2005

Supporting awareness in heterogeneous collaboration environments

Vijayanand Bharadwaj

West Virginia University

Supporting Awareness in Heterogeneous Collaboration Environments

Vijayanand Bharadwaj

Dissertation submitted to the
College of Engineering & Mineral Resources
at West Virginia University
in partial fulfillment of the requirements
for the degree of

Doctor of Philosophy
in
Computer and Information Science

Y.V. Ramana Reddy, Ph.D., Chair
Srinivas Kankanahalli, Ph.D.
Sumitra Reddy, Ph.D.
V. Jagannathan, Ph.D.
Ranjan Sen, Ph.D.

Lane Department of Computer Science and Electrical Engineering

Morgantown, West Virginia
2005

Keywords: CSCW, Awareness, Awareness Model, Heterogeneity, Groupware, Interoperability

ABSTRACT

Supporting Awareness in Heterogeneous Collaboration Environments

Vijayanand Bharadwaj

Rapid technological advancements have made it possible for humans to collaborate as never before. However demands of group work necessitate distributed collaboration in very heterogeneous environments. Heterogeneity as in various applications, platforms, hardware and communication infrastructure. User mobility, lack of availability and cost often make imposing a common collaboration environment infeasible. Awareness is essential for successful collaboration. Awareness is a key design criterion in groupware but often collaboration occurs with applications not designed to support useful awareness. This dissertation deals with the issue of effective group awareness support in heterogeneous environments.

Awareness propagation is effective if the appropriate amount of information, relevant to the user's sphere of activity is delivered in a timely, unobtrusive fashion. Thus issues such as information overload, and distraction have to be addressed. Furthermore ability to establish the appropriate balance between awareness and privacy is essential. Enhanced forms of awareness such as *intersubjectivty* and *historical awareness* are often invaluable. Heterogeneous environments significantly impact the above *quality factors* impeding effective awareness propagation. Users are unable to tailor the quality of awareness received.

Heterogeneity issues that affect awareness quality are identified. An *awareness framework* is proposed that binds various sources of awareness information. However for effective awareness support, physical integration must be augmented by information integration. As a solution, an *awareness model* is proposed. Specification of the awareness model and framework's architecture and features is the key contribution. The proposed model has been validated through simulations of realistic collaboration involving human participation. Scenarios created, have tested the model's usefulness in enhancing the quality of group work by propagating effective awareness among users. To accomplish the same, an *Awareness Simulator* application has been created. In the validation process, efforts made to create an experimental methodology revealed some techniques related to awareness evaluation in CSCW, which are proposed. Various issues required to successfully engineer such awareness frameworks are identified and their impact on requirements such as security and performance, discussed. With various standards and technologies that can be harnessed to create awareness frameworks, there is great promise that barriers in heterogeneous collaboration environments can be overcome.

Dedication

To my mother
for
her unrelenting tapas (penance), sacrifices
and for
<u>never</u> even once ceasing to believe in me all
along the way!

Acknowledgements

Words fail me as I seek to express my gratitude towards my advisor and committee chairperson Dr. Y.V. Ramana Reddy. I am overwhelmed by the tremendous amount of encouragement, support and guidance that he has showered on me throughout my studies at the Concurrent Engineering Research Center (CERC) at West Virginia University. His abundant enthusiasm and energy for ideas, creativity and his strong support for research and the people who pursue it never ceases to amaze me. I am greatly inspired by him, not only in his capacity as a teacher and researcher but also a leader. Leading by example, he has nurtured in all his students, including myself, an attitude of genuine willingness to learn from anyone who has knowledge, and an open positive attitude welcoming everyone's point of view. He has always instilled in us confidence in our abilities tempered by the grace to acknowledge that there is so much more to learn. This disposition I term as the "CERC culture" that Dr. Reddy has fostered in all the CERCians over the years. Among the many gems I take away with me from my experiences at CERC and West Virginia University, I consider this "CERC culture" to be among the most valuable. I am greatly indebted to Dr. Reddy for all the effort he has taken to be my mentor, the tremendous positive influence he has had on me and for equipping me to confidently embrace life. But for the faith and confidence he has had in my abilities, I would never have accomplished what I had set out to do and for that I am forever grateful to him.

