The API used in our implementation, specifically for the integration with Microsoft Outlook, was the BSCW Python API, a.k.a. X-API. This is an XML-RPC API to BSCW, which serves end-user calls to BSCW. It uses users' authentication and checks access rights for each call. The X-API provides almost all operations which a registered user may invoke in BSCW's normal Web user interface (BSCW X-API, 2005).

The X-API requires that an XML-RPC Client and a Server are used in the communication with the BSCW server, as shown in Figure 5-3 (BSCW X-API, 2005), and the data is transferred between the BSCW and another system, via XML format.

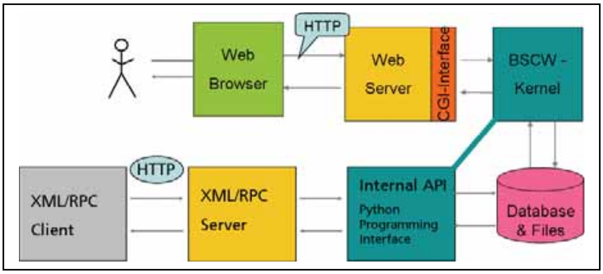


Figure 5-3: Accesses to BSCW via XML-RPC and the Web interface (BSCW X-API, 2005)

Since we have extended the BSCW system with our package, we have also created new functions for the X-API in order to support the use of the new objects (e.g. issues, decisios). For that, we have also used the same standards of the BSCW X-API. These new API's, which are exclusive of the Coordination package, are used in the communication process with Microsoft Outlook.

5.1.1.4 Microsoft Outlook Integration

Microsoft Outlook was used in our implementation for the integration of semi-structured messages with BSCW. The implementation was done using Visual Basic, according to the VBA (Visual Basic Application) technology. Besides Outlook, we needed an XML-RPC Client, which we have obtained in (Yoursurfice, 2005). This is an XML-RPC implementation for COM that complies with the specifications published on the XML-RPC site. It implements an XML-RPC client that can be called from any COM compliant application and it implements an XML-RPC server that can be used to call objects running on a Microsoft Internet Information Server machine. Another library was used to help improving the interactive features of our implementation (Redemption, 2005).

Two modules were implemented in Microsoft Outlook: a configuration profile to BSCW and a semi-structured email message, which uses information extracted from the BSCW system. More detailed information will be provided later in this chapter.

5.2 The COORDE Prototype

The Coordination package is composed of various modules that represent the concepts discussed in chapter 4 (e.g. project, meetings, agendas, minutes, issues, decisions). Figure 5-4 shows a high level schema containing the modules of our mechanisms and how they are associated to each other.

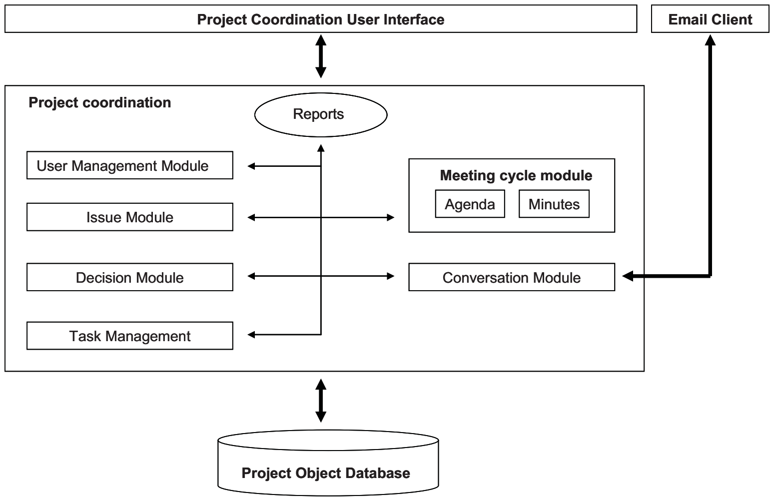


Figure 5-4: Modules of the Coordination Package

The **mapping between the conceptual model and the implementation modules** is almost transparent:

- Projects, in the conceptual model, are implemented as the highest element in the coordination hierarchy (i.e. the Coordination folder).
- Inside of projects there are four basic modules, the issues, decisions, tasks and meeting modules, where the elements described in chapter 4 are managed.
- The user management module follows the principles of the standard BSCW, but implements the different roles required in our conceptual model (e.g. the task requester, task performer, decision maker, etc.).
- The task management module implements the project and personal tasks, as well as the negotiation mechanisms.
- The conversation module is responsible for linking the semi-structured emails, sent from Outlook extended with our functions, to the other coordination elements (e.g. decisions, tasks).
- The report module communicates with all the others in order to provide the traceability mechanisms.
- In Outlook, the mechanims to send semi-structured messages were implemented, through two features. The first is a profile form for each BSCW server the user wants to send contextualized messages, and the second as an extension of the basic toolbar to send emails.

Inside of BSCW, the structure of the coordination elements was implemented following the general concept of folder. Thus, there is a project folder, with all other elements under its hierarchy. In order to distinguish our special coordination folders from standard BSCW folders, we have implemented awareness mechanisms, using different icons and visualization pages.

The next two sections show details about the implementation of COORDE, followed by a scenario of use of a ficticious project, so that the reader can understand how the system would assist project coordination in the daily work activities.

5.2.1 Details about the implementation

5.2.1.1 Coordination Folder

The Coordination folder is similar to a normal folder in BSCW; however, because this folder is the basis for the coordination of a project, a semi-fixed structure was implemented, in order to allow for objects' traceability. When a Coordination folder is created, automatically four sub-folders are created under its hierarchy: the issue folder, the decision folder, the project task folder and the meeting folder.

5.2.1.2 User Roles

There are three user roles in the context of Coordination folders, the project manager, i.e. the creator of the project structure, the project members, i.e. all BSCW users who can access information belonging to the project structure, and a new concept named "trustees". These are project members with management priviledges. Trustees can be seen as representatives or coordinators of organizations in distributed projects. They can create, change and delete data, such as agendas, minutes, decisions or tasks, inside of the coordination folder, even if they are not the creators of those data.

5.2.1.3 The Issue Folder

This is where all **issues** are stored and managed. Issues can be directly created inside of this repository or they can be resultants from agendas and meeting minutes.

Besides the basic issues attributes discussed in chapter 4, in the implementation we have added some others in order to allow for controlling multiple versions of the same **issue,** and for the management between issues, decisions and tasks. Each **issue**, then, **contains the following attributes**: name, description, state, version number, state number, active decision list, suspended decisions list, inactive decisions list, and version entries list.

The following actions are applicable to an issue:

- Create an issue – it is the action of creating a new item in the repository.
- Close issue – change the status of an issue to closed. This action can only take place before decisions, associated to an issue, are made. Otherwise, to close an issue, all decisions under its hierarchy must also be closed first.
- Edit issue – allows for editing issue's description and a new version is automatically created.
- Postpone issue – changes the issue's status to suspended and the cascade effect is applied to all decisions and tasks, which are related to it.
- Cancel issue – changes the issue's status to cancelled, and the cascade effect is applied to all decisions and tasks, which are related to it.

- Add a decision – allows the user to associate a decision to an issue.
- Complete issue – changes the issue's status to completed, but differently than "closed" issue, the cascade effect is not automatically applied to all decisions and tasks, which are related to it. It works the other way around, i.e. the decisions and tasks must be first completed, or closed, so that an issue can be set to completed.
- Reopen issue – allows an issue to be reactivated.
- Resume issue – changes the status of a suspended issue to active again.
- Archive issue – this action changes the status to "archived" and the issue disappears from the active list. However, an issue can only be archived after its status has been set to "completed" or "cancelled".
- Revive issue – changes the status of "archived" issues to active again. The cascade effect on dependent decisions and tasks does not take place automatically here.

In most of these situations, the user is invited to enter the reason or the explanation for the action taking place, so that the history of the issue can be recorded together with the version control.

5.2.1.4 The Decision Folder

This is where all **decision** are stored and managed. Decisions can be directly created inside of this repository, as a consequence of an issue, or as automatic results from agendas and meeting minutes.

Besides the basic decisions' attributes discussed in chapter 4, in the implementation we have added some others in order to allow for controlling multiple versions of the same **decision,** and for the management between issues, decisions and tasks. Each **decision**, then, **contains the following attributes**: name, description, state, version number, state number, parent issue list, active task list, suspended task list, inactive task list, version entries list and decision maker list. Some of them are created by the users (e.g. decision name, description, list of decision makers), and some are generated automatically by the system (e.g. version entry) based on the decision evolution and its relatioship with other issues and tasks.

The following actions are applicable to a decision:

- Create a decision – it is the action of creating a new item in the repository.
- Close decision – change the status of a decision to "closed. A decision can be closed when no tasks have been added to it. Otherwise, all tasks must be closed first.
- Edit decision – allows for editing decision's description and a new version is automatically created.
- Postpone decision – changes the decision's status to suspended and the cascade effect is applied to all tasks, which are related to it.
- Cancel decision – changes the decision's status to cancelled, and the cascade effect is applied to all tasks, which are related to it.
- Add a task – allows the user to add a task which is dependent to the decision. Tasks can be created already with the "confirmed" status, meaning that a task will be automatically created in the Personal task list of the task performer.
- Complete decision – changes the decision's status to completed, and the cascade effect is not automatically applied to all tasks, which are related to it. The tasks must be first completed, or closed, so that a decision can be set to completed.

- Reopen decision – allows a decision to be reactivated.
- Resume decision – changes the status of a suspended decision to active again.
- Archive decision – this action changes the decision status to “archived” and the decision disappears from the active list. However, a decision can only be archived after its status has been set to “completed” or “cancelled”.
- Revive decision – changes the status of “archived” decision to active again. The cascade effect on dependent tasks does not take place automatically here.

In most of these situations, the user is invited to enter the reason or the explanation for the action taking place, so that the history of the decision can be recorded together with the version control.

5.2.1.5 The Task Folder

The task folder, inside of the project context, is the repository of all project tasks. Different from issues and decisions, tasks in this context cannot be created independently. Tasks can be directly created inside of this repository, as a consequence of a decision, and as automatic results from agendas and meeting minutes.

Besides the basic task attributes discussed in chapter 4, in the implementation we have added some others in order to allow for controlling multiple versions of the same **task,** and for the management between issues, decisions and tasks. Each **task**, then, **contains the following attributes**: name, description, state, version number, state number, requester, performer, end-date, status, report, feedback, related decisions and version entries.

The task management is more complex than the issues and decisions’ folders management, since they involve two user roles: the *requester*, i.e. the person who requests a task to be done, and a *performer*, i.e. the person supposed to execute a task. The requests take place inside of a coordination folder structure, while the answer to them, i.e. the tasks execution and the negotiation, in case they occur, take place in a combination between the individual tasks’ folders of task performers and the project task folder.

Inside the **Project Task Folder**, *requesters* are allowed to:

- Confirm task – changes the task’s status to “confirmed”, which means the task is then automatically sent to the Personal Task Folder of the perfomer. The Personal folders are dynamically created in the user’s home directory, by the first time a user is assigned to a task.
- Edit task – allows for editing task’s description, end date and the task performer. A new version is automatically created for each editing actions.
- Cancel task – changes the task’s status to cancelled.
- Withdraw task – changes the task’s status to suspended, and the user is supposed to provide a reason for that, which is stored in the version entry list.
- Unsuspend task – changes the task’s status back to confirmed.
- Archive task – changes the task’s status to archived. Note that only completed or cancelled tasks can archived.

Inside the **Personal Task Folder**, *performers* are allowed to:

- Accept task – changes the task's status to accepted and the task requester is informed about the acceptance.
- Decline task – changes the task's status to declined and the task requester is informed about the declination. A reason must be provided for justifying this action.
- Complete task – changes the task's status to completed and a report may be provided.
- Update task status – this is just a simple way of providing follow up in a text format. No formal changes in the task's state takes place.
- Resume task – changes the task's status from declined to accepted.
- Repropose task – allows for changing task's parameters after a task has been accepted. This might be useful in case a task performer is unable to go further with the task executing.
- Retract reproposal – changes the proposed parameters to the original settings.

For both cases, we have used different coloured icons to ease the distinction between tasks.

5.2.1.5.1 *The Task Negotiation*

Our task negotiation mechanism is rather simple, because this is not the main goal of our research, but an add-on to the other functionality.

The task negotiation involves the two user roles: *requester* and *performers*. The negotiation takes place either changing explicitly the status of a task, informing via a small message the reason for the change or via the integration with Microsoft Outlook, where users can send messages associated to decisions and tasks in order to discuss, negotiate and clarify relevant aspects.

All these changes are stored with version control, allowing at any time that either the requester or the performer understands the evolution of the status and the reasons why they took place.

5.2.1.6 Meeting Module

The meeting module contains repositories for each meeting organized in a project. Additionally, it is directly related to the issue, decision and task folders, if these items become either an item in the agenda or in minutes. There is a central repository where all meetings are stored, and inside of it, users can create sub-folders according to the meetings that take place in the project (e.g. a meeting folder for a meeting named "*Developers Paris Meeting*", or "*Kick-off meeting*"). And inside of each meeting folder, users can create agendas and meeting minutes. The central meeting folder and the sub-folders are ordinary folders in BSCW, with the exception that only inside of them users are allowed to create agenda and minutes.

5.2.1.6.1 *Agenda*

Agendas contain a set of metadata and the items that are going to be discussed in the meeting. By the time an agenda is created, the agenda owner decides who will be invited for the meeting or who will be only notified about it. The agenda owner can reuse items from the repository, like decisions still in progress, or postponed items from previous meetings, or tasks that need to be discussed, and she can create new items. All these items together form an agenda for a meeting. After the creation process, the owner has the possibility of publishing the agenda, and consequently of notifying other users, so that the collaboration on the agenda's changes can take place.

5.2.1.6.2 Minutes

Meeting Minutes contain set of metadata and the items discussed during the meeting. By the time of minute's creation, which we suggest to be at the same time the meeting is taking place, the minutes' owner can decide to use an agenda as a basis for it, and automatically inherit the agenda items inside the minutes. Similar to the agenda functionality, the minutes allow for new items to be included, or for reusing information from the repository.

Besides including items in the minutes, and based on the dynamics of the meetings, the minutes owner (i.e. the person who is responsible to elaborate the minutes) can define if an item is a decision and, optionally, add tasks to it. For decisions, the users can also define if the decision was an individual decision or a group decision. And for tasks, the user can define the time the task should be accomplished and the task perfomer.

Different from the agendas, once minutes are published, the minutes' items are stored in the meeting folder, and also sent automatically to all respective repositories (i.e. issue, decision and task). For example, if a decision was made during the meeting, a new decision will appear in the decision folder of the project, with a reference to the meeting where it took place. The same principle is applied for tasks, but here tasks are automatically created in the project task folder and additionally in each personal task folder of the responsible party.

5.2.1.6.3 Notification

A notification occurs by the time of creation and change of some elements in our system.

In the context of meetings' organization, by the time agendas and meeting minutes are set with the status "published", the system sends notification messages to the related parties, i.e. project members explicitly selected to be notified. What we wanted to avoid here with this explicit selection was an unnecessary email overload.

The second context where notifications are used is at the task negotiation mechanism. Notification messages are used to inform, mainly the task requester, about any change with tasks under the project context, and to the performer in the cases of task confirmation and cancellation.

5.2.1.7 Semi-structured messages

The semi-structure functionality is composed of a feature that enables the configuration of BSCW servers and the feature of sending emails extended with our implementation. The functionality "Set BSCW Profile" serves as a profile for the user to connect to the BSCW servers they work with, retrieve data related to the coordination folders they are members of and define which of them the default project is.

For the default project coordination folder one email alias should be provided, in order to facilitate sending messages to the respective BSCW folder.

The functionality of “sending a semi-structure” message does not disturb the user experience in sending a message in a normal way, but additionally it provides the possibility of using contextual data in the message that are automatically sent to the specific folders in the BSCW server. After the user chooses to send a message, and the Outlook standard form is displayed, the user can press a button, which retrieves information from the BSCW server defined in the “Set BSCW Profile” functionality. Based on what is retrieved, i.e. a list of projects, with the default showed on top, a list of decisions, and a list of tasks, the user can choose these elements alleatorilly, write the message and press the send button, like in any other email message. However, here, three actions take place in the background. First, the message is sent to the enlisted people (i.e. in the fields TO, CC, and BCC), second, a copy of the complete message is sent to the email folder of the project and third, smaller messages, are sent to the specific decisions or tasks repositories.

Project members may use email as they are used to, but they also have the option to send contextualized message to the project repository, sharing their doubts, or clarifications, and at the same time feeding the project memory with the communication process. As a consequence, we get a better synergy between communication and coordination. Project members can, then, see these messages either in the general project repository, which actually is a standard BSCW feature, or inside of the specific item as a “note”, a standard BSCW object.

5.2.1.8 Reports

One of the main elements in our implementation are the reports. They are the best way to perceive the integration we are proposing in this thesis. Our objetive with the reports was to enable coordinators and project members to see, in quick views, how the development of the project is taking place and act accordingly.

Several reports were implemented:

- Reports on Life-Cycle: two versions were implemented, one for decisions and one for tasks. Both are text-based reports that show the evolution of one or more of these elements and the changes they had since their creation time.
- Report on Meeting: this report shows the evolution of elements regarding a specific meeting.
- Report on Decision: two versions of this report were implemented, a text-based and a graphic version. In both cases, coordinators or project members can check whether decisions made are evolving according to the plans. Additionally, in these reports users can see related information to a determined element.

5.3 Scenario of Use

We have included this section in order to illustrate the dynamic aspects of the COORDE prototype, and how it could be used in a project life-cycle. For that, we use a ficticious scenario which was created using common characteristics of European Union (EU) projects, i.e. of distributed groups working in teams, who have face-to-face meetings once in while. Thus, together with the conceptual model described in Chapter 4, the reader can have a more holistic understanding of the system and its usage.

We first show the complete scenario and then we break the scenario in various parts. For each part, we include a table showing its relevance to the project life-cycle, the conceptual elements involved in this part of the scenario, and the respective added value.

5.3.1.1 The MDSS Project

Nick is leading an EU project that requires collaboration of various partners, who work in different organizations, cities and countries. The official communication language is English. Because partners are located in different places, and since traveling is not always an option for exchanging information, there is a need for a platform to support the management and sharing of information. For this specific project, Nick has chosen to use BSCW with the extended Coordination package (COORDE). The standard BSCW offers many functions that support cooperative work, and together with COORDE, it enables the coordination of meetings, decisions and tasks in an integrated way.

The first activity Nick does on BSCW is to create a coordination folder for managing the MDSS (Mobile Decision Support System) Project and, at the same time, he defines project members (e.g. Andreas Maltez and Elisa Merici) and coordinators of each organization (e.g. Davis Mind and Caroline Winter). Coordinators have access and control over all elements in the project folder. Whereas ordinary project members have read access to all information, but can only change what is assigned to or created by themselves.