I would like to express my deep sense of gratitude to Dr. Srinivas Kankanahalli. "Dr. Srini", as he is fondly known to all his students and colleagues has been greatly instrumental in the successes that I have reaped in my studies at West Virginia University. His invaluable support not only enabled me take up graduate studies but also helped me throughout the duration. He has been a very positive influence both in my professional and personal life. His genuine, friendly helpful attitude has always inspired me and I have been the beneficiary of that so many times. As a teacher and researcher Dr. Srini has always instilled in me the drive to question, and never be satisfied with the trivial. He has encouraged me to go after the hard problems with confidence when most would be dissuaded. He has demonstrated time and again the same drive in tackling issues both in the discipline as well as in endeavors beyond, all from which I have and continue to learn from. I would like to thank him for taking so much effort on my behalf from the beginning and all the way during this long journey I had embarked on.

I would like to express my heartfelt sincere thanks to Dr. Sumitra Reddy for all that she has done for me during my studies at CERC, West Virginia University. Her affectionate helpful nature and sunny disposition, towards all the students at CERC, WVU has me considering her a mother figure in the lives of all the CERC students including myself. Be it suggestions on how to improve my research work, or my presentation skills, or telling me to drive home safely in the snow, or requesting me to keep warm and not fall ill during the harsh West Virginia winters, she has always been genuinely concerned and looked out for me. Along with Dr. Ramana Reddy she has been

the co-founder and architect of the "CERC culture" and practices it steadfastly while nurturing it in all of us students. I am so thankful to her for all her efforts in greatly enhancing my work at CERC. I am fortunate to have her play such a crucial role in my development not only as a student, a researcher but also a citizen of society.

I am so grateful to Dr. V. Jagganathan for being my teacher over all these years. I have always admired his keen intellect, scholarship and mastery of his discipline as well as his dedication to teaching, that Dr. Jagganathan, or "Dr. Juggy" – as he is popularly known to all, exhibits time and again. His constant enthusiasm to learn is evident from his creation of new and unique courses where he has demystified cutting edge technology with his grasp of the fundamental principles of the discipline. I am extremely thankful to him for all the effort he has taken to be part of my dissertation committee and his invaluable advice and teaching from which I have greatly benefited.

I must express my sincere thanks to Dr. Ranjan Sen, Microsoft Corporation. I am so grateful to him for readily willing to be part of my committee. I am extremely honored for the tremendous amount of interest and concern he has shown towards my work as well as very thankful for the invaluable suggestions he has given to improve my research. In spite of so many constraints he has been so accessible and helpful to me in my research work, and in being part of my committee, and I am ever grateful to him for the same.

I would like to thank my colleagues from the Awareness Project of which this dissertation is a part. Madhukar Kakaraparthi, Adarsh Nara, and Satyadev Adiraju all graduate students have been my research teammates in this project. I am extremely thankful to them for all the unending enthusiasm, tremendous hard work and excellent team spirit they exhibited. They helped me evaluate the theoretical concepts in this dissertation. Their invaluable effort was one the key factors to the successful completion of my dissertation.

I would like to specially mention Mr. Ravi Raman (former Assistant Director at CERC). Mr. Ravi has been an excellent teacher and guide to me over all the years and an ardent supporter of my work. I have been a great fan of his abilities to understand the challenges involved in large research efforts, be they technical, administrative, or what is sometimes the most crucial, the social challenges. The acumen he displays in tackling those challenges and the thought processes he employs in doing so are a lesson in itself for anyone, be they a novice graduate student or professional involved in the complex, intellectually challenging realm of research and development. His deep sense of awareness of various seemingly unrelated aspects of a project and their intricate dependencies has always astounded me, and to a large extend been the inspiration for my research on awareness. His ability to bring out the best in people working with him to meet the challenges, while guiding and mentoring them, and encouraging them, towards their individual successes is a treat to watch. I am ever grateful to him for spending so much time and effort with me during all these years at CERC, and teaching me how one approaches problems, understands their complexity and nuances, and utilizes ones

abilities and acquires those not present to tackle the challenges. This training has been one of the most crucial factors in my accomplishing this research effort.

I am very thankful Dr. Floyd Russell (former Director, Virtual Medical Campus, WVU) for the tremendous genuine support and encouragement he has provided for my research work, as well as helping me with the financial support without which I could not have pursued my dissertation at all.

I extremely thankful to Mr. Alan Butcher (Associate Director, Virtual Medical Campus and Assistant Director, CERC) for all the help he has rendered me over the years. His fantastic disposition and helpful nature helped me navigate various difficulties, be it technical or otherwise during my years at CERC. I am amazed by his organizational capabilities as well as his technical savvy and foresight and I think of him as my teacher as I continue to learn from him so many essential lessons that one needs to succeed in the discipline.