When Nick creates the coordination folder, the system automatically creates four sub-folders: issues (for issue management), decisions (for decision management), tasks (for task management) and meetings (for meetings, agenda and minutes management). Additionally, Nick creates normal folders for managing workpackages, according to the EU project style of dividing activities in Workpackages. For example, he creates a normal BSCW folder named "Workpackage 1: Requirements" and a folder to receive emails directly on the server.

After this first structure is created, project members can input new issues, decisions and project tasks in the respective folders, which can be used inside the context of meetings or discussed and solved independently. These elements can be referred to at the preparation of a face-to-face meeting, as items in the agenda, and later, as decisions made and tasks that need to be implemented, in the meeting minutes.

Nick starts the usage of these coordination elements proposing an issue named "papers to submit", followed by a decision of "two journals (one top and a second one)", which is an important strategic issue that needs to be discussed in the coming meetings. Doing so, Nick creates awareness of an issue that needs to be managed whithin the project. Likewise, project members and coordinators can include other issues they consider relevant to share and discuss in future opportunities. And different from exchanging information via email, where some times people are not copied, the COORDE issue-decision-task management folders support the idea of centralizing these elements in the project folder, so that every project member can consult and check whether to contribute or critisize, at any time, from any place.

After the first month of activities, there is a need to organize a meeting about the user requirements deliverable. This deliverable is the first activity a group of project members needs to do together. Nick starts a draft for the meeting agenda, which includes a location, date, time and some items to be discussed. He uses the opportunity to include the issue on "papers to submit", and the decision on "two journals (one top and a second one)" in the agenda. Besides, he adds the specific items about the deliverable, for example: comparative study on the market, related scientific work, benchmarking with EU projects, choice of methodology and methods, and writing the final version.

When he finishes the first draft of his proposed agenda, the system automatically notifies project coordinators, so that they can revise what Nick has written, adding new items, rephrasing items, etc. This process is different from revising agendas via email exchanges, which is a common practice nowadays. With the support of COORDE, agendas are elaborated in a collaborative, but centralized way, and are integrated to other project elements.

As time goes by and the meeting date arrives, the agenda is already revised and agreed upon. During the meeting time, Nick takes the role of writing down the meeting notes using COORDE. He first imports the items from the agenda, and changes what is necessary (for example, the people who are present, and the absentees, or the date and time). Besides indicating new decisions or tasks, Nick can also consult the system and import items that were not predicted in the agenda. Still during the meeting, issues concerning the deliverable are discussed, decisions are made, tasks are defined and project members are assigned to execute them. For example, a decision that a comparative study on the market is made, and three tasks are associated to its implementation: Christian Schmidt is assigned to “Evaluate Microsoft products”, Andreas Maltez is assigned to do a “Research on IBM products” and Olivia McIntere is assigned to “Evaluate Siemens products”. All this information is noted by Nick, using the system.

At the end of the meeting, Nick has two options, one is to share immediately his notes with the other project members via the system, or to work on it, first alone, and then share it with the others. Nick chooses the first option and publishes the minutes immediately after the meeting is finished.

As a consequence to publishing the minutes two events take place. Firstly, all users with coordinator roles can revise what is written and suggest changes. Secondly, all project members who were assigned to tasks receive a copy of the specific task in her/his personal task list. Andreas Maltez is one of them.

When a task gets to a personal task list, the assigned person can initiate a negotiation process, either reproposing the task, and offering a justification, or simply negleting to doing the task, again specifying a reason for that. Based on this negotiation a process of going back and forth can take place between task performers and task requesters, until they get to an agreement (e.g. between Nick and Andreas). The task requester has also the possibility of reassigning the task, if necessary. And all these steps are logged in the life cycle of a task, thus allowing, at any time, that one learns how the task has evolved. Also, as part of the task management module in COORDE, the task performers can change the task status based on their assignment, for instance, completing a task. And as a cascade effect, once all tasks associated to a decision are completed, a decision implementation is also completed. And similarly, if there are problems with tasks, which are necessarily associated to decisions in our model, a decision implementation is affected.

In the personal task folder, users can manage project tasks together with their private tasks. For example, Andreas keeps track of all his tasks, but not only from the project that Nick manages. He keeps also his private tasks, and he is the only person to have access to them. Private tasks are kept in each individual task folder, while project tasks are linked to the project they belong to, allowing for two levels of access, one from the task performer perspective, and the other from the project task management folder.

Still after a meeting has been held and tasks are being executed, project members can exchange emails and explicitly quote elements from the coordination folder. In order to do that they use an extended version of Microsoft Outlook, which retrieves decisions and tasks from COORDE automatically, and allows users to write semi-structured messages referring to them. These messages are used to clarify aspects of the decision implementation, and at the same time they work as a group communication memory. Andreas, for instance, has a doubt about a task he is supposed to execute and he chooses to send an email with reference to the task and decision he has doubts about. His message goes to any person he may list in the to, cc and bcc fields in his email client, and copies are sent to the specific task and decision folders in COORDE.

In the meanwhile, while negotiations are taking place and tasks are being executed, project members can check how the activities are evolving, through the use of reports available in the system. Caroline Winter checks the task life cycle report of tasks that were assigned to the members of her team and see that Olivia McIntere declined to execute the task that was assigned to her, which requires she interferes to understand why and see if another person can execute it. While Nick is interested in having an overall picture of the project schedule. Thus, he consults the reports on decisions and see whether tasks are late, and if necessary, he can react to them. These reports, which show results of the integrated solution, represent information that cannot easily be obtained without a system like COORDE, since all these pieces would be managed separately in each organization, by each individual.

This cycle of inputing data in the system (i.e. issues, decisions, tasks), managing meetings (i.e. with agendas and minutes), and following up how decisions and tasks are implemented (i.e. with the task management module, with semi-structured contextualized messages and with reports), can be repeated several times along a project life cycle. The COORDE allows the linkage between these elements and offers a consolidated facility (i.e. the various reports) in order to allow the visualization of the elements together. As a positive consequence, since all elements are managed in a centralized and integrated way, there is a possibility of awareness about problems that occur during the decision implementation.

5.3.1.2 Part I: the creation of a project coordination folder

Nick is leading an EU project that requires collaboration of various partners, who work in different organizations, cities and countries. The official communication language is English. Because partners are located in different places, and since traveling is not always an option for exchanging information, there is a need for a platform to support the management and sharing of information. For this specific project, Nick has chosen to use BSCW with the extended Coordination package (COORDE). The standard BSCW offers many functions that support cooperative work, and together with COORDE, it enables the coordination of meetings, decisions and tasks in an integrated way.

The first activity Nick does on BSCW is to create a coordination folder for managing the MDSS (Mobile Decision Support System) Project and, at the same time, he defines project members (e.g. Andreas Maltez and Elisa Merici) and coordinators of each organization (e.g. Davis Mind and Caroline Winter)—**Figure 5.5**. Coordinators have access and control over all elements in the project folder. Whereas ordinary project members have read access to all information, but can only change what is assigned to or created by themselves.

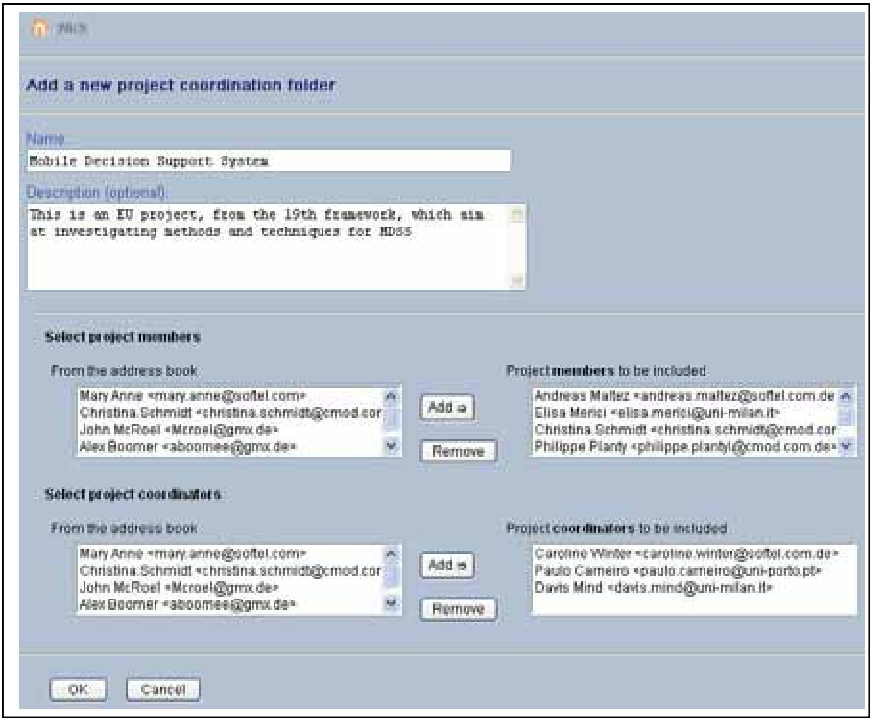


Figure 5-5: Nick creates the Coordination Folder for the MDSS Project

Relevance to the Project Life-Cycle	
Meeting cycle	At this moment there is **no reference** to the meeting cycle. This part of scenario only defines the place (i.e. folder) where all information about the project will be stored.
Decision cycle	At this moment there is **no reference** to the decision cycle, since no activities have started yet.
Conceptual model elements	- Project - User Roles
Added value	The contribution of the COORDE here is on the coordination facilities with the new user roles implemented: ordinary project members are similar to users who share workspaces in BSCW, but project coordinators have other priviledges in the project life-cycle, for e.g. they can revise agendas, minutes, change tasks' information. The concept of the coordination folder is a core element in our solution, because it enables the integrated view of all project related information.

5.3.1.3 Part II: input of issues, decisions as candidates for agendas

When Nick creates the coordination folder, the system automatically creates four sub-folders: issues (for issue management), decisions (for decision management), tasks (for tasks management) and meetings (for meetings, agenda and minutes management). Additionally, Nick creates normal folders for managing workpackages, according to the EU project style of dividing activities in Workpackages. For example, he creates a normal BSCW folder named "Workpackage 1: Requirements" and a folder to receive emails directly on the server—**Figure 5.6.**

At any time, users can input new issues, decisions and project tasks in the respective folders, which can be used inside the context of meetings or discussed and solved independently. These elements can be referred to at the preparation of a face-to-face meeting as items in the agenda, and later, as decisions made and tasks that need to be implemented. Thus, allowing for links between meetings, and more important, for tracking possibilities.

Nick starts the usage of these coordination elements proposing an issue named "papers to submit", followed by a decision of "two journals (one top and a second one)"—**Figure 5.7**, which is an important strategic issue that needs to be discussed in the coming meetings. Doing so, Nick creates awareness of an issue that needs to be managed whithin the project. Likewise, project members and coordinators can include other issues they consider relevant to share and discuss in future opportunities. And different from exchanging information via email, where some times people are not copied, the COORDE issue-decision-task management folders support the idea of centralizing these elements in the project folder, so that every project member can consult and check whether to contribute or criticize, at any time, from any place.

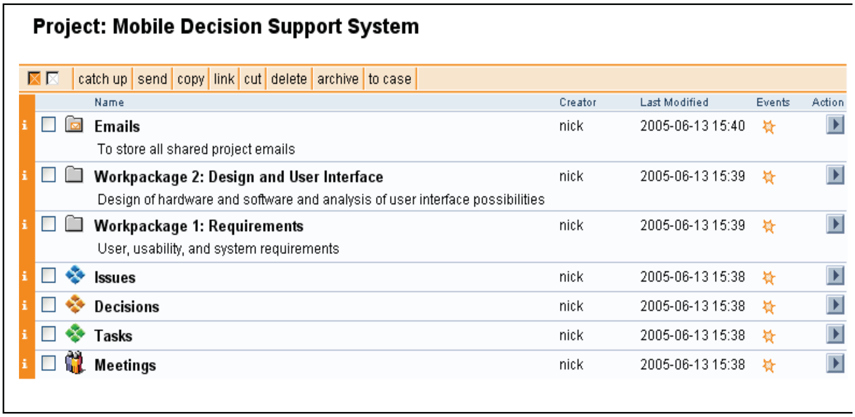


Figure 5-6: The Coordination Folder structure

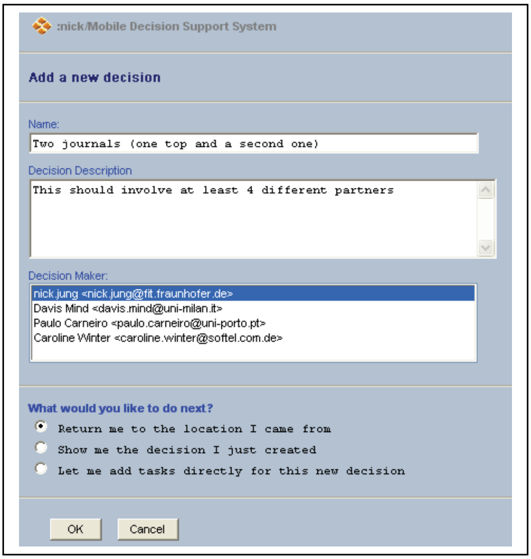


Figure 5-7: Adding a decision to the repository outside of a meeting context

Relevance to the Project Life-Cycle	
Meeting cycle	This part of the scenario is related to the **Pre-meeting phase**, when project members are using the system to input new issues and decisions that need to be discussed in meetings later.
Decision cycle	Project members can include new decisions made, which are outside of the scope of meetings and later these **decisions can become candidates to agenda items**.
Conceptual model elements	- Issues - Decisions - Project Tasks - Meetings
Added value	These four pre-defined folders guarantee that these elements will be traceable after meeting minutes are created and refer to issues, decisions and tasks. Project members have two options: they can refer to issues, decisions and tasks inside of meetings, or they can use these coordination mechanisms outside of meetings.

5.3.1.4 Part III: elaboration of an agenda

After the first month of activities, there is a need to organize a meeting about the user requirements deliverable. This deliverable is the first activity a group of project members needs to do together. Nick starts a draft for the meeting agenda—**Figure 5.8**, which includes a location, date, time and some items to be discussed. He uses the opportunity to include the issue on “papers to submit”, and the decision on “two journals (one top and a second one)” in the agenda. Besides, he adds the specific items about the deliverable, for example: comparative study on the market, related scientific work, benchmarking with EU projects, choice of methodology and methods, and writing the final version.

When he finishes the first draft of his proposed agenda, the system automatically notifies project coordinators, so that they can revise what Nick has written, adding new items, rephrasing items, etc. This process is different from revising agendas via email exchanges, which is a common practice nowadays. With the support of COORDE, agendas are elaborated in a collaborative, but centralized way, and are integrated to other project elements.

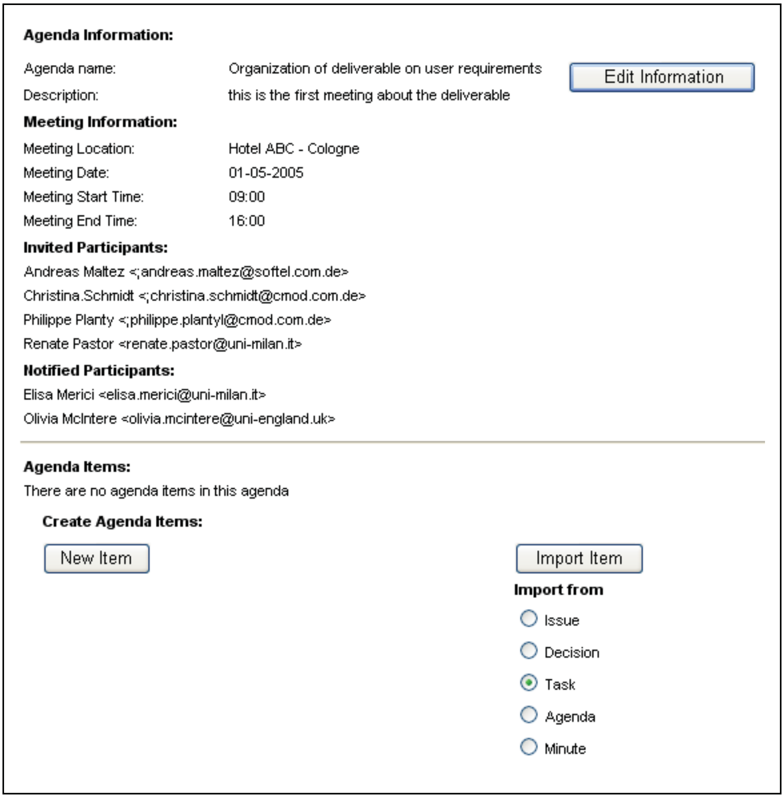


Figure 5-8: Starting the agenda elaboration for a meeting

Relevance to the Project Life-Cycle	
Meeting cycle	This part of the scenario is related to the **Pre-meeting phase**, when project members are using the system to create agendas for meetings in the future. The agenda organizer can refer to issues, decisions and tasks, which are already stored in COORDE (import item button) or can include new items. After finishing to create **agendas**, a collaborative activity of revising it, remotely, takes place.
Decision cycle	In this part of the scenario, **decisions stored in the decision folder can become items in the agenda,** and will be discussed in the meeting.
Conceptual model elements	- Meetings - Agenda - Issues - Decisions - Project Tasks
Added value	Users can create meeting information and propose agendas using either new items, which are part of the current context (e.g. here the deliverable) or **reuse** items which are already in the system, for example any pending or recurrent problem. Additionally, the collaborative process of working on an agenda in a centralized, but shared way, is a plus in COORDE.

5.3.1.5 Part IV: elaboration of meeting minutes

As time goes by and the meeting date arrives, the agenda is already revised and agreed upon. During the meeting time, Nick takes the role of writing down the meeting notes using COORDE. He first imports the items from the agenda, and changes what is necessary (for example, the people who are present, and the absentees, or the date and time)—**Figure 5.9**. Besides indicating new decisions or tasks, Nick can also consult the system and import items that were not predicted in the agenda—**Figure 5-10**. Still during the meeting, issues concerning the deliverable are discussed, decisions are made, tasks are defined and project members are assigned to execute them. For example, a decision that a comparative study on the market is made, and three tasks are associated to its implementation: Christian Schmidt is assigned to “Evaluate Microsoft products”, Andreas Maltez is assigned to do a “Research on IBM products” and Olivia McIntere is assigned to “Evaluate Siemens products”. All this information is noted by Nick, using the system—**Figure 5-11**.

At the end of the meeting, Nick has two options, one is to share immediately his notes with the other project members via the system, or to work on it, first alone, and then share it with the others. Nick chooses the first option and publishes the minutes immediately after the meeting is finished.

As a consequence to publishing the minutes two events take place. Firstly, all users with coordinator roles can revise what is written and suggest changes. Secondly, all project members who were assigned to tasks receive a copy of the specific task in her/his personal task list. Andreas Maltez is one of them.