I cannot thank enough, William Hunt (Research Associate, VMC and CERC) for everything he has done for me. Bill has been en excellent teacher and friend to me.
Be it looking over the code I write and gently informing me how I should do better, or trouble shooting my computer system numerous times, Bill has been very instrumental in smoothening out the various bumps along the road for me all through my years at CERC.
I have learnt many a valuable lesson from his technical and analytical problem-solving prowess and eye for detail as well as his quick and strong grasp of technology. His ever-friendly and helpful attitude made working with him at CERC a most enjoyable experience.

I am extremely grateful to Dr. George Trapp, Dr. John Atkins and Dr Jim Mooney. All three of them have been such wonderful teachers, and have strongly encouraged all my efforts over the years at LDCSEE. They helped me take my first steps in the discipline and the foundations they laid in teaching me has been the reason I have had the courage and confidence to even begin this endeavor and build this research effort upon. I am very inspired by their erudition, superb command of the discipline, phenomenal teaching ability as well as the strength of character, integrity and fairness that they exhibit both professionally and personally, and I think of them as the pillars on which LDCSEE rests. I consider myself extremely fortunate to have them be part of my life and deeply influence me in such a positive manner.

Special thanks are due to Dr. Bojan Cukic. He has been extremely supportive of me all through the years. He was instrumental in teaching me the basics of the research process and providing me the support, and encouragement to embark on my dissertation. I am forever thankful to him for taking me under his wing as his research student and giving me the opportunity to begin my journey into the research world.

I would like to thank Dr. Brian Woerner (Chairman, LDCSSE) for all the support, encouragement and help he has provided me without which none of this would be possible at all.

Table of Contents

List of Tables

List of Figures

Chapter 1
Introduction

The chapter introduces this dissertation in terms of the research problem it is addressing, the research outcomes, the methodology followed and the contributions of the research effort. The context for the research problem is set in the first section. The research problem itself is discussed next. Subsequent sections list the outcomes, followed by a brief overview of the steps taken to accomplish the outcomes. This chapter concludes with the contributions that this dissertation makes and an overview of the rest o chapters in this work.

1.1 Technology and Group Work

Rapid technological advancements have made it possible for humans to collaborate as never before. Unprecedented computing speed, large storage capacities, high-speed networking, and sophisticated graphical user-interfaces are prime examples of this phenomenal progress. The popularity of the Internet and WWW has resulted in their being used with same degree of familiarity and comfort as common household appliances in many parts of the world. Enabling technologies have turned these once esoteric entities into massive substrates on which a variety of collaborative applications can be hosted. Newer paradigms and innovations such as Ubiquitous and Pervasive computing, Adaptive and Self-healing systems hold great promise for productive group work and the possibilities of exploiting them seems to be only limited by our imagination.

However there are some fundamental issues that strongly influence the productivity of collaborative work. Technological progress has resulted in the proliferation of a variety of applications for group work. We find ourselves working with a wide variety of applications and groupware that run on different platforms, trying to exchange data (with different syntax and semantics), and communicating over networks of different types and capabilities. In other words we are collaborating today in **highly heterogeneous environments**. Problems such as the lack of interoperability among these applications inhibit effective group work. The varied nature of group work and the **high degree of flexibility** observed in teams does not make this any easier. Interesting to note the observation by Eugene Kim [Kim 2004] that in spite of sophisticated groupware applications, often we find that one cannot achieve the same degree of flexibility as one can with a piece of paper, where one can pass it around in a meeting, have everyone read it and annotate if necessary. The fundamental elements of collaboration such as *access to information, communication, coordination* and *computation* are all affected by the lack of interoperability among the myriad of groupware applications and systems that are commonly used. Thus the **heterogeneity of collaboration environments** is a key contributor to the problems involved in effective collaboration. Furthermore the demands placed on these tools and groupware systems to adapt to the highly flexible in nature of everyday collaboration adds to the problem. Consequently each of the above fundamental elements necessary for collaboration is greatly affected. If access to information, ability

to communicate, coordinate and compute are hampered by lack of interoperability and the high degree of flexibility then the processes involved and outcomes of collaboration suffer.

1.2 Awareness - A Key Enabler

1.2.1 The I3C Elements of Collaboration

An examination of the four basic elements reveals a common thread that is pivotal to defining the research problem.

1.2.1.1 Access to Information

Members of a collaborating group must have access to all necessary information about the group's collaboration. Information includes the artifacts that are worked on and created as an outcome of collaboration as well as the information necessary to coordinate and execute the collaborative processes. Information about people involved, places of collaboration, resources, goals, tasks, deadlines and various other aspects related to the group work must be available. Also ascertaining the current state of the group effort is essential to gauge the progress, allocate scarce resources and tasks, resolve dependencies, and ensure timeliness both in the planning stage as well as execution. Thus the state of the group effort must be visible to participants. In project management terminology this is referred to as *"Project- Visibility"* [Patterson 2004] and is essential for effective coordination and execution in group work.

Successful *access to information* requires not only knowing the above aspects but also how to actually acquire the information, including where can the information be found, how can it be accessed, by whom, when and so on.

1.2.1.2 Communication

Communication is essential to execute collaborative processes. Participants involved must be able to communicate with one another to coordinate project-related issues as well as disseminate the coordinated plans to the team members. The frequency of communication would depend on the nature of work in question. Communication can be between humans, humans and systems (software) and between systems. The processes that constitute access to information described above involve communication.

Successful *communication* involves knowing all the aspects related to the act of communication such as the subject of communication, the recipients, their status (availability), the mode of communication, and timing to name a few.

1.2.1.3 Coordination

Coordination in a team is essential to successfully achieve the goals of a joint effort in an efficient manner. Misplaced efforts, wastage, redundancy, and under utilization of resources result in sub-optimal performance of the group and can lead to delays and even failure of the effort. Planning and coordination before the group's activities commence can help in uncovering potential problems. Even during the course

of a group's work, coordination is essential. To successfully complete tasks teams need to coordinate for various reasons, such as dealing with exceptions, allocation of scare resources, and ascertaining the status of ongoing activities to name a few.

However successful *coordination* itself requires knowledge of what needs to be coordinated, among whom, the constraints involved and so on.

1.2.1.4 Computation

The actual activity (apart from communication) performed by individuals and systems to accomplish work processes we term as computation. Computation does not necessarily mean an activity related to mathematical computing though it could involve those also. Any activity performed with the aid of a computing system (application, groupware) is considered as computation. Some examples are creation and editing of documents and running programs that processes some business logic.

For the computation processes to be executed successfully, one has to have knowledge of what to compute, when and how and the expected outcomes and other details.

Figure 1.1 below illustrates this concept of the fundamental elements of technology-supported collaboration, which we can term as I3C short for Information, Communication, Coordination and Computation.
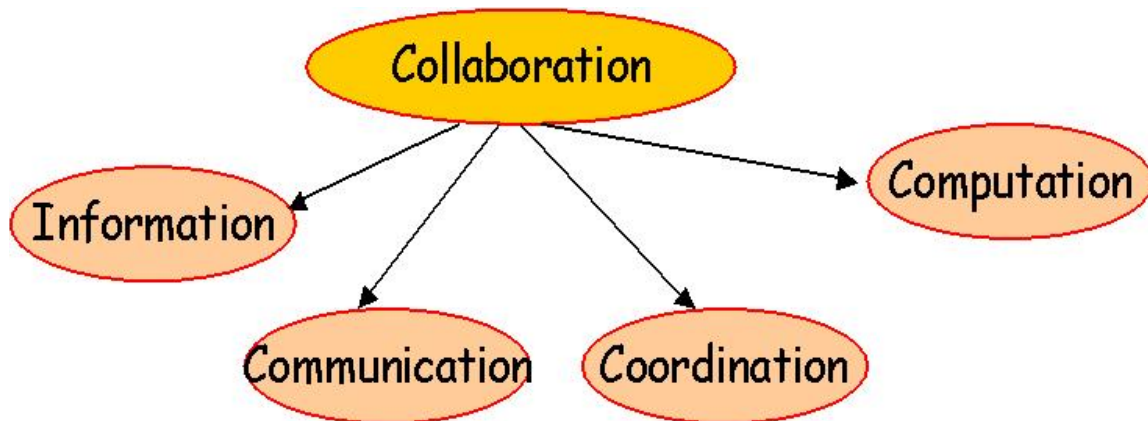
Figure 1.1 I3C Elements of Collaboration

1.2.1.5 Activities in the Absence of Information Technology Support

In collaboration there are various activities that are performed without the use of computing systems or for that matter any information technology support. Even in this day and age we find that the above four elements are often accomplished paper-based. Meetings are still held face-to-face and the support is often in the form of handwritten minutes from previous meetings and a simple whiteboard in the meeting room. People still use yellow sticky-notes as opposed to sophisticated software personal agents. In fact all the above, *access to information*, *communication*, *computation* and to a very large extent *coordination* occur using no technology support at all or a mixture of some

technology. For instance someone may receive data via email, manually compute using the data as input and then email the results back. We find such mixed mode of execution to be more prevalent than purely technology based means. One could also claim that there are activities that are by very nature cannot be supported by technology such as human thought and analysis.