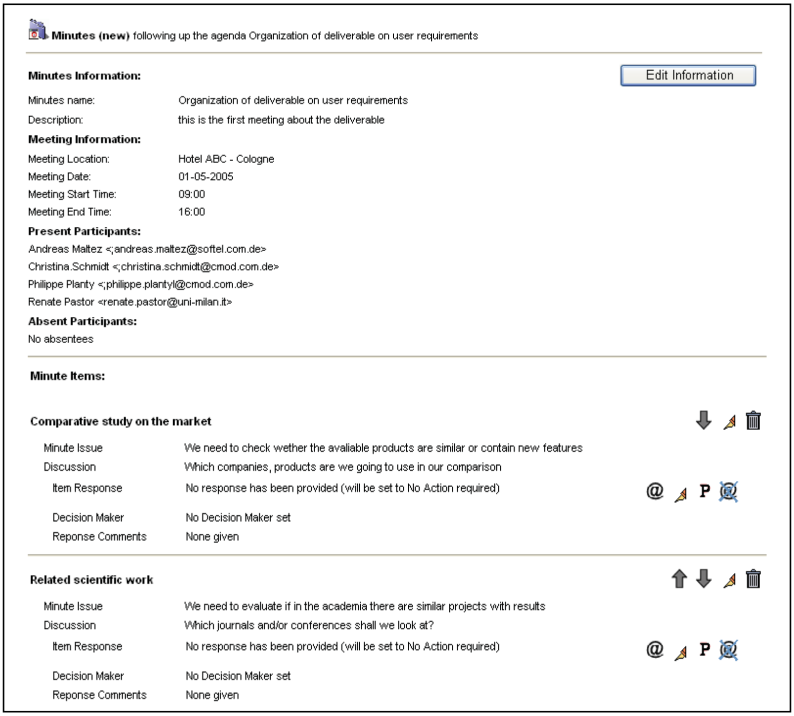


Figure 5-9: Minutes Metadata (upper part) and items discussed (lower part).

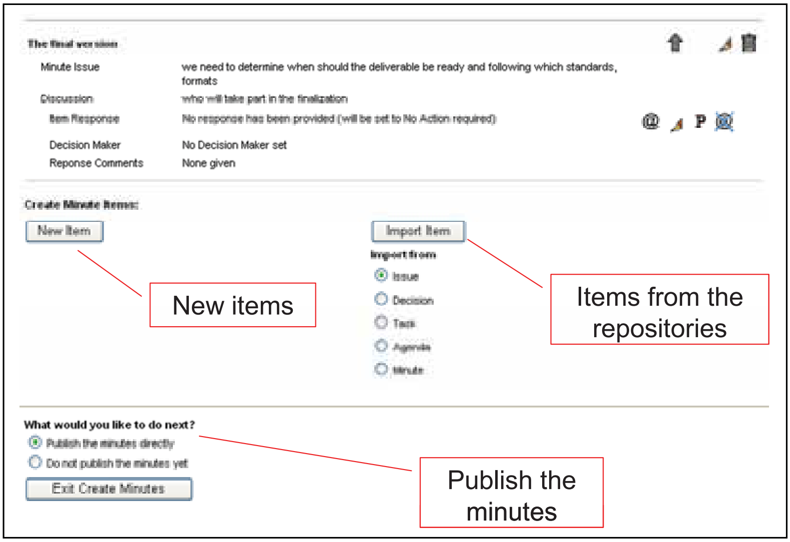


Figure 5-10: Minutes inclusion of items

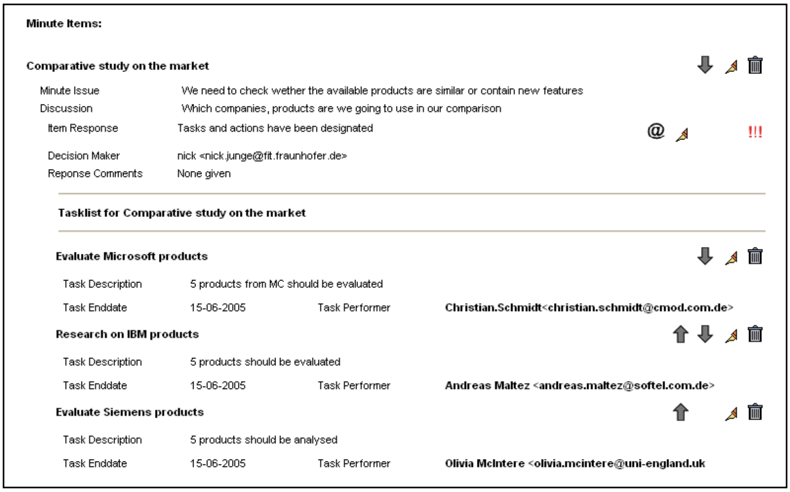


Figure 5-11: Minutes item that became a decision with three tasks associated to it

Relevance to the Project Life-Cycle	
Meeting cycle	This part of the scenario is related to the **Meeting phase**, when project members are using the system to create **meeting minutes**. After finishing to create minutes, a collaborative activity of revising it, remotely, takes place.
Decision cycle	In this part of the scenario, **decisions are made and tasks are associated to them.**
Conceptual model elements	- Meetings - Agenda - Minutes - Issues - Decisions - Project Tasks
Added value	Users can create meeting minutes, **reusing the proposed agenda,** and reusing items from the database (e.g. issues, decisions) that were not predicted in the agenda. **Decisions are explicitly represented** and from then own, they will be traceable. Additionally, the collaborative process of working on meeting minutes in a centralized, but shared way, is a plus in COORDE.

5.3.1.6 Part V: follow up of decisions with tasks

When a task gets to a personal task list, the assigned person can initiate a negotiation process, either reproposing the task, and offering a justification, or simply negleting to doing the task, again specifying a reason for that. Based on this negotiation a process of going back and forth can take place between task performers and task requesters, until they get to an agreement (e.g. between Nick and Andreas). The task requester has also the possibility of reassigning the task, if necessary. And all these steps are logged in the life cycle of a task, thus allowing, at any time, that one learns how the task has evolved. Also, as part of the task management module in COORDE, the task performers can change the task status based on their assignment, for instance, completing a task. And as a cascade effect, once all tasks associated to a decision are completed, a decision implementation is also completed. And similarly, if there are problems with tasks, which are necessarily associated to decisions in our model, a decision implementation is affected.

In the personal task folder, users can manage project tasks together with their private tasks. For example, Andreas keeps track of all his tasks, but not only from the project that Nick manages—**Figure 5-12**. He keeps also his private tasks, and he is the only person to have access to them. Private tasks are kept in each individual task folder, while project tasks are linked to the project they belong to, allowing for two levels of access, one from the task performer perspective, and the other from the project task management folder.

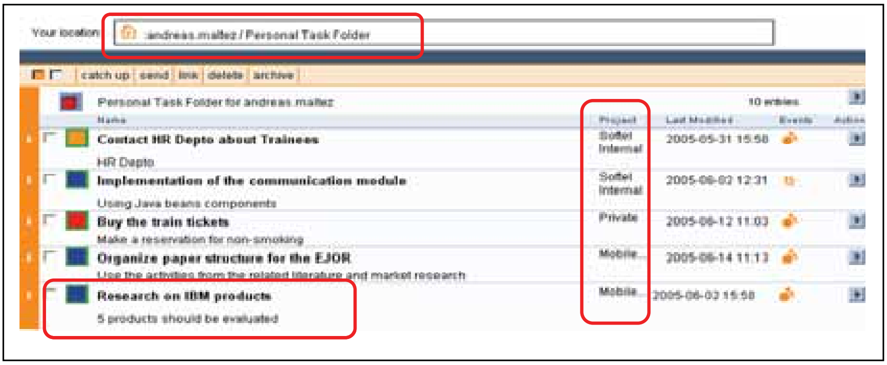


Figure 5-12: Example of a personal task folder of a ficticious user

Relevance to the Project Life-Cycle	
Meeting cycle	This part of the scenario is related to the **Post-meeting phase**, when project members are using the system to execute tasks and, consequently, to implement decisions.
Decision cycle	In this part of the scenario, **decisions are implemented** through the tasks that are associated to them.
Conceptual model elements	- Decisions - Project Tasks - Private Tasks - Conversation for action
Added value	Users receive, automatically, in their respective private task folders, the tasks they are assigned to execute. A negotiation between task requesters and performers can take place inside the system, and a log of the task evolution (i.e.

	versioning) is kept. Any reference to a task, either a change on status, or a change in the assignment, or a delay is automatically linked to the decision implemented, which the task is part of. Thus, tasks give follow ups on decisions.

5.3.1.7 Part VI: follow up of decisions with semi-structured messages

Still after a meeting has been held and tasks are being executed, project members can exchange emails and explicitly quote elements from the coordination folder. In order to do that they use an extended version of Microsoft Outlook, which retrieves decisions and tasks from COORDE automatically, and allows users to write semi-structured messages referring to them—**Figure 5-13**. These messages are used to clarify aspects of the decision implementation, and at the same time they work as a group communication memory. Andreas, for instance, has a doubt about a task he is supposed to execute and he chooses to send an email with reference to the task and decision he has doubts about. His message goes to any person he may list in the to, cc and bcc fields in his email client, and copies are sent to the specific task and decision folders in COORDE—**Figures 5-14 and 5-15**.

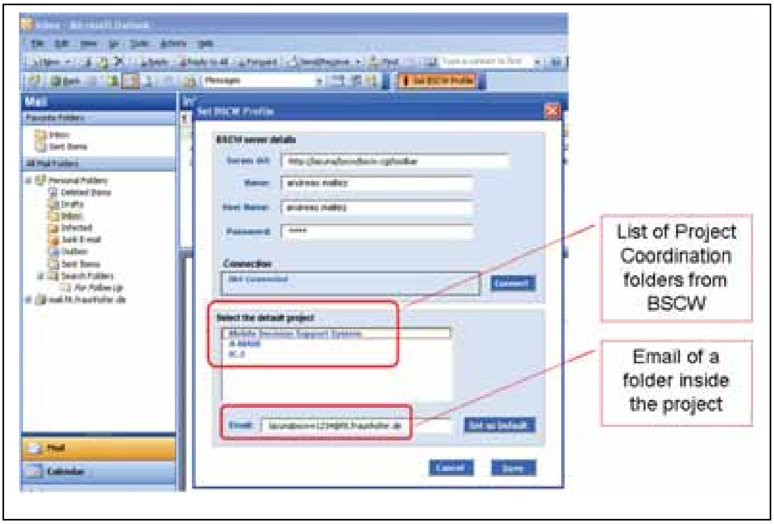


Figure 5-13: Setting the communication between BSCW and Microsoft Outlook

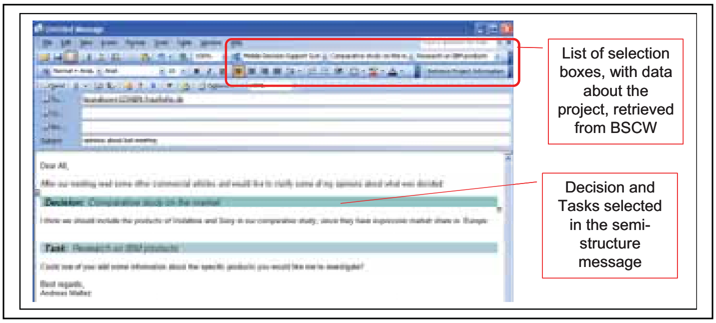


Figure 5-14: Semi-structured message showing a decision and a task selection

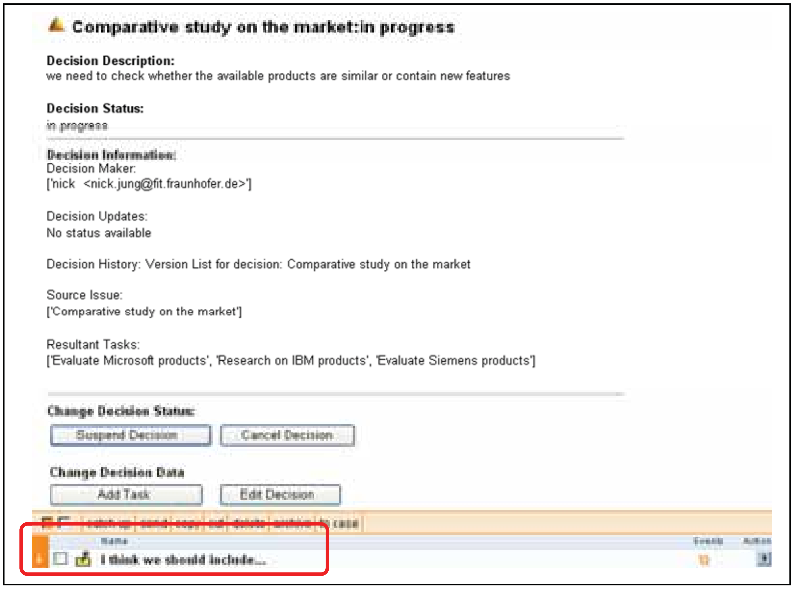


Figure 5-15: A decision and a note, written via the semi-structure message

Relevance to the Project Life-Cycle	
Meeting cycle	This part of the scenario is related to the **Post-meeting phase**, when project members are using the system to **exchange semi-structured messages**, which are related to decisions and tasks.
Decision cycle	In this part of the scenario, **decisions are implemented** through the tasks that are associated to them, and they can have **conversations associated to them**.
Conceptual model elements	- Decisions - Project Tasks - Conversation for action - Conversation for clarification
Added value	Users can send contextualized messages to any project member, and keep a link to the decisions and tasks they refer to, creating a possibility of integrating messages and actions. Additionally, this practice can be seen as a first step for creating a group memory of the communication exchanged in a project, with explicit links to decisions and tasks.

5.3.1.8 Part VII: consolidated views, the reports

In the meanwhile, while negotiations are taking place and tasks are being executed, project members can check how the activities are evolving, through the use of reports available in the system. Caroline Winter checks the task life cycle report of tasks that were assigned to the members of her team and see that Olivia McIntere declined to execute the task that was assigned to her—**Figure 5-16**, which requires she interferes to understand why and see if another person can execute it. While Nick is interested in having an overall picture of the project schedule. Thus, he consults the reports on decisions and see whether tasks are late, and if necessary, he can react to them—**Figures 5-17 and 5-18**. These reports, which show results of the integrated solution, represent information that cannot easily be obtained without a system like COORDE, since all these pieces would be managed separately in each organization, by each individual.

This cycle of inputing data in the system (i.e. issues, decisions, tasks), managing meetings (i.e. with agendas and minutes), and following up how decisions and tasks are implemented (i.e. with the task management module, with semi-structured contextualized messages and with reports), can be repeated several times along a project life cycle. The COORDE allows the linkage between these elements and offers a consolidated facility (i.e. the various reports) in order to allow the visualization of the elements together. As a positive consequence, since all elements are managed in a centralized and integrated way, there is a possibility of awareness about problems that occur during the decision implementation.

Figure 5-16: Task life-cycle report

Decision Name: (Decision From Minutes Organization of deliverable on user requirements) Comparative study on the market

Decision Creation Date: 15/06/2005 Decision Creater: nick <nick.junge@fit.fraunhofer.de>

Decision is **in progress**

Decision Description: This is a decision resulted from Organization of deliverable on user requirements
item Market research

Decision Maker: Nick Junge

Tasks available under the decision:

Task Name: Evaluate Microsoft products

Task Creation Date: 15/06/2005 Task End Date: 15-07-2005

Task Performer: Christian Schmidt <christian.schmidt@cmod.com.de>

Task Description: 5 products from MS should be evaluated

Task is confirmed

VersionList for task: Task was not edited

No Documents related to the task

Task Name: Evaluate Siemens product

Task Creation Date: 15/06/2005 Task End Date: 15-07-2005

Task Performer: Olivia McIntere <olivia.mcintere@uni-england.uk>

Task Description: 5 products should be analyzed

Task is declined

VersionList for task: Task was not edited

No Documents related to the task

Task Name: Research on IBM products

Task Creation Date: 15/06/2005 Task End Date: 15-06-2005

Task Performer: Andreas Maltez <andreas.maltez@softel.com.de>

Task Description: 5 products should be evaluated

Task is accepted

VersionList for task: Task was not edited

Messages related to the decision

Discussion Name: I think we should include...

Start Date: 21-06-2005 Started by: andreas maltez <andreas.maltez@softel.com.de>

Subject: I think we should include...

Posted on: 21-06-2005

Message:
I think we should include the products of Vodafone and Sony in our comparative study, since they have

Figure 5-17: Decision Report (text-version) with tasks and notes associated to it

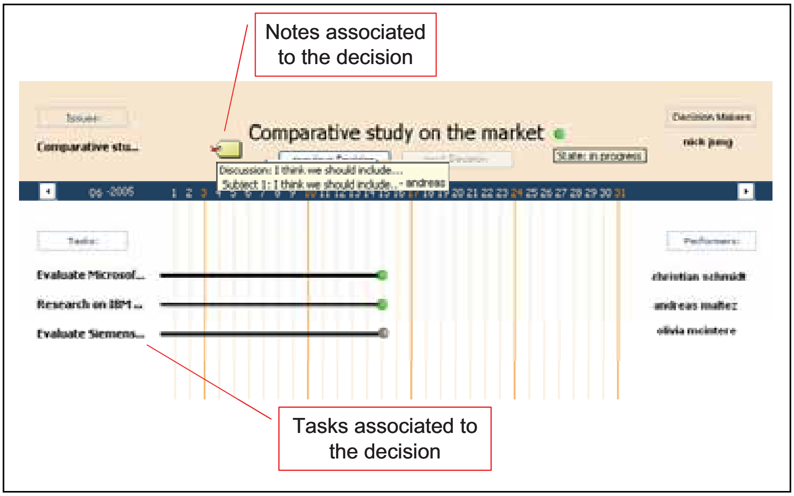


Figure 5-18: Decision Report (graphic-version) with tasks and notes associated to it

Relevance to the Project Life-Cycle	
Meeting cycle	This part of the scenario can be useful in **all parts of the meeting cycle**, since reports can illustrate indicators for preparing agendas, be useful for clarifying aspects during a meeting, and for following up in the post-meeting phase.
Decision cycle	In this part of the scenario, **decisions are being followed-up with the assistance of the various reports**
Conceptual model elements	All elements
Added value	The consolidated views are powerful tools for information management and visualization. The reports available in COORDE shows different perspectives on meetings, tasks, and decisions. They also help coordinators recognize where there are problems during the decision implementation.

5.4 Conclusion

In this chapter we have shown our implementation and how it deals with the integration of the concepts discussed in chapter 4. However, this is just a starting point. The current implementation could be extended in several ways. First, the BSCW system allows for scalability and inclusion of other functionality, either as part of the Coordination Folder or together with other packages available. Features regarding the support for other meeting activities like brainstorming, or mechanisms to enable the documentation of the decision making process itself could be integrated to what we have implemented so far. From the Microsoft Outlook perspective, other integration levels could be developed, like a synchronization mechanism between tasks, for example.

Lastly, this implementation did not mean to be the solution for all problems regarding coordination of projects, but our idea was to create a set of mechanisms to support the meeting cycle and decision implementation activities. Moreover, these mechanisms should not be (according to the requirements discussed in chapter 4) an extra tool. That is one of the reasons to implement them as part of tools (BSCW and Outlook), which are used for the means of project management and email communication. It is important to note that these mechanisms could be implemented as part of other tools, such as Microsoft Project, or any other document management system. And the same can be said about the email client.