In any case for the successful accomplishment of collaborative processes the humans and systems must have the relevant information at the appropriate time in the process. Especially with some activities supported by technology and others done manually, we often find the need that lack of knowledge about details can hamper the overall execution. For instance we try to import a set of data created manually by somebody into a spreadsheet program and the results of the import action are not as expected since the data has not been formatted as required.

The common thread of thought from the above elements is that there needs to be knowledge of all relevant aspects related to that element to ensure its successful initiation, execution and completion. In other words one needs *to have an awareness of* all those relevant aspects. The challenge from a computer and information sciences perspective is how best to design and develop systems that support awareness in collaboration.

1.2.2 Awareness - A Definition

Awareness in group work has been a topic of significant interest to the CSCW community. Different types of awareness have been named in literature (details are provided in Chapter 2). For instance a very popular definition for *group awareness* in CSCW is one by Dourish and Belloti [Dourish and Belloti 1992], which is**,** *"An understanding of the activities of others which provides a context for your own activity"*. However activities are only one aspect of awareness. Various other elements of information make up our state of awareness. This includes all the sensory input we continuously receive as well as the information we already know from prior learning and experience, but is currently in our mental state. It is in on sense our "working memory". Awareness is the sum total of all the sensory information and the contents working memory. This sum total of information is dynamic and is mutating as new sensory inputs arrive and more stored information is brought to the foreground. This state provides a current context and influences our actions. Also we have the ability to choose as only some of that sensory information is used at the present time, some is stored away and some is discarded. Thus by its very nature, awareness is difficult to define and characterize. Thus the challenge for CSCW systems is how best to capture and propagate awareness.

1.2.3 Significance of Awareness: Problems and a Silver Lining

As is a common experience our knowledge of all the aspects of a certain situation help us greatly in dealing with the problems associated with it. Often situations that we have dealt poorly with can be attributed to our ignorance of some aspects that lead us to take certain decisions and as consequence perform certain actions. Furthermore in group work as actions taken by someone can directly affect the state of awareness of others and in turn their actions, it is essential that one has the appropriate awareness of all relevant aspects. In other words, awareness, especially in collaboration involves knowing the *"Who", "What", "When", "Where", "How" and "Why"* about some aspect of collaboration.

For collaboration to be successful the fundamental elements of collaboration must be enabled and awareness is an essential enabler for that to occur. Better awareness empowers us to take better decisions, perform appropriate actions, be proactive, anticipate and prevent problems and remedy necessary situations. The significance of awareness in group work only reinforces the age-old adage, "*Knowledge is power* ".

However the nature flexible nature of group work means that the state of various elements is constantly changing. To be aware means one must be cognizant of these changes and our knowledge of the state must be relevant at any given time. Added to this is the technical barrier in terms of lack of interoperability in maintaining appropriate awareness. Thus the quest of this research effort is to examine the impact of the above problems on awareness propagation. It is interesting to note that there is a silver lining in the problems itself. Though the heterogeneity of collaboration environments and dynamic nature of group work pose problems to awareness propagation, even little information that is propagated about the barriers themselves can help members. With awareness about the lack of interoperability or the fact that changes are occurring, a group can organize their work and the environment in such a manner that work can be accomplished in spite of the barriers. Thus awareness in whatever limited form can be beneficial in creating workarounds to the impediments in group work and discover alternative means to accomplish the work processes.

The rest of this chapter provides an overview of this dissertation proposal by discussing the research problem of interest, the steps taken in investigating the problem (research methodology), the expected outcomes and the contributions of this effort.

1.3 Research Problem

Having examined the nature of awareness and the significance of awareness in group work as a key enabler, we will define the research problem. Our awareness includes information about various aspects. Many of these aspects may be completely unrelated to our sphere of activity with respect to our collaboration with others for example personal messages that we receive from home. On the other hand a personal message may be of such high priority that we may have to reschedule our collaborative activities so the we can attend to the personal matter. Thus awareness of other seemingly unrelated aspects can play a very important part in the overall outcome of our collaboration with others. If the awareness that is propagated to a user is of the relevant type, arrives in a timely fashion then it can have a positive impact on the actions that the user takes with respect to collaboration. On the other hand the lack of awareness, or awareness of the information that is incorrect, or information that is not timely can all result in inaction or incorrect actions or actions that do not produce the desired end results. Furthermore as other individuals and systems depend on a user's actions, this can have a negative ripple effect in terms of the entire group.