6 Evaluation

6.1 Introduction

This chapter describes how the evaluation of this research was realized, which methods were used and the justification for our choices. The evaluation considered the research questions proposed in chapter 1 and the concepts proposed in chapter 4, which were implemented and presented in chapter 5. At the end of this chapter, we discuss the results obtained.

The evaluation was done in three parts: a validation, a partial evaluation, and a complete evaluation. These activities were different, and had complementary goals. The first activity realized was a formative evaluation, aiming at understanding "what" and "how" we should design some of our ideas. This activity was a mix of validation to check whether the research ideas were sound, and at the same time exploratory, since we have inquired experts about new requirements in the field of decision implementation. The second and the third activities were summative evaluations, in the sense that they were used to evaluate "how" the prototype was adequate to the working practices. The second activity took place in the context of a master thesis (Koh, 2005) under the supervision of the author of this thesis, and has focused on the meeting cycle and on task management; the third, and final, activity considered the integrated approach, and some improvements based on the results from the second evaluation.

6.2 Evaluation objectives

Our evaluation considered the two research questions proposed in chapter 1, with what Elliot Stern (quoted in Sommerland et al., 1992) defines as evaluation:

> *Evaluation is any activity that throughout the planning and delivery of innovative programmes enables those involved to learn and to make judgements about the starting assumptions, implementation processes and outcomes of the innovation concerned.*

The two research questions, their respective sub-questions and their relationship to the requirements defined in chapter 4, are the following:

Question, sub-question		Requirement
Research question 1:	can a set of automated group support mechanisms that enables the link between pre-meeting, meeting and post-meeting activities, representing decisions explicitly, provide a basis for decision tracking, allowing for better coordination during project development?	R1, R2, R3, R4, R5, R6, R7, R8, R9, R10, R11, R13, R15, R17, R18, R21, R22, R23, R25, R26, R30
Sub-question 1.1:	is the link between pre-meeting, via agenda representation, meeting, via minutes representation, and post-meeting via task and decision representation, a means to improve coordination during project development?	
Sub-question 1.2:	is the integration of email, task and project/document management system a means to better understand, in a unique view, how decisions (made in meetings) are being implemented?	
Research	can common electronic elements, which belong to daily project	R10, R18,

question 2:	scenarios, i.e. tasks and emails, provide means for creating awareness of problems that occur during decision implementation?	R23, R25, R26

The research question 1 encompasses all requirements used for our prototype, since it aims at evaluating the holistic approach and the elements which it is composed of. While research question 2 is mapped to the requirements which are more specific to deal with two elements of our approach: tasks and emails in the context of decision implementation and follow-up.

6.3 Qualitative research

According to Myers (Myers, 1997), quantitative research methods were originally developed in natural sciences to study natural phenomena. Examples of quantitative methods are survey methods, laboratory experiments and formal methods. Whereas, qualitative research methods were developed in the social sciences to enable researchers to study social and cultural phenomena. Examples of qualitative methods are case study research and ethnography. Qualitative data sources include observation and participant observation (fieldwork), interviews and questionnaires, and the researcher's impressions and reactions.

One of the motivations for doing qualitative research, instead of quantitative research, comes from the ideas that humans are different than the natural world, mainly because we talk. Thus, qualitative research methods are designed to help researchers understand people and the social and cultural contexts within which they live (Myers, 1997).

There are many ways to distinguish qualitative and quantitative research: methods, objetives, data gathering, and data analysis, among others. But one of the most relevant criteria that influenced our decision for realizing a qualitative research, instead of a quantitative research, is how context could be treated according to (Harvey & Myers, 1995). In more traditional quantitative techniques, context is treated as either a set of interfering variables that need controlling, known as noise in the data, and other controlled variables which are experimentally set up in order to seek for cause and effect relationships. In the more traditional qualitative approaches, context is treated as the socially constructed reality of a named group, or groups, of social agents and the key task of observation and analysis is to unpack the webs of meaning transformed in the social process whereby reality is constructed. In quantitative techniques, cause and effect are the main objects being searched for, while, in qualitative techniques, meaning in context is the most important framework being sought. Because context is crucial to qualitative observations and analyses, techniques which explore contextual webs of meaning are important (Harvey & Myers, 1995).

Since our research combines aspects from IS and CSCW, we have checked which methods these areas used to realize evaluations. In IS there has been a general shift in research away from technological to managerial and organizational issues, hence an increasing interest in the application of qualitative research methods (Myers, 1997). While, in the area of CSCW the main method used in evaluation activities has been ethnography. This involves intensive and detailed study of a real workplace over a prolonged period. Data collected include audio, video and annotations about the observations executed. The drawback from this method is the effort and time required to establish an adequate ethnographical study. We do not deny its value, but we share the opinion that in many situations they are not possible to be realized. Methods such as interviews, questionnaires and group discussions are also very common in the area of CSCW (Ramage, 1999). The same author discusses other methods, which are also common in the CSCW area, like heuristic evaluations, user testing, and participatory design. Quantitative methods are also common in CSCW (Olson & Olson, 1991), but with the objective of collecting quantitative data about a single specific factor, attempting to screen out other influences (Ramage, 1999).

In our evaluation, we have conducted qualitative research activities, aiming at having a broader understanding of the adequacy and completeness if the prototype supports the requirements in the context of distributed project management, and how our proposal was divergent from the users' perspective.

6.4 Selected methods

In our three activities we used a combination of different research methods, one from the area of social research (surveys, specifically in the format of intensive interviews), scenarios (from the area of software engineering and usability) combined with intensive interviews, and expert walthroughs (from the usability area).

Surveys are the most common form of research in the social sciences (Williamson et. al, 1982): questionnaires, interviews are examples of them. Surveys research are systematic attempts to collect information to describe and explain the beliefs, attitudes, values, and behaviour of selected groups of people. The broad adaptability of this research method is one of its strengths (Williamson et. al, 1982). Survey method is not only flexible and adaptable to a number of research purposes; it is also capable of producing, from a relatively small sample, results that can be generalized to a much larger population of interest. The main weakness of this method is its total dependency on the respondents—on their memory, their interest, their clarity of self-perception, their frankness, and their honesty. Thus many factors determine how people report about their attitudes. Because the value of a survey depends on the representativeness of the group surveyed, the sampling plan and its execution are almost as crucial to final success as are the planning and execution of the overall survey itself.

Intensive interviews (Williamson et. al, 1982) differ from other interviews in that they are less structured. They allow the interviewer flexibility in questioning the respondent—flexibility that enables the interviewer to ask for or give immediate clarification in cases of misunderstandings. Intensive interviewing emphasizes the incipient relationship between interviewer and interviewee. The quality of this relationship is what determines how productive the interview will be. Intensive interviews employ no questionnaires, although they may use topic guides, which may include a few structured questions to be asked of every respondent (semi-structured interview).

Intensive interviews are sometimes employed in exploratory studies, in which the researcher wants to obtain a "feel" for what to ask and how to ask it in a large survey research study. In other studies in may be the only method of data collection. One of its limitations is that data generated do not allow for quantitative analysis and do not permit statistical inferences to be made about the population from which the respondents were drawn (Williamson et. al, 1982).

We have used intensive interviews in the initial validation phase, and in the second evaluation.

Usability evaluation aims at realizing analysis or empirical study of the usability of a prototype or system. Its main goal is to provide feedback in software development in support of an iterative development process, and at the same time, to recognize problems, understand underlying causes, and plan changes.

There are many methods used in usability evaluation: cognitive walkthroughs, task-analysis, heuristic evaluations (UsabilityNet, 2005). We have chosen to execute what in the literature is known as "expert walkthrough" in the later phase of our evaluation process, since, at that moment, we wanted mostly to evaluate if the implementation was sound according to the concepts involved in the scenario of coordination of distributed projects. At the same time, we wanted to raise questions regarding what was missing in our prototype.

Expert walkthroughs are organized based on usability guidelines, often working from a checklist. In general it is suggested to use more than one expert and provide them with guidelines that can be at many levels e.g., screen layout, detailed analysis of cognitive states.

The last method for evaluation used was scenarios (Carroll, 1995), (Maguire & Bevan, 2002). A scenario is a typical interaction between the user and the system or between two software components, described in the form of a story that is used in order to provide a clear view of the system being designed and a more concrete understanding to stakeholders and/or users. Scenarios are normally used in inquirying phases of projects, however, we have used scenario to help illustrate a set of relevant tasks of our implementation, so that users and experts could easier understand the applicability of the system. The end goal of using scenarios in our evaluation was to build user experiences that could met the expectation of the user group and result in the successful accomplishment of users' goals.

6.5 Activities realized

Before describing our activities, we would like to emphasize a key aspect of our evaluation: the target public used. The target user group of our system is mainly composed of project coordinators, or project managers, although other people could be directly or indirectly influenced by the system as well. For example, project members like system developers, sociologists, users in general, stakeholders, and so on. The reason why we want to stress this information at this point is to explain our choices during the evaluation phases regarding whom we have interviewed and whom we have invited to our expert walkthroughs. Since we could not afford to execute evaluations with all possible users, we focused our evaluation on the opinions, and critics of the experts, i.e. project members with the responsibility of coordinating and managing, who could better understand if the balance between our proposal and the daily practice would be adequate or far, and if the benefits were enough to justify more structure and coordination as it is proposed in our solution.

6.5.1 First evaluation: concepts validation and requirement inquiry

The first evaluation took place in November 2003 in the form of intensive interviews. Six experts, i.e. project managers of European and German Projects, were interviewed with the objectives of validating some of the concepts of this research, and to discuss a first version of a form to document meetings and represent decisions explicitly. The questions used as a basis for these interviews can be found in Annex II, and the form example in Annex III.

Five men and one woman were interviewed, five participants from Germany, and one from Italy. All of them, by that time, had managed or were managing distributed projects with multi-disciplinary teams. All were using some type of technology to support their management activities.

Each interview was individually realized, where five were face-to-face interviews, and one by telephone. The duration of each of them was of approximately 1 hour and a half.

6.5.1.1 Results of the first evaluation

The results of the first evaluation were very inspiring and at the same time indicated some limitations, and some requirements for the proposal.

❖ **Use of computer supported tools**

All interviewees were using computer supported tools to elaborate meeting organization and to share information with partners. Two interviewees commented the use of paper-based to-do lists since it was easier and quicker than typing them in electronic format, but the drawback was that they could not share them with their colleagues. All of them have extensive experience with the BSCW system.

❖ **Meeting agendas**

In all projects they were taking part of at that time, project members used to share agendas for meetings via emails, and, in some cases, upload the documents to the BSCW system.

❖ **Meeting, minutes and decisions**

Regarding the meeting organization and minutes, all of them said the management was very similar to the agenda, i.e. a first version was circulated via email, and some times uploaded to the BSCW system, and a final version was done by one or more individual’s collaboration. Meeting minutes were elaborated either at the time of the meetings, or afterwards. In all cases, they said one individual was responsible to take the main notes and the others could send their comments via email after the meeting. No editing collaboration was taking place at the meeting time, and no one used an extra tool for editing documents, but Microsoft Word.

The meeting minutes were said to contain some meta information, confirming our proposal, which contained mostly the location, the date and hours, and the list of attendants. Additionally, the items discussed in the meeting were documented in different ways, which was suggested to be very dependent on who was taking the notes. Some people said to be more direct, and write short sentences, while others write long paragraphs documenting the whole discussion that took place.

The format used to represent decisions varied among the interviewees. Half of them said to represent decisions explicit only through assertive sentences, but the other half said they represent decisions also in other ways, and not only sentences. For example, through sketches, diagrams, pictures of the white boards or flip-charts. In the text format, some used highlights (colors, effects) to emphasize decisions. And mostly, in many meetings, they do not represent them explicitly at all. All of them represent to-do lists in the minutes, or a list of task, already associated to the person or organization that is responsible for the respective execution.

Most of the interviewees said that these organizations of agendas, minutes and representation of decisions depend on the characteristics of the project. Some of their experiences were of very structured and organized projects, while others were freer of structure. The latter type was indicated as the most difficult one to obtain such information.

❖ **The form template for the minutes**

Concerning the minutes form presented to the interviewees, they all understood it and suggested to be a good initiative, but too structured. Besides, most of them considered that the form would require a big cognitive effort to categorize so much information during the meeting. Maybe as a process that could take place after the meeting, this could work, but if more flexibility was proposed.

❖ **Track of decisions**

All participants said it is difficult to track decisions using their current methods and supporting tools. When they need to do it, they normally use different means: via meetings, phone calls, emails and looking up at old meeting minutes. Ideally, all of them would like to have an easier way of doing it. Two of the interviewees shared their Microsoft Project squemes in order to facilitate, at least, the understanding of what needs to be done in an overall view.

Another difficulty pointed by one of the interviewees is that in many cases the sentence written in the minutes, named “decision” is not enough to explain what the real decision is. For him, decisions sometimes are “hidden” and not explained.

Some of them also mentioned the need to have a decision, in the minutes, named “pending”, which could be further discussed in a future meeting.

❖ **The wished and not wished support**

When asked about their needs and requirements for a tool to support these coordination activities of distributed projects, organization of meetings, documentation of meetings, decisions and tasks, the interviewees suggested many interesting ideas.

- The tool could not be too structured
- The tool could not be too invasive
- They wanted to be reminded or informed of things still open, things done, and things that are doing well

- The tool should work during the meetings, and after the meetings
- The tool should work connected to the Internet or not
- The tool should have an extension for mobile situations
- The tool should make use of graphical elements in order to facilitate the usability and an easier understanding than extensive texts
- The tool should provide "control points"
- The tool should represent responsible parts explicitly
- The tool should provide flexibility on receiving or visualizing information about decision follow-up (e.g. a web-page with an overall view)
- The tool should have more than one template for meeting minutes
- The tool should be result oriented
- The tool should provide different views about the information managed
- The tool should have an explicit representation of a decision
- Decisions should be highlighted in meeting minutes
- The tool should not send follow ups via email, since there is already an overload of emails
- Users should be able to classify critical decision, critical impacts (like a flag)
- The tool should enable several versions of the same meeting minutes (e.g. draft, 1st, 2nd, final)
- The tool should enable a representation of a life-cycle of a decision (e.g. pending, being implemented, finalized, obsolete)
- The tool should be integrated to Microsoft Project
- The tool should allow to connect people to slides
- The tool should represent decisions in Gantt-charts and Pert-charts

Regarding audio and video, all of them recognized that these other types of media could offer richer capabilities to the system, but all were skeptical if they would use them because of the amount of effort and setup required. Besides, they all doubted if the technology would be mature enough to provide automatic results.

6.5.2 Second evaluation: a partial evaluation

The second evaluation took place in April 2005, in the context of a master thesis (Koh, 2005). Two methods were used: scenario and intensive interviews. Seven people took part in this phase.

Five men and two women were interviewed. And all of them are Germans. All interviewees had some experience in distributed projects (European and German), and some had managed project themselves.

The focus of this evaluation was on the meeting cycle support and on its relationship with the task management mechanism. The main objectives were: (i) to verify if the meeting cycle support was achieved and (ii) to verify the usability implications of the implementation.

Each interview was individually realized. The duration of each of them was of approximately 1 hour and a half.

6.5.2.1 Results of the second evaluation

Since this was part of the master thesis, the complete description about the activities and details about the questions and scenario can be found in (Koh, 2005). Here we present the main results of the activities, regarding the two objectives proposed in the evaluation:

❖ **To verify if the meeting cycle support was achieved**

The sofware was well evaluated by the interviewees, and confirmed that the idea of support the meeting cycle was achieved. Additionally, the participants indicated that the coordination proposed could help the management of activities, but also the documentation of the process.

Some of their complaints were concerned about the structure of the forms used, although they suggested that only with a few improvements they could imagine using the tool in their own projects.

Another important comment was concerned the scalability of the tool. Although they all understood the idea of task management linked to meeting minutes, the management of tasks was indicated as critical if the number of tasks would increase too much.

The idea of having agenda and minutes shared in a central place was very well evaluated. They all saw advantages in working in this way.

As missing features, the participants commented the need to attach a document to an item in the agenda and in the minutes.

❖ **To verify the usability implications of the implementation**

All participants suggested that the tool would require some training, since it deals with the complexity of managing projects, but that was not seen as a negative feature. The learnability curve would compensate for the coordination and documentation benefits. The concepts were clear, but would require some cognitive adaptation.

Some adaptation of working practices would also be necessary. This was seen not so positively, but as suggested by most of the participants, with few usability changes, they would get more convinced about the required changes.

On the other hand, they also commented that the new way of working could facilitate and lead the working such that agendas and minutes would be naturally more organized and easy to understand. Some comments were also done regarding the clarity of icons and the use of colors. Others required more flexibility on the personal task lists, for instance a mechanism to keep the tasks done separated from the active tasks. Another relevant critic was regarding the easiness of the agenda and minutes. Some times users complained of having too much text as metadata, making it difficult to distinguish the most important items from those of less importance. More highlights were suggested as a possible improvement.

Some missing features were indicated on the ease of use aspect. Graphics were missing, in order to facilitate the visualization of a group of information. And an easy way of linking a task to a future meeting. This would be very helpful for the time of organizing a new meeting, when the coordinator would check the task status.

6.5.3 Third evaluation: an integrated evaluation

The third evaluation took place in June 2005 in the form of expert walkthroughs. Six experts, i.e. project managers of European Projects, were invited to take part of the two sections, three in each. The sections were organized in three parts: (i) an introductory part to provide context about the research and evaluation objectives, (ii) the presentation of a scenario of use, used as a guideline during the walkthrough, and (iii) the system demo showing how the support of each part of the scenario would take place. This last part was sub-divided in six other parts, and between each part a set of questions concerning a cluster of concepts was discussed. Annex IV shows the scenario used in these activities, and Annex V presents the list of questions used as a basis for the expert walkthrough. The scenario in Annex IV is a simplified version of the scenario used to illustrate the system in chapter 5.

Four men and two women took part of the activities, five participatns from Germany, and one from Slovak Republic. Four of them had managed, or were managing distributed projects and two had more than five years of experience in the same type of project, but not as project managers. All had experience of projects, which used some type of technology to support management activities. The users were asked to mainly concentrate on the role of project coordinators, or managers, but they could also make comments regarding any other role in a project development.

Each section was realized with three individuals and a moderator (the author of this thesis), and took approximately 2 hours and a half each.

The focus of this evaluation was the integrated approach, but also specific concepts used in the implementation, as well as the usability of the system. We used as a reference the ISO 9241-11, Part 11, Guidance on Usability, which indicates *Efficiency*, *Efficacy* and *Satisfaction* as the main usability goals to consider while evaluating a system.

6.5.3.1 Results of the third evaluation

The third evaluation focused on two usability aspects: effectiveness (representing efficacy) and satisfaction. The aspect of efficiency was commented, but since the experts did not try out the system themselves, this aspect was not explicitly evaluated.

Another interesting aspect of this activity was that among the six participants four are also experts in usability. In this sense, the combination of project experience and usability was very enriching.

In the following items, we present the results of the evaluation of each of the six cluster of concepts discussed in the expert walkthroughs.

- Section 1 dealt with the coordination folder, issue, decision and task’s folders, project task and personal task folders,
- Section 2 dealt with the meeting folder, agenda, and minutes,
- Section 3 dealt with the task negotiation mechanism,
- Section 4 dealt with semi-structured messages’ integration,

- Section 5 dealt with the reports, and
- Section 6 dealt with the integrated approach.

❖ **Section 1:** Coordination Folder, Issue, Decision and Task's Folders, Project Task and Personal Task Folders.

The effectiveness of the "Coordination Folder" was rated between good and normal. The suitability for the task was evaluated as good, in most cases, but experts pointed out some details missing, like the representation for sub-structures as part of the concepts, for example in the European projects, where the work is mostly divided in "workpackages" (i.e. clusters of tasks, activities and deliverables under the responsibility of an individual or an organization, with sub-coordination responsibilities also present). The completeness was considered good, except for a date to show the end of the project and a distinction between active projects and finalized projects.

The satisfaction of the "Coordination Folder" was highly rated between good and very good. Experts supported the idea of having the distinct roles of manager, coordinators (trustees) and project members with different access rights, and they also commented the advantage of having these roles defined at the beginning of the project and not dependent on the access rights feature of BSCW, which allows for similar definitions, but which would require lots of management. Additionally, some of them shared the opinion that other roles should be available and easily managed as these ones already implemented. They also observed that sub-levels for representing the organization of workpackages could involve other sub-levels of management.

Regarding the "Issue, Decisions and Tasks' Folders" the experts rated the effectiveness as good, since these concepts showed the possibility of managing elements in a clearer way when compared to the free use of documents, which are difficult to parse and broadly used in projects.

All experts, except one, understood the idea of a "cascade" between "issues-decisions-tasks" as a natural link, and useful. The only one who did not immediatelly understood it observed she thinks in a different way. First, she thinks in tasks or actions, which can raise issues, which can they be solved through certain decisions. That comment showed that although the concepts are clear, maybe their naming and relationship could be more flexible, allowing that each individual could make use of them in a different way.

Experts missed more awareness about the different status of issues, decisions and tasks and their responsible parts. For them, the folder concept was fine, but inside of it there should be more options for reorganization according to the users' need (e.g. status type, responsible, due date) and icons that could facilitate the distinction between their status, which represent where their concentration should be put on. Treating them equally at the first level of visualization requires the users to enter each of them to check which ones need reactions.

The experts observed that these folders could easily be the basis for extended functionality on statistics, which might be interesting in certain phases of a project, for example in quality control.

A missing functionality suggested by one expert was the reuse of information from other projects. Not necessarily in the context of meetings only, but in general. Issues, decisions and tasks used as basis for previous projects could be imported according to his opinion.

Concerning "Project Task and Personal Task Folders" the effectiveness and satisfaction were rated good.

The idea of having the relationship between project task lists and personal tasks lists was considered very useful, and well done, since they at the same time allow for the general view of a project and an invidual view. One of the experts commented she would prefer to receive a message together with a task assignment instead of simply receiving the task in her personal task list, since this would be more personal. But all the others commented that this was not an invasive action of the system, since the users could also deny executing a task.

The colors used to distinguish the tasks status were considered too many. The experts suggested that a smaller range would facilitate the learning process and it would still be representative enough.

An important critic was done regarding tasks' assignments. At least in German and European distributed projects, frequently, tasks are assigned to organizations and not to one individual. Normally, they are assigned to the representatives of an organization, who later delegates the tasks, or the sub-tasks to her own personnel. In this sense new concepts would have to be added to our mechanism, in order to be closer to what happens in their reality.

❖ **Section 2:** Meeting Folder, Agenda, and Minutes

The experts evaluated the concepts as good and adequate in most of the activities. The effectiveness and satisfaction had many details discussed and commented.

The aspect of efficiency was critical in these activities more than in the others. Experts raised the importance of having highly usable electronic forms for agendas, but mainly for minutes documentation. They also stressed the required high levels of ease of use and performance aspects of these forms.

Regarding the effectiveness, the idea of having these three distinct concepts of meeting folder, agenda and minutes was positively evaluated as good. On the other hand, experts suggested that some of the ideas implemented could be too simple for extremely big projects, like the example of "Integrated Projects" funded by the European commission. In this case, the three concepts of meetings, agendas and minutes would have to consider sub-divisions to be more adequate to their daily activities.

The participants also pointed out some missing data and functionality. For the case of Agendas, the time allocated to discuss each item during the meeting should be an optional input field, as well as the option to include direct links to documents. Moreover, two experts suggested that a moderator for each item should also be represented. The idea of a workspace or a similar concept was again raised by the majority. It seems that for European projects this is a “must”, which is missing in our implementation.

From the satisfaction perspective, most of the experts evaluated the mechanisms as good, except for the missing parts, which were considered important.

Some smaller comments were done regarding the input of multiple items in one single screen. One of the experts considered the implementation a little confusing and suggested that the areas which were not enabled in the first instance should be already presented in the screen, with a light gray color, and become black, after being activated. The experts commented that the items could also appear in the agenda not only as “agenda items” but also using the terminology proposed in the system, of issues, decisions and tasks.

Regarding the agendas and minutes, the idea of notification was highly evaluated.

All agreed that the idea of having a centralized repository to share this sort of information would bring benefits for the management and organization of projects, and for the understanding of people who was not involved in all activities, who, then would access the information at any time. This was considered a big advantages over the methods used by most of them of sharing information via email (the majority) or uploading documents to project and document management systems. Another positively classified aspect was the life-cycles of agendas and minutes. This feature also made evident the collaborative process.

Specifically concerning the meeting minutes, some interesting comments and critics were observed. First, the structure was mostly clearly understood, but some experts had difficulty in differentiating “issues” from “decisions” in this context. They also observed that an easier mechanism to document discussions should be provided, and that the field dedicated to that purpose was not enough to represent the diversity of discussions they need to document in meetings they participate. A graphic mechanism to input and rearrange data should also be provided.

The core functionality highly valued was the explicit representation of decision, its association to tasks, with the consequence that tasks, after the minutes being published, are sent to the personal task lists. This feature, according to the experts shows an extreme advantage when compared to the normal documents used to represent minutes in many projects, where decisions traceability is very difficult to be realized.

Another highly valued feature was the possibility of reusing items from previous agendas, minutes and mainly, from the repositories of issues, decisions and tasks. This feature was easily perceived as one of the main possibilities to allow for continuous management. But the possibility of importing only some items either from agendas or from minutes was suggested as necessary, since the implementation only offers the possibility to import the whole set of data at a time.

A missing functionality which was pointed by some of the experts was the need to document the arguments (or the rationale) that brought a group to a certain decision.

However, they also commented that this would require a deeper study, since it is a feature dependent on factors such as culture, project and technology, that is not easily supported by a system and used by meeting participants.

❖ **Section 3:** Task Negotiation

The functionality to support task negotiation was easily understood. But because the previous critic that task should be assigned to organizations and not only to individuals, the perception of the task negotiation mechanism was affected.

The experts suggested that the negotiation support is relevant for distributed project management, or even for any type of project, but not too frequent. And when necessary, the support for tasks between individuals would be fine by our mechanism. They suggested that the mechanism could be extended to represent responsibility at the organization level, allowing for negotiation between organizations, partners and not only individuals.

❖ **Section 4:** Message Integration

The integration of messages was highly evaluated, mostly as very good, with a few exceptions which were specific to the usage of Microsof Outlook.

All experts recognized the ease of use of the semi-structure mechanism, which did not affect much the experience of sending an email. Besides, they commented that since the mechanism retrieves that from the system this would facilitate some standardization for the communication.

The experts also pointed out that just like the other structured mechanisms of our implementation, this one would require a project decision and some levels of discipline.

One of the experts suggested that a similar mechanism could be an alternative to prepare agendas and minutes.

One of the questions that remained open was scalability. The experts commented that if the lists of decisions and tasks would increase too much, probably the perception of the mechanism would be different.

❖ **Section 5:** Reports

The effectiveness and satisfaction of the reports weres highly rated, as good and very good. The participants asserted that the reports are very useful tools for coordination, mainly for a bigger amount of information, as is the case of European and German projects.

Some comments concerned user interface aspects for the filter screens and for the visualization of the text-based reports were discussed. For example, highlights and colors were indicated to be not adequately explored. Improvements in these two details would make the understading of clusters of information more useful.

The highest qualified report was the graphic one, i.e. the decision report. Experts pointed that this type of methaphor used (similar to a gantt-chart) is easily understood and very useful. The colors and the graphic elements help the recognition of problems and successful points in a very easy and deductive way.

A couple of comments regarding the representation of issues and decision makers, if necessary or not, were raised. The conclusion was that half of the participants considered this information relevant and the other part was in doubt about its real use. Their suggestion was to have them configurable.

The integration of messages to the decisions was highly qualified as well and pointed out as possibility to have in one view the history and the discussions regarding that topic.

Although the experts were mostly satisfied with the reports, they suggested that other reports could improve the coordination of projects. Some experts suggested a “network” like report, with a decision in the center and tasks and emails around it. Others suggested the same metaphor (i.e. a network), but with the possibility for the user to choose which elements to visualize at the same time, including agendas, and minutes as parameters. A time view control, allowing for chronological views, similar to the decision report, but with other elements was also considered an improvement to the existing reports.

And the third idea was related to a report showing an overview of the project. In this case, the metaphor could be similar to a gantt-chart, or based on bars, showing the status of the project as a whole. In all these ideas, a common feature should be present: the highlights of problems, so that in easy way coordinators could easily recognize where to concentrate their attention. The last comment was to have reports associated to smaller sub-structures, as mentioned before, for example the workpackages representation.

❖ **Section 6:** Integration

The last discussion realized concerned the whole integration concept. Since this activity took place after going through a scenario, experts could perceive clearly the concept of integration, its benefits and drawbacks.

The negative aspects discussed were the lack of flexibility in some sub-parts, e.g. the minutes representation. For them, the integration can bring a lot of better understanding about the processes and the project, but some usability drawbacks would have to be overcome to compensate for it.

The concepts were considered natural to understand, and rather common for project management. All experts suggested that such a set of mechanism could help coordination if a decision at a project level was done, and some training organized, since the benefits require that users would have some discipline (e.g. the message integration, or the task feedbacks).

The positive aspects observed include the elements chosen to enable this integration. Tasks, emails, meetings, decisions, were clearly understood and each role in the integration view was also positively accepted. It was also clear that other elements, like documents, could be also integrated, or that a communication with other project management tools, like Microsoft Project, would improve even more the usefulness of our ideas. On the other hand, it was not clear wether this extended integration and flexibility would bring more complextity to the daily usage of such tools.

For a real setting usage, some usability aspects would have to be improved, but in general, the concepts showed their relevance and benefits.

6.6 Conclusion

This chapter discussed the evaluation of our prototype and the concepts which were part of it. The three evaluations considered different and complimentary aspects of the research and, in general, the results were very positive. Many critics and suggestions for improving our implementation were proposed, which could be used in a sequence of this research.

We would like to shortly discuss some of the limitations of the evaluations done. Although most of the results obtained were positive, we did not consider some variations of the scenario where the usage of the system could be of particular interest. For example, in our evaluation, most of the users, experts or interviewees, knew the BSCW system. Probably, an evaluation with users who are not familiar with the system would show other aspects that did not appear in our evaluations.

More importantly, the introduction of new technology, such as our proposal, impact organization practices, and work practices, which were not deeply investigated in our approach. We are aware that evaluations and improvements in these directions would be necessary if the system would be brought to a real setting. Also, the particularities of working group culture were not considered in our activities.

Finally, specific evaluations on tasks negotiation and communication improvements were done in a controlled way. For better understanding the impacts, changes, benefits and limitations of our proposal in this respect, we would need to realize other specific evaluations.

Regarding our research questions and their sub-questions, it was clear in the evaluations that the set of mechanisms proposed provide a basis for decision tracking, allowing for better coordination in the context of project development. Mainly when users and experts experienced the visualization of the reports, this aspect became very evident.

The link between the elements we have proposed, i.e. agenda, minutes and decision-task representation also proved to be useful and natural for most of the cases in project coordination. The integration of email and tasks used as means of coordination was very well evaluated, and once more, were confirmed to be adequate elements to be used for providing signs for coordination improvement. But additionally, the flexibility of visualization and relationship with other elements like meetings, workspaces and documents were seen as improvements to our proposal in this respect.

7 Conclusion

We started this document proposing the main motivations of our work, and the two research questions that guided the development of the work.

- **Research question 1:** Can a set of automated group support mechanisms that enables the link between pre-meeting, meeting and post-meeting activities, representing decisions explicitly, provide a basis for decision tracking, allowing for better coordination during project development?
- **Research question 2:** Can common electronic elements, which belong to daily project scenarios, i.e. tasks and emails, provide means for creating awareness of problems that occur during decision implementation?

The answers to these questions have many perspectives to be analysed. What we have learned with our prototype and its evaluation shows that some aspects related to the first question are valid (i.e. meeting cycle, decisions explicitly represented in the cycle, decision tracking, and consequent better coordination).

First, the set of mechanisms supporting the complete meeting-cycle was seen as an innovation from the CSCW and coordination perspective. The automation of the complete cycle shows a possibility to improve the coordination in a rather "natural" way, since the concepts that are part of our prototype are common in many project management scenarios. The new façade of this support, though, is the discipline or change required in the work conditions. Users would have first to agree to use the system in a group setting, and second, to adapt part of their way of realizing certain activities (e.g. writing an agenda, or sending a semi-structured message) to our proposed way. Nevertheless, as we discussed in our evaluation results, the automation of the processes seem to bring other benefits that might compensate for these changes to occur. One of them is documentation. Another is a natural knowledge base of communication exchange, for example.

The concept of having a decision representation is also innovative, although a few other systems discussed in chapters 2 and 3 contain a similar approach. But the most innovative part of it is the integration of this representation to daily activities that project members execute, i.e. tasks and emails. The idea of having decisions in this specific concept of meeting cycle and project management was very clear and accepted in our evaluation, although this is only one possibility of doing it.

The traceability activities proved to be a very good means for coordination. Mainly through the reports, we could observe that this concept is very understandable, useful and used in many project contexts. Thus, the link between issues, decisions and tasks in sequential way, as well as their representation in the meeting cycle, allow for various levels of traceability, although we have explored only a couple of them.

We can conclude about our first research question that we have achieved some levels of success with our approach. We have also learned that this work was an interesting way of dealing with the coordination problem and decision implementation problem, but there is a lot of space for improvements and extensions, which we will discuss in the second section of this chapter.

Regarding the second research question, the two elements chosen in our approach, i.e. tasks and emails, proved to be a natural and useful way of linking coordination concepts. But, as discussed in the evaluation results, other elements could be used in this scenario as well, for example, documents.

Thus, we can assert that we have achieve many aspects of the goals discussed in chapter 1: (i) improving the communication between decision makers and meeting participants with decisions implementers, (ii) providing means to support the track of decisions, using the data team members already generate “naturally” (i.e. tasks and emails) and (iii) improving users awareness of problems in decision implementations, through a set of reports.

Although we have got many positive results, other aspects of our ideas, concepts and implementation could evolve and become more comprehensive and adequate to the scenario of project management. For instance, regarding the concept of interdependency, ideally the relationship between issue, decision and task could clearly evolve to a complexer one. Our implementation allows that “one issue contains many decisions” and “one decision contains many tasks”, but we have learned that for big projects this relationship could be too limited. Besides the inner relationship between these elements, means of visualizing them in a more flexible way was also one aspect discussed in our evaluation. Some users suggested that a network like relationship should be present in the systems and so should be the report possibilities to visualize all of them together. In our opinion, this critic is sound. However, we think that an empirical study, specific for understanding better this aspect, would be required in order to determine the scalability, the usefulness and the usability issues involved.

Another interesting aspect which would require improvement is the representation of sub-levels of coordination and macro-levels representations. The system could help better the target users if it would contain more granular representations (e.g. workpackages in EU projects, sub-tasks representation), and macro representation (e.g. reports about the project overview, reports about the organization overview—for representing the results of many projects together).

7.1 Contributions

After the analysis provided in the previous section, we can summarize the contributions of this thesis as follows:

- This is a proposal of an integrated solution with decisions made explicit, and with clear means of providing follow-ups, using elements which are part of daily activity project scenarios.
- From the CSCW perspective, it is a contribution to propose an approach of supporting the complete meeting cycle.
- From the DSS perspective, it is a contribution because of the link to the CSCW area, dealing with the decision implementation phase. Moreover, this is an approach which is not only “raw data” oriented, but looks after other aspects related to the decision making process, such as the context of meetings, the electronic conversation associated to it and the tasks associated to a decision being implemented.

- Additionally, this proposal can serve as a starting point for further investigations either from the CSCW or from the DSS perspective. I have created the basis for project managers to learn about their own decisions, while they do their daily work. I have proposed some concrete means of obtaining this information, through the links between the different coordination elements, but more could be done. For example, in the area of semi-dynamic or dynamic generation of storytelling, process improvement analysis and information visualizations.

7.2 Future Work

This work can be extended in various ways.

From the evaluation perspective, as mentioned in chapter 6, more specific evaluations could be realized in order to observe phenomena that we did not include in our own evaluations. For example, a deeper study of the impact electronic communication has in the decision implementation process. Or a study looking at the organizational and cultural impacts this approach would generate.

From the system perspective, many extensions are foreseen. Starting from a list of features mentioned in the evaluation phase, which comprises usability details and aspects that would make some of the user's interaction easier. Among them is the possibility to execute the meeting minutes offline, which would be followed by an upload in the system, parsing the content and including it in the respective repository (i.e. issues, decisions, and tasks) automatically. Additionally, because of the integration with Microsoft Outlook regarding emails, some of our experts supported the idea of having task synchronization between BSCW and Outlook.

From the CSCW perspective, an aspect cited in our evaluation was the support of meeting activities, and not only the documentation we propose through the meeting minutes, similar to many systems cited in chapter 3 (e.g. anonymous voting, brainstorming). In this case, since most of this support should be avaliable in a real-time setting, i.e. support for *synchronous* activities, the BSCW would have to be extended, since most of its functions support only *asynchronous* activities.

A more ambitious extension of this work could be done in the direction the most recent EMS are concentrated on, i.e. multimedia integration. Our mechanisms could be integrated to video and audio analysis algorithms. Thus, providing at the same time, the coordination we propose, and richness in the meeting cycle environment. However, as we said, this sounds very ambitious, since we would be touching the ground of various research areas such as speech analysis, video analysis, just to cite a few.

Still in the area of automatic analysis, the content analysis of messages was a topic of our interest, which we could not explore. In our approach users can send semi-structured and normal messages to the centralized project repository. The normal messages do not have any explicit correlation with the rest of the repository, though. With content analysis methods, other views of the relationship between the electronic communication and other project elements could possibly provide different understandings of the project. This topic could be implemented in specific domains, for instance, making use of ontologies to classify the messages and the other project elements.