Thus to ensure the success of collaboration, users must be aware of all aspects of group work that they require to perform their actions successfully and this information must be available in a timely fashion. Better awareness can have a very positive effect on the overall quality of work itself, as described in the previous section. Thus a broader requirement that includes the above would be that *users must be made aware of all aspects of group work that are relevant to their sphere of activity in a timely fashion.*
The research challenge then is to be able design and develop information systems that can support propagation awareness information that enhance and improve the quality of collaboration.

1.3.1 Awareness Quality and Effective Awareness

Even if information propagation can be facilitated to every member of the group, this awareness may not necessarily constitute useful awareness. We find in today' world we are well connected by various means and receive a lot of information. In fact we can said the have the problem of having too much information. Be it from the twenty-four cable and news channels on television (with their endless tickers and breaking news items), radio stations to the ubiquitous internet, endless number of websites, personal web logs (blogs) and online communities. Subscription to these communities and forums can result in alerts and email from them. Even the act of navigating the World Wide Web can be tedious when one is bombarded with numerous pop-up windows with advertisements. Then there is a sea of email messages that flood our inbox with messages related to work, our personal life or alerts from our travel agent to our personal-trainer reprimanding us for missing yesterday's workout (and also informing us that we will still be charged). Then there is the unsolicited and unwanted (spam) that makes its way to our email accounts. Often leading to wastage of valuable time and effort in getting to the messages we expect. Thus we seem to be inundated with information all the time and often we are under a sensory overload. Very easily due to the information deluge, and the distraction it causes, our awareness of information that is really useful may be missed.

It is interesting to examine what constitutes *useful awareness* i.e. the awareness that is useful with respect to our collaboration with others. This leads us to ask, how can CSCW systems be designed to help users to receive and utilize this useful awareness? But before we can begin answering that question we have need a means to define useful awareness

Useful awareness can be thought of as *"information that has certain qualities that make it useful to the recipient"*. This research effort aims to define those quality factors or characterize the quality of awareness information. These are as follows:

Quality of Awareness: Quality of awareness can be characterized by factors such as *relevancy*, *information overload*, *obtrusiveness*, *access control and privacy*, and support for *enhanced awareness.*

1.3.1.1 Relevancy

How relevant is the awareness information to my sphere of work, both current and for the future (planning and coordination)? The type or the subject matter of the information being received and the time at which we are made aware of that information determine if the information is relevant to us or not. To accomplish certain activities awareness of very specific types of information is necessary. For example: Information that is about issues that are not related to our sphere of activity are not useful. Information that is related but arrives too late to be of any effect is also not useful. If information arrives too early, then there must be some means to ensure that it is made available again at the appropriate time. If such mechanisms are not in place then the information may not be usable. Thus the type of information i.e. what is this information about, as well as the time it is made aware to its recipient is of utmost importance in its usefulness.

1.3.1.2 Information Overload

The amount of information made available is an essential factor in determining its utility. Information that is relevant must be completely received. Incomplete information may not be usable. On the other too much information can be a hindrance too since not all of it may be necessary and the overhead of processing the inputs (accessing, reading, storing) may be detrimental to other necessary activities. Too much information can cause unnecessary distraction (discussed below). As discussed above due to our state of connectivity and the proliferation of information sources, avoidance of information overload is essential. Often users can be inundated with information in an effort to be kept aware of ongoing group activities. Managing this aspect has been an area of interest in awareness research and various techniques such as subscribing to selective events and filtering techniques have been devised to deal with this problem. On the other hand there are certain types of information that the users must be aware of due to the nature of their work. Even if the user perceives that this is too much information to deal with, there must be means of keeping him or her informed but in a manner that is not distracting and counter-productive to his/her current activities.