From the IS perspective, there are many possibilities to extend this work as well. For example, we could imagine new features to allow for project predictions and simulations. Based on past experience, the system could check if certain decisions and tasks plans would contain high or low risk aspects regarding time and resource, for example. This analysis and simulation would require the integration of algorithms to calculate the resource needs, similar to other project management tools. A more specific extension could be done in the task management and negotiation module. This module could be extended to specific domains, using other theoretical models for the negotiation support.

A much discussed extension of this work is related to the decision element. An often cited possibility of extension was a way to document the decision rationale integrated to our implementation. The most frequent example was the Issue Based Information Systems (IBIS) model that deals with issues, positions, and arguments for which the emphasis is on recording the argumentation process for a single decision. But even informal representations of decisions in documents, or graphics, would already enable part of this aspect. Other models could also be investigated and compared in the scenario of decision meetings, for example the questions, options, and criteria (QOC) notation (Shum & Hammond, 1994), for which assessments are relationships between options, and criteria and arguments are used to conduct discussions about the status of the entities and relationships. Or even, the Decision Representation Language (DRL), which combines and extends the two previous approaches to provide support for automatic services like dependency management, precedence management, and plausibility management.

Still analysing the extension from the decision perspective, as cited in (Power, 2004a), there won't be a single next generation for decision support. Some characteristics that will appear in decision support systems include greater customization, more integration with other information systems, integration of interactive voice and video, voice driven user interfaces, and more powerful representations including 3-D graphics and more realistic, graphical business and process simulations.

Especially regarding visualization, our implementation offers a couple of reports, but many data is collected through the processes automated in our mechanisms. An investigation on other means to visualize the relationship of this data sounds motivating. As cited in (Borges et al., 1999), visualization can be offered in many ways. A simple one is a text-based, hierarchical display of information. However, this approach does not scale well and does not convey the temporal domain. New approaches can explore emphasis on continuous temporal navigation cues or spatial cues as ficheye views.

Finally, a discussion and investigation on how this data could assist other phases of project management, for example, after a project is finished, with lessons learned, or even reuse of data in further projects, sounds promising as well. The mechanisms we have proposed offer a first step of creating a group or project memory, in an automatic way, during the usage of the system. But the extraction of data and the semantic links between the elements were topics not explored in this thesis.

7.3 Theories Revisited

The two theories used as a basis in our work, the Coordination theory (Malone & Crowston, 1994) and the Language Action Perspective (Mora et al., 1992), (Winograd & Flores, 1986) deserve some conclusive comments.

The aim of providing better means of coordination was achieved. Some of the features implemented in our system would need some refinement and further development to be more adequate to daily usage, e.g. notification, resource management and decision decomposition. Nevertheless, we have realized what (Malone & Crowston, 1994) assert: the concepts of coordination theory can help identify similarities among concepts and results in different disciplines. We have proposed one way of coordination, but users tend to re-invent technology, they develop their own way of using it for their particular purposes, and they may invent completely new purposes (Orlikowski, 1996).

The objective of using the LAP in order to improve communication possibilities was also achieved to some extent. We developed two relevant features which used aspects of communication: the task negotiation mechanism and the semi-structured messages. These two features enable a bridge between acting and communicating, which is in accordance with the idea of (Winograd & Flores, 1986) that "cooperative work is coordinated by the performance of language actions which are speech acts". In our evaluation, this aspect was positively seen by users and experts. However, a deeper analysis specifically about this topic could show other façades of this integration. As (Michelis & Grasso, 1994) suggest "the complexity on human communication is both the ultimate cause of all the breakdowns occurring within a group of persons and the space where they can afford them as opening new possibilities".

7.4 In conclusion

Any scientific theory (indeed, any statement about the world) must neglect some things, in order to focus on others (Malone & Crowston, 1994). The reality implies that many aspects are addressed, but in scientific work, very often, a set of aspects might be more relevant than others, in order to make studies feasible. This is also the case of the work discussed in this thesis. Following this natural limitation, dealing with the problem of decision follow up, we have chosen some aspects and have discarded others. This choice comprises a natural *exclusion*, which comes from the Latin word *excludere*, meaning that if you choose one path, other paths are automatically out of choice.

Thus, we would like to finalize this thesis saying that we hope this work can bring contributions and lessons learned to the related communities, serving to indicate paths that could be discarded, and paths that could be followed…

8 References

(Aalst & Hee, 2002) Aalst, W.v.d., Hee, K.v., “Workflow Management”, Cambridge, The MIT Press, 2002.

(Abecker et al., 1999) Abecker, A. Bernardi, A. and Sintek, M. “Proactive Knowledge Discovery for Enterprise Knowledge Management” in: Proceedings of the 11th Software Engineering and Knowledge Engineering Conference - SEKE’99, June 16-17, Kaiserslautern, Germany (1999), Lecture Notes in Computer Science, pp. 103 – 117.

(Advizor, 2005) http://www.advizorsolutions.com, consulted in January 2005

(Aggarwal, 2001) Aggarwal, A.K., “Taxonomy of Sequential Decision Support Systems”, in: Proceedings of the 2001 Informing Science, June 19-22, Krakow, Poland (2001), pp. 1-12.

(Ackerman & Halverson, 1998) Ackerman, M. S., Halverson, C. “Considering na Organizational’s Memory”, in: Proceedings of the Conference on Computer Supported Cooperative Work – CSCW’98, November 14-18, Seatle, Washington, USA (1998), ACM Press, New York, pp. 39-48.

(Alter, 1980) Alter, S., “Decision Support Systems: current practices and continuing challenges”, Addison-Wesley Publishing Company, 1980.

(Anthony, 1965) Anthony, R.N., “Planning and Control Systems: A Framework Analysis”. Boston Harvard University Press, 1965.

(Appelt & Koch, 1998) Appelt, W., Koch, T., “Beyond Web Technology - Lessons Learnt from BSCW”, in: Proceedings of IEEE 7th International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WET ICE), June 17-19, Stanford University, California, USA (1998).

(Antunes et al., 1999) Antunes, P., Ho, T, Carrio, L,. "A GDSS Agenda Builder for Inexperienced Facilitators", in: Proceedings of the 10th EuroGDSS Workshop. June, Copenhagen, Denmark (1999), Delft University of Technology, pp. 1-15.

(Araujo, 2000) Araujo, R.M. “Ampliando a Cultura de Processos de Software - Um enfoque baseado em Groupware e Workflow” – Ph.D. Thesis, COPPE Sistemas, Federal University of Rio de Janeiro, 2000.

(AskMe, 2005) http://www.askmecorp.com
consulted in January 2005

(Auramäki et al., 1992) Auramäki, E., Hirschheim, R. and Lyytinenm K. "Modelling Offices Through the Discourse Analysis: The SAMPO Approach". The Computer Journal, 35(4), 1992, pp. 342-352.

(Austin, 1962) Austin, J.L. "How to do Things with Worlds", Harvard University Press, Cambridge, MA, 1962.

(Autonomy, 2005) http://www.autonomy.com
consulted in January 2005

(Balasubramanian et al., 1999) Balasubramanian, P., Nochur, K., Henderson, J.C., Kwan, M.M., "Managing process knowledge for decision support", Decision Support System, 27 (1-2), 1999, pp. 145-162.

(Basoglu et al., 2001) Basoglu, A. N., Darcan, O. and Ozdemir, O. "A Framework for a Web-based Group Decision Support System", in: Proceedings of the 2001 Informing Science, June 19-22, Krakow, Poland (2001), pp.88-93.

(Beard et al. 1990) Beard, D., Palanlappan, M., et al., "A visual calendar for scheduling group meetings", in: Proceedings of the Conference on Computer Supported Cooperative Work – CSCW'90, October 7-10, Los Angeles, California, USA (1990), ACM, N.Y., pp. 279-290.

(Begole et al., 1999) Begole, J., Rosson, M.B., and Shaffer, C.A., "Flexible Collaboration Transparency: Supporting worker Independence in replicated application-sharing systems", ACM Transactions on Computer-Human Interaction, vol. 6, no. 2, June 1999, pp. 95-132.

(Belbin, 2005) http://www.belbin.com
consulted in February 2005

(Bentley et al., 1997) Bentley, R., Appelt, W., Busbach, U., Hinrichs, E., et al., "Basic Support for Cooperative Work on the World Wide Web", in: International Journal of Human Computer Studies: Special issue on Novel Applications of the WWW, 46, 1997, Academic Press, Cambridge, pp. 827-846.

(Borges et al., 1999) Borges, M.R.S, Pino, J.A., Fuller, D., and Salgado, A.C., "Key issues in the design of an asynchronous system to support meeting preparation", Decision Support Systems, vol. 27 (3), 1999, pp. 271-289.

(Borges et al., 2000) Borges, M.R.S., Pino, J.A., and Salgado, A.C., "Requirements for Shared Memory in CSCW Applications", in: Proceedings of

	WITS'2000, 10th Annual Workshop on Information Technologies and Systems, Brisbane, Australia, December 2000, pp. 211-216.
(Borges, Pino and Valle, 2002)	Borges, M., Pino, J.A., and Valle, C., "On the implementation and follow-up of decisions", in: Proceedings of DSIAge 2002, July 04-07, Cork, Ireland (2002), pp. 366-376.
(Borges, Pino and Valle, 2005)	Borges, M., Pino, J.A., and Valle, C., "Support for decision implementation and follow-up", European Journal of Operational Research, vol. 160 (2), 2005, pp. 336-352.
(Bostrom, 1989)	Bostrom, R.P., "Successful application of communication techniques to improve the systems development process", Information and Management, vol. 16 (5), 1989, pp. 279-295.
(Bowers & Churcher, 1989)	Bowers, J. and Churcher, J. "Local and Global Structuring of Computer Mediated Communications: Developing Linguistic Perspectives on CSCW". in Cosmos Office: Technology and People, 4(3), pp. 197-227.
(BSCW, 2005)	http://bscw.fit.fraunhofer.de consulted in June 2005
(BSCW Administrator Manual, 2005)	http://bscw.fit.fraunhofer.de consulted in June 2005
(BSCW X-API, 2005)	http://www.fit.fraunhofer.de/~elke/bscw-xapi/documentation/X-BSCW.html consulted in July 2005
(Bui & Jarke, 1986)	Bui, T. X., Jarke, M., "Communications Design for Co-oP: A Group Decision Support System", ACM Transactions on Information Systems, vol. 4(2), 1986, pp. 81-103.
(Carrol, 1995)	Carroll, J.M. "The scenario perspective on system development". In J.M. Carroll (ed.) Scenario-based design: envisioning work and technology in system development. J. Wiley, New York, 1995.
(Casati et al., 2000)	Casati, F., Castano, S., Fugini, M. Mirbel, I. and Pernici, B., "Using Patterns to Design Rules in Workflows", IEEE Transactions on Software Engineering, vol. 26, no. 8, 2000, pp. 760-784.
(Chen et al., 2004)	Chen, H., Finin, T., Joshi, A., et al., "Intelligent Agents Meet the Semantic Web in Smart Spaces," IEEE Internet Computing, vol. 08, no. 6, November/December 2004, pp. 69-79.

(Chiu et al., 2001)	Chiu, P., Boreczky, J., Girgensohn, A., Kimber, D., "LiteMinutes: An Internet-Based System for Multimedia Meeting Minutes", in: Proceedings of the 10th international conference on World Wide Web, May 02-05, Hong Kong, China (2001), pp. 140-149.
(Clawson & Bostrom, 1993).	Clawson, V., and Bostrom, R.. "The facilitation role in group support systems", in: Proceedings of the 1993 Conference on Computer Personnel Research, April 01-03, St. Louis, Missouri, USA (1993), pp. 323-335.
(Conklin & Begeman, 1988)	Conklin, J., and Begeman, M.L. "gIBIS: a hypertext tool for exploratory policy discussion", in: Proceedings of the Conference on Computer Supported Cooperative Work – CSCW'88, September 26-28, Portland, Oregon, USA (1988), ACM Press, New York, pp. 140-152.
(Conklin, 1997)	Conklin, E.J. "Designing Organizational Memory: Preserving Intellectual Assets in a Knowledge Economy", 1997. Retrieved in June 2005 from http://www.cognexus.org
(Conklin, 2005)	Conklin, J. "Wicked Problems and Social Complexity" *in the forthcoming book* "Dialog Mapping: Making Sense of Project Fragmentation", Wiley, October 2005.
(Courtney, 2001)	Courtney, J.F., "Decision making and knowledge management in inquiring organizations: toward a new decision-making paradigm for DSS". Decision Support System, 31 (2001), pp. 17-38.
(Croft & Lefkowitz, 1988)	Croft, W.B. and Lefkowitz, L.S., "Using a planner to support office work", in: Proceedings of the ACM Conference on Office and Information Systems, March 23-25, Palo Alto, CA, USA, ACM Press, New York, pp. 55-62.
(Crystal, 2005)	http://www.decisioneering.com consulted in January 2005
(Cybozu, 2005)	http://www.cybozu.com consulted in January 2005
(Cecez-Kecmanovic & Janson, 1999)	Cecez-Kecmanovic, D. and Janson, M., "Communicative Action Theory: An Approach to Understanding the Application of Information Systems", in: Proceedings of the 10th Australasian Conference on Information Systems ACIS'99, Wellington, New Zealand (1999), pp.183-195.
(Davenport, 1996)	Davenport, G. "Smarter Tools for storytelling: are they just around the corner?", IEEE Multimedia, Spring 1996. pp.10-14.

(Davenport & Prusak, 1998) Davenport, T., and Prusak, L., "Working Knowledge: How organizations manage what they know", Harvard Business School, 1st ed. 1998.

(De Sanctis & Gallupe, 1987) DeSanctis, G. and Gallupe, B., "A Foundation for Study of Group Decision Support Systems", Management Science, vol. 33 (5), 1987, pp. 589-609.

(DeSanctis & Dickson, 1987) DeSanctis, G. and Dickson, G. W., "GDSS: a shell system in support of a program of research", in: Proceedings of the 19th Hawaii International Conference on System Sciences (HICSS-19), Hawaii, USA, 1987.

(Dickson et al, 1989) Dickson, G.W., Lee, J.E., Robinson, L. and Heath, R. "Observations on GDSS interaction: Chauffeured, facilitated and user-driven systems", in: Proceedings of the 22nd Annual Hawaii International Conference on System Sciences (HICSS-22), January, Kailua Kona, Hawaii, USA (1989), pp. 337-343.

(Documentum, 2005) http://www.documentum.com consulted in January 2005

(Dourish & Bellotti, 1992) Dourish, P., and Bellotti, V., "Awareness and coordination in shared spaces", in: Proceedings of the Conference on Computer Supported Cooperative Work – CSCW'92, November 01-02, Toronto, Ontario, Canada (1992), ACM Press, New York, pp. 107-114.

(Dyson, 2004) Dyson, E., "Meta-mail: A Medium for Meaning", June, 2004, Retrieved in June 2005 from http://www.edventure.com/release1/abstracts.cfm?Counter=8229630

(Dzida & Freitag, 1999) Dzida, W., and Freitag, R., "Documentation of Prototypes in Terms of Use Scenarios: Nice to Have to Indispensable?" In: Bullinger, H.-J., Ziegler, J. (eds.). Human-Computer Interaction, Volume I, 905–908. Lawrence Erlbaum Associates, 1999.

(Ellis et al., 1991) Ellis, C.A., S. Gibbs, and G.L. Rein, "Groupware - Some Issues and Experiences", Communications of the ACM, 1991. 34(1), pp. 38-58.

(Entopia, 2005) http://www.entopia.com/information_center.html consulted in January 2005

(Enzer, 2002) Enzer, M., "Glossary of internet terms", retrieved in June 2005 from http://www.matisse.net/files/glossary.html

(Exsys, 2005) http://www.exsys.com/ consulted in January 2005

(Facilitate.com, 2005) http://www.facilitate.com, consulted in January 2005

(Fialho, 1998) Fialho, L. "TASKER: Sistema de Coordenação de Equipe". Master Thesis of the COPPE Sistemas, Federal University of Rio de Janeiro, Brazil, 1998.

(Garcia et al., 2004) Garcia, A.C.B, Kunz, J. and Fischer M. "Cutting to the Chase: Improving Meeting Effectiveness by Focusing on the Agenda", in: Proceedings of the Conference on Computer Supported Cooperative Work – CSCW'2004, November 06-10, Chicago, Illinois, USA (2004), ACM Press, New York, pp. 346-349.

(Google, 2005) http://www.google.com consulted in June 2005

(Gory & Scott Morton, 1975) Gorry, G. A., and Scott Morton, M.S., "A Framework for Management Information Systems", Information for Decision Making: Quantitative and Behavioral Dimensions. Alfred Rappaport (ed.). Second Edition. Prentice-Hall Inc, 1975, pp. 16-30.

(Gotel, 1995) Gotel, O. "Contribution Structures for Requirements Traceability". London, England: Imperial College, Department of Computing, Ph.D. Thesis, 1995.

(Groove, 2005) http://www.groove.net/index.cfm/pagename/Tools/, consulted in January 2005

(Gross, 2001) Gross, T., "Towards Ubiquitous Awareness: the PRAVTA Prototype", in: Proceedings of the Ninth Euromicros Workshop on Parallel and Distributed Processing (PDP 2001), February 7-9, Mantova, Italy (2001), IEEE Computer Society Press, Los Alamitos, CA, pp 139-146.

(Gross & Prinz, 2003) Gross, T. and Prinz, W. "Awareness in Context: A Light-Weight Approach", in: Proceedings of the 8th European Conference on Computer-Supported Cooperative Work (ECSCW 2003), September 14-18, Helsinki, Finland (2003), Kluwer Academic Publishers, Dortrecht, NL, pp. 295-314.

(Grudin, 1988) Grudin, J. "Why CSCW applications fail: Problems in the design and evaluation of organizational interfaces", in: Proceedings of the Conference on Computer Supported Cooperative Work – CSCW'88, September 26-28, Portland, Oregon, USA (1988), ACM Press, New York, pp. 85-93.

(Grudin, 1994) Grudin, J. "CSCW: History and focus", IEEE Computer, 27(5), 1994, pp. 19-26.

(Grudin & Palen, 1995) Grudin, J. and Palen, L. “Why groupware succeeds: Discretion or mandate?”, in: Proceedings of the 4th European Conference on Computer-Supported Cooperative Work (ECSCW 1995), September 11-15, Stockholm, Sweden (1995), Kluwer Academic Publishers, Dortrecht, NL, pp. 263-278.

(Grudin & Poltrock, 1997) Grudin, J. and Poltrock, S.E., “Computer-supported cooperative work and groupware”, in M. Zelkowitz (Ed.), “Advances in Computers”, vol. 45, Orlando, FL: Academic Press. 1997, pp. 269-320.

(Gutwin et al., 1996) Gutwin, C, Greenberg, S., and Roseman, M. “Supporting Awarenes of Others in Groupware”, in Conference Companion of the Conference on Human Factors in Computing Systems (CHI '96), April 13-18, Vancouver, BC, Canada (1996), ACM Press, New York, page 205.

(Gwizdka, 2002a) Gwizdka, J., “TaskView: design and evaluation of a task-based email interface”, in: Proceedings of the 2002 conference of the Centre for Advanced Studies on Collaborative research, Toronto, Ontario, Canada (2002), page 4.

(Gwizdka, 2002b) Gwizdka, J., “Reinventing the Inbox – Supporting the Management of Pending Tasks in Email”, in: Proceedings of the Conference on Human Factors in Computing Systems (CHI '2002), April 20-25, Minneapolis, Minnesota, USA (2002), ACM Press, New York, pp. 550-551.

(Gwizdka, 2004) Gwizdka, J., “Email Task Management Styles: The Cleaners and the Keepers”, in: Proceedings of the Conference on Human Factors in Computing Systems (CHI '2004), April 24-29, Vienna, Austria (2004), ACM Press, New York, pp. 1235-1238.

(Habermas, 1984) Habermas, J., “The Theory of Communicative Action”, Vol. 1: Reason and the Rationalisation of Society. Heinemann, London, 1984.

(Hammond et al., 1999) Hammond, J.S., Keeney, R. L. and Raiffa H., “Smart Choices: A Practical Guide to Making Better Decisions”, Harvard Business School Press, Boston, Massachusetts, 1999.

(Harvey & Myers, 1995) Harvey, L. and Myers, M.D., “Scholarship and practice: the contribution of ethnographic research methods to bridging the gap”, Information Technology & People, 1995, vol 8 (3), pp. 13-27.

(Hayne, 1999) Hayne, S. C. “The Facilitators Perspective on Meetings and Implications for Group Support Systems Design”, The Data Base for Advances in Information Systems, Summer-Fall, 1999, vol. 30, no. 3-4, pp. 72-91.

(IBM, 2005) http://www-306.ibm.com/software/lotus/
consulted in January 2005

(ISO 13407) ISO 13407:1999. Human-centred design processes for interactive systems. Ed. 1, 1999.

(ISO 9241-11) ISO 9241-11. Ergonomic requirements for office work with visual display terminals (VDTs) – Part 11: Guidance on usability. Ed. 1, 1998.

(ISO 9126-1) ISO 9126-1. Software Engineering – Product quality – Part 1: Quality model, Ed. 1, 2001.

(Ijiri et al., 1975) Ijiri, Y., Jaedicke R.K and Knight, K.E. "The Effects of Accounting Alternatives on Management Decisions" in Information for Decision Making: Quantitative and Behavioral Dimensions. Alfred Rappaport (ed.), Second Edition. Prentice-Hall Inc., 1975, pp. 263-271.

(Jain et al., 2003) Jain, R., Kim, P., Li, Z. "Experiential Meeting System", in: Proceedings of the 2003 ACM SIGMM Workshop on Experiential Telepresense (ETP), November 7, Berkeley, California, USA (2003). ACM Press, New York, pp. 1-12.

(Java Sun, 2005) http://java.sun.com/
consulted in June 2005

(Jay, 1976) Jay, A., "How to Run a Meeting", Harvard Business Review, 1976, pp. 43-57.

(Johansen, 1988) Johansen, R., "Groupware: Computer Support for Business Teams", Free Press, New York, 1988.

(Keen & Scott Morton, 1978) Keen, P.G.W. and Scott Morton, M.S., "Decision Support Systems: An Organizational Perspective", Reading, MA: Addison-Wesley, Inc. 1978.

(Keen, 1987) Keen, P.G.W., "Decision Support Systems: The Next Decade", Decision Support Systems, 3 (1987), pp. 253-265.

(Kerr & Wilcox, 2004) Kerr, B. and Wilcox, E., "Designing Remail: Reinventing the Email Client Through Innovation and Integration", IBM Research Report, RC23127 (W0402-146) February 27, 2004. Retrieved in June 2005 from http://www.research.ibm.com/remail/publications.html

(Kleidorfer et al., 1993) Kleidorfer, P., Kunreuther, H.C., and Schoemaker, P.J.H., "Decision sciences: an integrative perspective", Cambridge

University Press, UK, 1993.

(Koh, 2005) Koh, J.C.P., "Meeting follow up tool for BSCW", Master thesis, RWTH Aachen, July 2005.

(Kraemer & King, 1988) Kraemer, K.L., and King, J.L., "Computer-based systems for Cooperative Work and Group Decision Making", ACM Computing Surveys, Vol 20., No. 2, June 1988, pp. 115-146.

(Kreifelts et. al., 1993) Kreifelts, T., Hinrichs, E., Woetzel, G. "Sharing To-Do Lists with a Distributed Task Manager", in: Proceedings of the 3rd European Conference on Computer-Supported Cooperative Work (ECSCW 1993), September 13-17, Milano, Italy (1993), Kluwer Academic Publishers, Dortrecht, NL, pp 31- 46.

(Kunz & Rittel, 1970) Kunz, W. and H. W. J. Rittel, "Issues as Elements of Information Systems" (Working Paper), Berkeley, CA: Center for Planning and Development Research, University of California, 1970.

(Lee, 1990) Lee, J., "Sibyl: a qualitative decision management system", in Artificial Intelligence at MIT: Expanding Frontiers, vol. 1, MIT Press, Cambridge, Massachusetts, USA, 1990.

(Levitt & Mahowald, 2002) Levitt, M., and Mahowald, R.P., "There should be More to Collaboration than Email", an IDC White Paper Sponsored by Groobe Networks, retrieved in June 2005 from: http://www.groove.net/pdf/idc-collaboration.pdf

(Lewicki et al., 1992) Lewicki, R., Weiss, S., and Lewin, D. "Models of conflict, negotiation and third party interventions: A review and synthesis", Journal of Organizational Behavior, 1992, vol. 13, pp. 209-252.

(Mackay, 1988) Mackay, W., "More than just a communication system: diverstiy n the use of Electronic Mail", in: Proceedings of the Conference on Computer Supported Cooperative Work – CSCW'88, September 26-28, Portland, Oregon, USA (1988), ACM Press, New York, pp. 26-28.

(Maguire & Bevan, 2002) Maguire, M., Bevan, N. "User Requirements Analysis – A Review of Supporting Methods", in: Proceedings of IFIP 17th World Computer Congress, August, Montreal, Canada (2002), 25-30, Kluwer Academic Publishers, p133-148.

(Malone et al, 1986a) Malone, T.W., Grant, K.R., and Turbak, F.A. "The information lens: an intelligent system for information sharing in organizations", in: Proceedings of the Conference on Human Factors in Computing Systems (CHI '1986), April 13-17, Boston, Massachussets, USA (1986), ACM Press, New York, pp. 1-8.

(Malone et al., 1986b) Malone, T., Grant. K, Lai, K., Rao, R., Rosenblitt, D. "Semi-structured messages are surprisingly useful for computer-supported coordination", in: Proceedings of the Conference on Computer Supported Cooperative Work – CSCW'86, December 03-05, Austin, Texas, USA (1986), ACM Press, New York, pp. 102-114.

(Malone & Crowston, 1990) Malone, T. W. and Crowston, K., "What is Coordination Theory and How Can It Help Design Cooperative Work Systems?", in: Proceedings of the Conference on Computer Supported Cooperative Work – CSCW'1990, October 07-10, Los Angeles, California, USA (1990), ACM Press, New York, pp. 357-370.

(Malone & Crowston, 1994) Malone, T. W. and Crowston, K., "The Interdisciplinary Study of Coordination", ACM Computing Surveys, March 1994, vol. 26(1), pp. 87-119.

(Malone et al., 1995) Malone, T. W., Lai, K.-W. and Fry, C., "Experiments with Oval: A Radically Tailorable Tool for Cooperative Work", ACM Transactions on Information Systems, April 1995, vol. 13(2), pp. 177-205.

(Malone et al., 1999) Malone, T.W., Crowston, K., Lee, J. and Pentland, B., "Tools for inventing organizations: Toward a handbook of organizational process", Management Science, 1999, vol. 45, pp. 425-443.

(Mark & Poltrock, 2003) Mark, G. Poltrock, S.E. "Shaping technology across social worlds: groupware adoption in a distributed organization", in: Proceedings of the 2003 International ACM SIGGROUP Conference on Supporting Group Work (GROUP 2003), November 9-12, Sanibel, Florida, USA (2003), ACM Press, New York, pp. 284-293.

(Markulla, 1999) Markulla, M. "Knowlegde Management in Software Engineering Projects", in: Proceedings of the 11th Software Engineering and Knowledge Engineering Conference - SEKE'99, June 16-17, Kaiserslautern, Germany (1999), Lecture Notes in Computer Science, pp. 20-27.

(Mason, 1975) Mason, R.O., "Basic Concepts for Designing Management Information Systems" in Information for Decision Making: Quantitative and Behavioral Dimensions. Alfred Rappaport (eds). Second Edition. Prentice-Hall Inc.1975, pp. 02-15.

(Meeting Corpus, 2005) http://www.icsi.berkeley.edu/Speech/mr/ consulted in February 2005

(Meeting Maker, 2005) http://www.meetingmaker.com
consulted in February 2005

(Meeting Room, 2005) http://penance.is.cs.cmu.edu/meeting_room/
consulted in February 2005

(Michelis & Grasso, 1994) Michelis, G. and Grasso, A. “Situating conversations within language/action perspective: the Milan Conversation Model”, in: Proceedings of the Conference on Computer Supported Cooperative Work – CSCW’1994, October 22-26, Chapel Hill, North Carolina, USA (1994), ACM Press, New York, pp. 89-100.

(Microsoft, 2005) http://www.microsoft.com/sharepoint/server/default.asp,
consulted in January 2005

(Microstrategy, 2005) http://www.microstrategy.com
consulted in January 2005

(Mora et al., 1992) Mora, R., Winograd, T., Flores, R. and Flores, F. “The Action Workflow Approach to Workflow Management Technology”, in: Proceedings of the Conference on Computer Supported Cooperative Work – CSCW’1992, November 01-04, Toronto, Ontario, Canada (1992), ACM Press, New York, pp. 281-288.

(Mitroff & Linstone, 1993) Mitroff, I.I., Linstone, H.A., “The Unbounded Mind: Breaking the Chains of Traditional Business Thinking”, 1993, Oxford Univ. Press, New York.

(M4, 2005) http://www.m4project.org/
consulted in February 2005

(Myers, 1997) Myers, M. D., "Qualitative Research in Information Systems," MIS Quarterly (21:2), June 1997, pp. 241-242. MISQ Discovery, archival version, June 1997, MISQ Discovery, updated version, last modified retrieved in June 2005 from http://www.qual.auckland.ac.nz

(Niederman & Volkema, 1996) Niederman, F. and Volkema, R., “Influence of agenda creation and use on meeting activities and outcomes: report on initial results”, in: Proceedings of the 1996 ACM SIGCPR/SIGMIS conference on Computer personnel research (SIGCPR’96), April 11-13, Denver, Colorado, USA (1996), pp. 192 – 205.

(Niederman & Volkema, 1999) Niederman, F., Volkema, R.J., “The Relation of Agenda Creation and Use to Group Support System Experience”, The Data Base for Advances in Information Systems – Summer-Fall 1999, vol. 30, no. 3,4, pp. 51-71.

(Nielsen, 1993) Nielsen, J. “Usability Engineering”, 1993, Academic Press, Boston, Massachussets, USA.

(Nonaka & Takeuchi, 1995)	Nonaka, I., and Takeuchi, H., “The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation”, 1995, Oxford University Press, Oxford.
(Nunamaker & Briggs, 1991)	Nunamaker, J.F., and Briggs, R.O. “Lessons From a Dozen Years of Group Support Systems Research: A Discussion of Lab and Field Findings”, Journal of Management Information Systems, vol. 13(3), 1991, pp. 163-205.
(Nunamaker et al., 1991)	Nunamaker, J.F., Dennis, A.R., Valacich, J.S., Vogel, D.R., George, J.F., “Eletronic Meeting Systems”, Communications of the ACM, July 1991, Vol 34(7), pp 44-6.
(Nunamaker et. al, 1995)	Nunamaker, J.F. Jr., Briggs, R.O., and Mittleman, D., “Eletronic Meeting Systems: Ten Years of Lessons Learned”, in Groupware: Technology and Applications, Englewood Cliffs Coleman, D. and R. Khanna (eds), NJ: Prentice-Hall PTR, 1995, pp. 146-193.
(Nunamaker et al., 1996)	Nunamaker, J.F., Briggs, R.O and Mittleman, D.D. “Lessons from a Decade of Group Support Systems Research”, in: Proceedings of 29th Annual Hawaii International Conference on System Sciences (HICSS-29), January 3-6, Maui, Hawaii, USA (1996), vol. 3, pp 418-427.
(O’Leary, 1998)	O’Leary, D., “Enterprise Knowledge Management”, IEEE Computer, March 1998, pp. 54-61.
(Olson & Olson, 1991)	Olson, G. M., and Olson, J. S., “User-Centred Design of Collaboration Technology”, in Journal of Organisational Computing, 1991, vol. 1(1), pp. 61-83.
(Olson & Olson, 1997)	Olson, G. M., and Olson, J. S., “Research on Computer Supported Cooperative Work” , in Helander, M., Landauer, T. K., Prabhu, P. (eds.) Handbook of Human Computer Interaction, 1997, Elsevier Science B.V., p.1433-1456.
(Orbiteam, 2005)	http://www.orbiteam.de consulted in June 2005
(Orlikowski, 1996)	Orlikowski, W., "Improvising Organizational Transformation over Time: A Situated Change Perspective," Information Systems Research, vol. 7 (1), 1996, pp. 63-92.
(Palen, 1999)	Palen, L., "Social, Individual & Technological Issues for Groupware Calendar Systems", in: Proceedings of the Conference on Human Factors in Computing Systems (CHI '1999), May 15-20, Pittsburgh, Pennsylvania, USA (1999),

ACM Press, New York, pp. 17-24.

(Palen & Grudin, 2002) Palen, L.A. and Grudin, J. “Discretionary Adoption of Group Support Software”. in Munkvold, B.E. (eds.), Implementing Collaboration Technologies in Industry: Case Examples and Lessons Learned, Springer Verlag, London, UK, 2002.

(Power, 1998) Power, D.J., “Web-based Decision Support Systems”, Part I and II. The online Executive Journal for Data-Intensive Decision Support, August 18 and 25, 1998. Vol 2. Nos. 33 and 34, retrieved in June 2005 from: http://www.taborcommunications.com/dsstar/backissues/1998.html

(Power, 1999) Power, D.J., “A Brief History of Decision Support Systems”, 1999, DSSResources.COM, retrieved in January 2005 from http://DSSResources.COM/history/dsshistory.html

(Power, 2000) Power, D. J., “Model-Driven and Web-based Decision Support Systems: Concepts and Issues”, in: Proceedings of Americas Conference on Information Systems (AMCIS’2000), Long Beach, California, August 10th-13th, 2000.

(Power, 2004a) Power, D.J., “Decision Support Systems: Frequently Asked Questions”, Editor DSSResources.COM, E-Book: ISBN: 0-595-78757-6, December 2004.

(Power, 2004b) Power, D. J. “Free Decision Support Systems Glossary”, DSSResources.COM, http://DSSResources.COM/glossary/ consulted in December 2004

(Prinz, 1999) Prinz, W., “NESSIE: An Awareness Environment for Cooperative Settings”, in: Proceedings of the 6th European Conference on Computer-Supported Cooperative Work (ECSCW 1999), September 12-16, Copenhagen, Denmark (1999), Kluwer Academic Publishers, pp. 391-410.

(Proclarity, 2005) http://www.proclarity.com consulted in January 2005

(Python, 2005) http://www.python.org consulted in June 2005

(Radicati, 2005) http://www.radicati.com/ consulted in February 2005

(Rainkundalia, 1998) Raikundalia, G. K., "A Web Tool for Asynchronous Collaborative Development of Electronic Meeting Agendas", in: Proceedings of 1998 Asia-Pacific Conference on Human-Computer Interaction (APCHI '98), July 15-17, Kanagawa, Japan, 1998, IEEE Computer Society, pp. 374-379.

(Ramage, 1999) Ramage, M., "The Learning Way: Evaluating Co-operative Systems", Doctor of Philosophy Thesis, Department of Computing, Lancaster University, 1999.

(Rendemption, 2005) http://www.dimastr.com/redemption/ consulted in June 2005

(Rodriguez & Favela, 2001) Rodriguez, J. and Favela, J., "Work Environments in Electronic Meeting Systems", in: Extended abstracts of the Proceedings of the Conference on Human Factors in Computing Systems (CHI '2001), Developing Consortium, March-April, Seattle, Washington, USA (2001), ACM Press, New York, pp. 57-58.

(Ruggles, 2002) Ruggles, R., "The Role of Stories in Knowledge Management", Journal of Storytelling and Business Excellence, retrieved in February 2002 from http://www.storytellingcenter.com/articles.htm

(Rumbaugh et al., 1999) Rumbaugh, J., Jacobson, I., Booch, G., "The Unified Modeling Language Reference Manual", Addison-Wesley Longman, Inc., 1999.

(Russo, 1989) Russo, J.E., Schoemaker, P.J.H., "Decision traps: ten barriers to brilliant decision-making and how to overcome them", 1989, New York, Simon and Schuster.

(Santhanam et al., 2000) Santhanam, R., Guimaraes, T., and George, J.F. "An empirical investigation of ODSS impact on individuals and organizations", Decision Support Systems, 30 (2000), pp. 51-72.

(Schultz et al., 2001) Schultz, T., Waibel, A., Bett, M., Metze, F., Pan, Y. Ries, K. et al., "The ISL Meeting Room System", in: Proceedings of the Workshop on Hands-Free Speech Communication (HSC-2001), Kyoto Japan, April 2001.

(Searle, 1969) Searle, J. R. "Speech acts: An essay in the philosophy of language", Cambridge University Press (January 2, 1969).

(Shell, 2001) "Story-Telling in Shell: Managing Knowledge through New Ways of Working", November 2001, Shell International Exploration and Production, Orgnanizational Performance and Learning, Rijswijk, The Netherlands, retrieved in June 2005 from http://www.knowledgeboard.com.

(Shim et al., 2002) Shim, J.P., Warkentin, M., Courtney, J.F., et al., "Past, present and future of decision support technology", Decision Support Systems, vol. 33, 2002, pp. 111-126.

(Simon, 1976) Simon, H. A., “Administrative Behavior”, 3rd Edition, New York, The Free Press, 1976.

(Simon, 1977) Simon, H. A., “The New Science of Management Decision”, New York, Harper & Row, 1977.

(Simon, 1996) Simon J. B. S. et al., “Graphical argumentation and Design Cognition”, Technical Report KMI-TR-25, Knowledge Media Institute, 1996, retrieved in February 2005 from http://kmi.open.ac.uk/publications/

(Simone & Divitini, 1999) Simone, C. and Divitini, M., “Integrating Context to Support Coordination: The CHAOS Project”, Computer Supported Cooperative Work, 1999, vol 8, Kluwer Academic Publishers, pp. 239-283.

(Sridhar, 1998) Shridar, S., “Decision Supoort using the Intranet”, Decision Support Systems, 23 (1998), pp. 19-28.