1.3.1.3 Obtrusiveness

This factor determines how distracting the awareness information is to the recipient's current activity? In order to be informed the recipient's attention must somehow be caught. If this process is very distracting or disturbs the recipient's current activities, it can be counter-productive. Furthermore the frequency with which someone is informed can also cause distraction. The process of becoming awareness by nature will involve some distraction, but it is of importance to keep the distraction to a minimum i.e. no more than absolutely required. If the information being received demands the recipients attention in terms of actions (access, save, process), it can interrupt the user from his current activity. The recipient of awareness information may prefer to be informed in a certain manner. Often information about the same aspect of collaborative work can be displayed as a text message, or a change in color of an icon, or the appearance or disappearance of some icon, or change in size. A user who is busy in a meeting may wish to receive messages on his cell phone but not disturb others. Thus we see that cell-phone, which are equipped to vibrate as opposed to playing a loud ring-tone that can disturb others. An active area of Human-Computer Interaction research is how best to create mechanisms (displays, voice) to inform users with minimum distraction. Furthermore users with physical disabilities have special needs and the need to be informed in ways that aid them.

Apart from the above factors two other aspects of awareness described below help the processes involved in collaboration greatly.

1.3.1.4 Access Control and Privacy

Can privacy in the form of access control be established to ensure that only one with appropriate permissions is aware of classified information? Privacy is directly related to awareness and this can have a tremendous impact on security. A project may have policies on who needs to be aware of certain types of information.

Especially in group work if more information about a user's activities, location and other details are available, it comes at the cost of the particular user's privacy. Often only a subset of individuals are on a "need to know" basis and the awareness mechanisms should allow such selective access to information. Though awareness and privacy are orthogonal, their impact on security of a group project (system, facility, information etc) can depend on how the awareness is used. In some cases depending on the types and sensitivity of information involved this awareness can lead to a compromise in the security of the system. Malicious use of increased awareness can compromise security. On the other hand increased awareness (at the cost of privacy) can enhance security and even safety as seen in video surveillance systems, motion sensors guarding perimeters of facilities, child monitors, and intrusion detection systems programs to name a few. In a nutshell the issue to be addressed is if we can establish who receives information and who does not.

1.3.1.5 Enhanced Awareness

Apart from the information content itself there is meta-awareness i.e. awareness about the state of awareness of members in a group (including oneself). Some examples are, *"Who else is aware of what I am aware of?"*, *"What is so and so aware of ?"*, *"Who*

else is aware of my activities?" and *"I know that he knows that I know"* (intersubjectivity). Such meta-awareness is often essential in organizing access to information and coordination in a group. Furthermore such awareness helps in avoiding the sheer overheard of ensuring everyone has the same information.

The quality of awareness determines how useful it is to a recipient and therefore impacts his or her role in group work. As each user has different roles and tasks he or she will use awareness in a different manner. Hence the information that is useful to someone may not be useful to another user. Thus useful awareness is from the perspective of the recipient and is subjective. However the above quality factors can be used as dimensions of the utility of information. We can now define ***effective awareness*** as the awareness with the desired quality (defined by the above factors) to aid in successful collaboration i.e. accomplishing the collaborative processes and where possible improving the quality of overall work. In other words effective awareness is awareness that is useful (relevant, timely, appropriate volume, non-distracting, adheres to the access control policy, and provides enhanced awareness if applicable) and this term is used interchangeably with useful awareness throughout the dissertation.

1.3.2 Awareness Information Characteristics

Awareness Information possesses certain characteristics that describe it. In fact any type of information in general can be described by these attributes:

Type (What am I aware of): Is this about an individual's activity or location, actions on an artifact, conversation transcript, streaming video of a meeting, or a combination of the above?

Form: Is it text, audio, visual?

Volume (How much of it am I aware of): Am I aware of every email exchanged by the group or just a synopsis; is it a long videoconference session? Do I receive a notification for every action taken on an artifact?

Time & Frequency (When and How often do I get that information): In a highly interactive chat session that I am monitoring, do I want to receive every line typed?

From the discussion on the quality factors it is clear that the quality of awareness information depends on the characteristics (described above) that it possess. How effective some information will be to collaborative work will depend on the above characteristics. In Chapter 3 we will explore this relationship in greater detail.

We see that it is not just awareness but quality of awareness is essential to collaboration. The information that one receives or one needs to be aware of must be relevant to ones spheres of activity, arrive at a time when it is needed or when it is of the most use, must not distract one unduly. Thus the challenge for the research now includes effective awareness as an essential aspect in awareness propagation.

1.3.3 Heterogeneous Environments, Morphing Collaboration and Awareness

The importance of effective awareness in group work has been discussed. This section discusses the other important part of the research problem, which is the heterogeneity of collaboration environments. Collaboration environments are made up of heterogeneous applications groupware systems, platforms and networks (infrastructure). But it is not just the environments that cause problems, but it is the changing nature of collaboration that is also a problem. We will examine in detail the barriers caused by these and related issues in this section. We begin by listing the various issues involved.