(Stefik et al., 1987) Stefik, M., Foster G., Bobrow, D.G., et al. “Beyond the chalkboard: computer support for collaboration and problem solving in meetings”, Communications of the ACM, January 1987, vol. 30 (1), pp 32-47.

(Streitz et al., 1994) Streitz, N., Geissler, J., Haake, J. M. And Hol, J., “DOLPHIN: integrated meeting support across local and remote desktop environments and LiveBoards”, in: Proceedings of the Conference on Computer Supported Cooperative Work – CSCW’1994, October 22-26, Chapel Hill, North Carolina, USA (1994), ACM Press, New York, pp. 345-358.

(Surysekar & Ramesh, 2001) Surysekar, K., and Ramesh, B., “On managerial incentives for process knowledge capture and use”, in: Proceedings of 34th Annual Hawaii International Conference on System Sciences (HICSS-34), January 3-6, Maui, Hawaii, USA (2001), vol. 3, p. 3043.

(Svenson, 1992) Svenson, O., “Differentiation and Consolidation Theory of human decision making: A frame of reference for the study of pre- and post-decision processes”, Acta Psychologica 80 (1992) pp. 143-168.

(Svenson & Benthorn, 1992) Svenson, O. and Benthorn, L.J., “Consolidation processes in decision making: post-decision changes in attracttiveness of alternatives”, Journal of Economic Psychology, 13 (1992), pp. 315-327.

(Schoop, 2001) Schoop, M. “An introduction to the language-action perspective”, in ACM SIGGROUP Bulletin, Vol. 22, Issue 2, August 2001, pp 3-8.

(Schoop et al., 2003) Schoop, M., Jertila, A. and List, T., “Negoisst: A Negotiation Support System for Electronic Business-to-Business Negotiations in E-Commerce”, Data and Knowledge Engineering, 2003, 47(3), pp 371-401.

(Schoop, 2004) Schoop, M., “The Worlds of Negotiation”, in: Proceedings of the 9th International Working Conference on the Language-Action Perspective on Communication Modelling (LAP 2004). Rutgers University, The State University of New Jersey, New Brunswick, New Jersey, USA (2004), June 2-3, M. Aakhus, M. Lind (eds.), pp. 179-196.

(Shum & Hammond, 1994) Shum, B. S. and Hammond, N., "Argumentation-Based Design Rationale: What Use at What Cost?", International Journal of Human-Computer Studies, April 1994, vol 40 (4), pp. 603-52.

(Schwartz, 1998) Schwartz, D. G., “Towards the use of user-centric meta-knowledge in applying organizational memory to email communications”, in: Proceedings of the 1st Workshop OM-98, Brighton, UK (1998), August 24-25, paper 7.

(Sommerland et al., 1992) Sommerland, E. et al. “A guide to local evaluation”, Evaluation Development and Review Unit, Tavistock Institute of Human Relations, London, 1992.

(Solver, 2005) http://www.solver.com/
consulted in January 2005

(Sprague & Carlson, 1982) Sprague, R. H. and E. D. Carlson, “Building Effective Decision Support Systems Englewood Clifts”, N.J.: Prentice-Hall, Inc.: 1982.

(Szerdy & McCall, 1997) Szerdy, J. and McCall, M., “Distributed Meeting Facilitation Using EMS Tools”, ch 7C In: D. Coleman. ed., Groupware: Collaborative Strategies for Corporate LANs and Intranets, 1997, Upper Saddle River NJ: Prentice Hall.

(Teradata, 2005) http://www.teradata.com
consulted in January 2005

(Tiwana & Ramesh, 2001) Tiwana, A. and Ramesh, B., “A design knowlegde management system to support collaborative information product evolution”, Decision Support System, 31(2001), pp. 241-262.

(Universal Usability, 2005) http://universalusability.org
consulted in June 2005

(UsabilityNet, 2005) http://www.usabilitynet.org/tools/methods.htm
consulted in June 2005

(Valle et al., 2002) Valle, C., Prinz, W., and Borges, M.R.S., "Generation of Group Storytelling in Post-decision implementation process", in: Proceedings of the 7th International Conference on CSCW in Design (CSCW-D'2002), September 25-27, Rio de Janeiro, Brazil (2002), pp. 361– 367.

(Valle et al., 2003) Valle, C., Raybourn, E.M., Prinz, W., Borges, M.R.S.: "Group Storytelling to Support Tacit Knowledge Externalization", 10th International Conference on Human - Computer Interaction, Crete, Greece, June, 2003, Universal Access in Human Computer Interaction - Inclusive Design in the Information Society. Mahwah, New Jersey, USA: Lawrence Erlbaum Associates, Publishers, 2003. v. 4, pp. 1218-1222

(Valle & Prinz, 2004) Valle, C., and Prinz, W. "Decision Follow-up Support Mechanism Based on Asynchronous Communication", in the Proceedings of the 6th International Conference on Enterprise Information Systems (ICEIS'2004), April 14-17, Porto, Portugal (2004), vol. 2, pp. 423-429.

(Verity, 2005) http://www.verity.com
consulted in January 2005

(Webster, 2005) http://www.m-w.com
consulted in February 2005

(Weigand et al., 2003) Weigand, H., Schoop, M., Moor, A. D., and Dignum F., "B2B Negotiation Support: the Need for a Communication Perspective", Group Decision and Negotiation, vol. 12 (1), 2003, pp. 3-29.

(Whittaker & Sidner, 1996) Whittaker, S. and Sidner, C., "Email overload. Exploring personal information management of email", in: Proceedings of the Conference on Computer Supported Cooperative Work – CSCW'1996, November 16-20, Boston, MA, USA (1996), ACM Press, New York, pp. 276-283.

(Wiberg, 2001) Wiberg, M., "Knowledge Management in mobile CSCW: Evaluation results of a mobile physical/virtaul meeting support system", in: Proceedings of 34th Annual Hawaii International Conference on System Sciences (HICSS-34), January 3-6, Maui, Hawaii, USA (2001), vol. 4, pp. 4027.

(Williams, 2001) Williams, E.B., Georgetown Law Library,
consulted in June 2005
http://www.ll.georgetown.edu/tutorials/definitions/decision.html

(Williamson et al., 1982)	Williamson, J.B., Dalphin, J.R., and Gray, P.S., "Research Craft: An Introduction to Social Research Methods", Scott Foresman & Co, 2nd edition September 1, 1982.
(Wilson & Sifer, 1988)	Wilson, D.N. and Sifer, M.J., "Structured Planning Project Views", Software Engineering Journal, 1988, vol. 3, no.4, pp-134-140.
(Winograd & Flores, 1986)	Winograd, T., and Flores, F. "Understanding Computers and Cognition – a New Foundation Design", Addison-Wesley, 1986.
(Wired News, 2002)	Study Refutes Email Myth – Wired News, http://www.wired.com/news/technology/0,1282,56781,00.html consulted in February 2002
(Yankelovich et al., 2004)	Yankelovich, N., Walker, W. Roberts, P., et. al., "Meeting central: making distributed meetings more effective", in: Proceedings of the Conference on Computer Supported Cooperative Work – CSCW'2004, November 06-10, Chicago, Illinois, USA (2004), ACM Press, New York, pp. 419 - 428
(Yoursurfice, 2005)	http://www.yoursurfice.com/Content/XMLRPCCOM/XMLRPCCOM.htm consulted in June 2005.

9 Annex I: Lessons Learned on EMS

These lessons learned were extracted from previous work of Prof. Nunamaker and his team. For more details on them, check (Nunamaker et al., 1995) and (Nunamaker et al., 1996)

Regarding **Bottom Line Benefits**, the key lessons learned were:

- Savings from shorter project elapsed times of 90%.
- Savings from reducing labor costs by 50-70%.
- Savings from improved decision quality.
- Savings from improved buy-in to the decision.

Regarding **Group Leadership**, the key lessons learned were:

- Technology does not replace leadership.
- Technology can support any leadership style.
- Some resist GSS because it requires different leadership skills.
- There is a need to develop group incentives.
- Be willing to accept criticism of you and the organization.
- Make sure there is an individual incentive to contribute to the group effort.

Regarding **GSS application software**, the key lessons learned were:

- Interface choices affect group dynamics.
- Separate special purpose modules permit flexible process design.
- Group support must integrate with individual desktop application.
- Simplify the interface. No more than 30 seconds of instructions. It can never be too easy to use.
- The user interface must flow seamlessly from tool to tool.
- Data must move from module to module seamlessly.
- Groupware has many more ways to go wrong, and a higher failure cost than individual software.
- Users must have ready access to external data and past session transcripts.

Regarding **Outstanding Participation,** the key lessons learned were:

- Anonymity will increase the amount of key comments contributed.
- Parallel nature of the interaction increases participation.
- Participants sense they are part of the plan, that they are moving towards consensus and resolution.
- Good ideas are a function of the quantity of ideas generated. Adding participants to an GSS meeting almost always improves the outcomes.
- When participants anonymously criticize ideas, performance improves. It keeps the group searching for better answers.
- Any idea may inspire a completely new idea that would not have otherwise occurred. Develop activities which encourage frequent generation of new ideas.
- Provide feedback to groups to let them know how each activity they undertake maps to the entire agenda.
- Groups stay better focused if they understand how what they are doing ties to the big picture.

- In face-to-face groups peer pressure keeps people moving. Distributed groups tend to lose momentum.

Regarding **Electronic Voting,** the key lessons learned were:
- Voting clarifies communication, focuses discussion reveals patterns of consensus and stimulates thinking.
- Anonymous polling can surface issues that remain buried during direct conversation.
- Voting can demonstrate areas of agreement allowing the group to close off discussion in those areas and focus only on areas of disagreement.
- Electronic polling can facilitate decisions that are too painful to face using traditional methods.
- Care must be taken to ensure that voting criteria are clearly established and defined.

Regarding **Electronic Meeting Facilities,** the key lessons learned were:
- Lighting issues are extremely important and a common source of error in room design.
- Public screen is important for focusing group attention.
- Sufficient desktop space allows for spreading papers and affords visual privacy for participant work screens.
- Provide space for social interactions (eating, chatting).
- Map seating configuration to expected group activities.
- Minimize background noise.
- Provide facilitator with easy access to all room systems.
- Select room appointments to match with the type of group expected to use the room.
- Be sure people can see one another in the room.
- Partially or fully recess monitors into tables if necessary.

Regarding **Facilitators and Session Leaders,** the key lessons learned were:
- Pre-planning is critical.
- Find a fast, clean way to do idea organization – people hate it and you will lose them if you take too long.
- The group must always see where they are headed and how each activity advances them towards the goal.
- Be cognizant of non-verbal interactions. Even small non-verbal cues can tell a facilitator a lot.
- Expect that the plan and the agenda will change.
- Group dynamics can be affected by the selection of switches in the GSS tools.

10 Annex II: Questions used in the inquirying activity

Date:	
Name:	
Company:	
Work Activities:	
Project Leader ☐ **Project names:**	**Manager** ☐ **Developer** ☐

Question 1: Do you document your meetings?
1.1. If yes, how? (which information is stored, how, who does the documentation)

1.2 If no, why not?

Question 2: Do you represent the decisions made in your documentation?
2.1. If yes, how?

2.2. If not, why not?

Question 3: How do you keep track of decisions made in meetings?
( ) Informal communication
( ) Emails with people involved
( ) New meetings
( ) Checking the results of the expected activities, if sucessful or not
In this case, how do you check the results
( ) I do not keep track at all

Question 4: Do you have problems to track decisions made?

Question 5: Do you have problems to manage decisions to be made? (reminders, list of decisions, who is involved, deadlines to be made, etc).
5.1. If yes, how do you deal with that, or what do you miss, or what are the consequences for not managing the decisions (late decisions, waste of time, waste of resources, recurrence of old decisions, etc)

Question 6: Would you use a semi-structured form/mechanism to document meetings and the decisions made?

Question 7: Would you like to receive automatic follow-up about decisions made, like "success", "not successful", "problems with context and involved people", etc?

Question 8: Would you put some extra effort in meeting documentation to have more feedback associated to it?

Question 9: Now, imagine there was a tool to support meeting protocols and decision follow-up. Ideally, how do you think this support tool should work?

You can mark more than one option.
( ) Meeting participants should have this tool available **during** the meetings
( ) Meeting participants should have this tool available **after** the meetings

( ) The tool should be available in electronic and paper versions
( ) The tool should be able to work offline in case there is no Internet connection
( ) The electronic version should be available in mobile devices (and PC's)
( ) The paper version should be scanned automatically and introduced in the system
( ) The voice conversations should be analysed by a system and only the "most important" information to the related people should be sent automatically
( ) The videoconferencing should be analysed by a system and only the "most important" information to the related people should be sent automatically

Question 10: How much flexibility on this tool and mechanisms would be necessary not to disturb your way of expressing yourself during or after the meetings? (Inclusion/deletion of new fields, open text, few choices, less hierarchies (topic-sub-topic), integrated with my normal tools – email, excel, etc)

10.1 During the meeting:

10.2 After the meetings:

11 Annex III: Form Model

Date:			**Location:**		
From:			**To:**		
Name of Noter:					
Objectives of the meeting:					
sentence 1					
sentence 2					
…					
sentence N					
Participants:					
name 1					
name 2					
…					
name N					
Topic 1	**To-do**	**Who**	**When**	**Type**	**Follow-up (y/n)**
sub-topic 1	**To-do**	**Who**	**When**	**Type**	
sub-topic 2	**To-do**	**Who**	**When**	**Type**	
…	…	…	…		
sub-topic N	**To-do**	**Who**	**When**	**Type**	
Topic 2	**To-do**	**Who**	**When**	**Type**	**Follow-up (y/n)**
sub-topic 1	**To-do**	**Who**	**When**	**Type**	
sub-topic 2	**To-do**	**Who**	**When**	**Type**	
…	…	…	…		
sub-topic N	**To-do**	**Who**	**When**	**Type**	
To-do: (general)		**Who:**	**When:**	**Type:**	**Follow-up**

12 Annex IV: Scenario of use in an EU project

Nick is leading a **project that requires collaboration of various partners**, who work in different organizations and cities.

Nick **creates the project folder for coordination** of the MDSS (Mobile Decision Support System) and **invites project members** and **coordinators** of each organization.

At the beginning of the project Nick **uses the issue, decision and task folders** to start organizing the items to be discussed in the next phases of the project (e.g. Papers to submit to conferences).

The first activity a group of project members needs to do together is to elaborate the User Requirements Deliverable. In order to do that, a **meeting is organized** so that this group can discuss and define the activities that need to be done.

After the meeting project members **can negotiate and update their tasks**, **send emails, etc,** while the project coordinator follows how the activities are evolving, through the use of **reports** available in the system.

13 Annex V: Questions used in the Expert Walkthrough

❖ **Concerning the concept of Coordination Folder**
- How do you like the use of the two different roles in the definition of the project: project members and coordinators?
- Which features are missing in the coordination folder context?

❖ **Concerning the concepts of Issues, Decisions and Tasks' Folders**
- How do you like the use of the new folders of issues, decisions and tasks‘ folders?
- Do you see any benefit in the cascade between „issue – decision – task“?
- Which features are missing in the issues, decisions, and tasks‘ folders?

❖ **Concerning the concept of Project Task and Personal Task Folders**
- How do you like the relation between **project task folders** and **personal task folders**?
- Which **features** are **missing** in the **personal task folder**?

❖ **Concerning the concept of Meeting Folder**
- How do you like the use of a meeting folder to manage information about meetings?
- Which features are missing in this context?

❖ **Concerning the concept of Agenda**
- How do you like the agenda structure and management (creation, addition of items, possibility of review by other coordinators)?
- Do you see any benefit of having a centralized repository to manage agendas of distributed groups?
- Do you see any benefit of having the possibility to reuse information which is alredy in the repository (other minutes, issues, decisions, tasks)?
- Which features are missing in the agenda?

❖ **Concerning the concept of Meeting Minutes**
- How do you like the minutes structure and management (creation, addition of items, etc)?
- How do you like the explicit representation of decisions, associated to tasks in the minutes?
- How do you like the automatic update of personal task folders after the minutes are published?
- Do you see any benefit of having a centralized repository to manage minutes of distributed groups?
- Do you see any benefit of having the possibility to reuse information which is alredy in the repository (other minutes, issues, decisions, tasks)?
- Which features are missing?

❖ **Concerning the concept of Task Negotiation**

- How do you like the possibilities for task negotiation?
- Do the coloured items help to identify the tasks‘ status?
- How do you see the process of notifying the task requester about tasks that need to be managed?
- Which features are missing?

❖ **Concerning the concept of Message Integration**
- How do you like the integration of Outlook and the Coordination Folder?
- How do you like the selection of data in the Email form?
- How do you like the consequences of sending a message with context to BSCW?
- Which features are missing in this integration or in BSCW?

❖ **Concerning the concept of Reports**
- How do you like the assistance that reports can provide to coordination?
- Do you see any advantage in using text reports?
- Do you see any advantage in using graphic reports?
- Which features are missing in the decision reports?
- Which other reports do you think are necessary?

❖ **Concerning the concept of Integration**
- Do you consider the integration of the concepts (e.g. Meetings, agendas, minutes, issues, decisions, tasks, messages, roles) discussed useful for project coordination? If yes, how useful?
- Do you consider that problems related to decisions and tasks would be easier to recognize in a distributed project using such integrated approach?
- Do you consider that the communication between meeting participants and decision implementers could benefit from the integrated approach?
- Do you consider that emails and tasks are adequate means to provide feedback, and documentation about decisions under implementation?
- Which features are missing?

Lebenslauf / Resume

Carla Valle was born on 06.10.1972 in Petrópolis - Brazil

Education and Scholarships
12/1989 – High-school - Centro Educacional de Niterói, Brazil
01/1994 – Scholarship for Scientific Initiation – CNPq, Brazil
07/1994 – Bachelor in Computer Science – Catholic University of Petrópolis, Brazil
08/1994 – Fellowship as Research Assistant – CNPq, Brazil
03/1995 – Scholarship for Master in Science – CNPq, Brazil
05/1997 – Master in Computer Science – Federal University of Rio de Janeiro, Brazil
05/2006 – Doctorate in Computer Science – RWTH, Germany

Professional Experience
10/1992 – 12/1993: Research Assistant – Catholic University of Petrópolis, Brazil
03/1995 – 12/1995: High-school teacher – Instituto Canarinhos de Petrópolis, Brazil
03/1997 – 07/1999: Assistant Professor - Catholic University of Petrópolis, Brazil
10/1997 – 02/1998: Consultant – Hypermedia Technologies, Brazil
03/1998 – 12/1999: Project Coordinator – Rio Sul Regional Airlines, Brazil
01/2000 – 04/2000: Project Coordinator – Globo.com, Brazil
05/2000 – 06/2001: Project Coordinator – Federal University of Rio de Janeiro, Brazil
07/2001 – 09/2004: Researcher – Fraunhofer Institut for Applied Technology – FIT, Germany
10/2004 – 08/2000: Research Assistant – Rheinisch-Westfälischen Technischen Hochschule Aachen (RWTH), Germany
Since 09/2005: System Analyst – United Nations Headquarters, United States of America