1. Groupware: Promises and Pitfalls for Awareness Propagation
The history of CSCW has shown that the design and development of groupware applications have made tremendous progress. We see very sophisticated groupware applications that claim to provide a complete solution in terms of fulfilling all collaboration needs. Groupware can be niche applications such as conferencing (video and audio) or user-friendly email (Novell's Groupwise [Novell Groupwise 2005], Microsoft Exchange [Microsoft Exchange 2005]), workflow applications (IBM's WebSphere suite [IBM WebSphere 2005]) or all-in-one suites such as Microsoft Collaboration suite [Microsoft Collaboration 2005] of products (LiveMeeting, Exchange, Project), which recently acquired Groove Networks [Microsoft and Groove 2005] [Groove Networks 2005], IBM's Workplace Collaboration Suite [IBM Workplace 2005], IBM's Lotus Suite [IBM Lotus 2005] and even open-source web-based systems such as eGroupware [eGroupware 2005]. Each groupware application supports awareness of aspects related to that particular groupware's primary purpose. For example, groupware for email will support awareness of not only the basic aspects of an email (such as displaying information in the headers of an email) but also awareness of priority of message, when a certain messages was accessed, notify a recipient as soon as a message arrives and so on. If all the members of a group use the same groupware then awareness propagation is not such a problem, because they are all empowered with the same features with respect to awareness. Thus groupware in their particular domain do support awareness well. However in reality this is not always the case as seen below:

1 a. Tools (applications) versus Groupware
Even though a wide variety of groupware applications are very popular and in use, we find that very often that people use applications or tools not designed for group work in while they collaborate. Many applications are created for specific purposes and they cannot be shared with others. Thus we often find ourselves creating information artifacts in one applications (a diagramming tool, a spreadsheet world-processor, a multi media production software) and then using another applications such as email or file transferring program to the intended recipients. There has been a trend to augmenting applications that are not designed for collaboration with functionality to be used by multiple users (Microsoft's Word and PowerPoint have the "Online Collaboration" functionality that allow you to set up meetings and in the case of PowerPoint the "Broadcast" the slideshow). In spite of the above trend, we very often find that users work with applications not equipped with collaborating capabilities. This could be for various reasons such that they are using an older version of the same program, which is not equipped with the collaboration functionality, or they have to install it separately and they

do not have the required packages. In any case to be truly useful everyone must have the same application with the same capabilities and this cannot be guaranteed. More specifically these applications are not designed with any intent to propagate awareness. Just having the ability to communicate and transfer information among multiple participants does not necessarily mean the application can propagate meaningful awareness information.

1 b. Mixture of Groupware
Even if groupware is used to propagate awareness, we find ourselves employing a various groupware applications together. These applications do support awareness however the awareness is specific to the groupware and associated with the groupware's primary purpose as mentioned before. Groupware differ based on the manner in which they facilitate work among groups distributed in space and time. The classical taxonomy of groupware [Ellis, Gibbs and Rein 1991] places groupware in four quadrants of the space and time axes -*same place same time, same place different time, different place same time* and *different place different time*. Even within the same quadrant, groupware applications differ based on the wide variety of capabilities they offer, the platforms they use, data formats, user–interface, networking support as well as factors such as usability and cost.

2. Heterogeneous Infrastructure
Apart from the variety of applications and groupware employed the infrastructure used to run the applications and connectivity exhibits great variety in terms of capabilities. Very different platforms are used, and network connectivity can vary in terms of bandwidth as well as some being wireless while the majority still being wireline. Even if every member of a group and entire groups working with other groups choose to use the same application, heterogeneity at the underlying infrastructure level does not permit them to have the same capabilities in terms of collaboration and consequently awareness propagation.

3. Mobile Users
The needs of work demand that users must be mobile. In spite of their mobility requirements they should be able to work together with the rest of the group members, some of who may also be mobile. Advances in wireless technology, cellular networks and handheld devices have made communication possible as well access to information servers, email and the WWW possible. However various wireless networking and cellular telephone standards that do not necessarily interoperate make it difficult propagate awareness and collaborate. Handheld devices also imply that information propagated to users must be tailored to their display capabilities.

4. Dynamic Nature of Collaboration
Any collaborative process changes over time. Changes can occur in terms of the objectives of the collaborative process, and its details, the policies of the group [Bharadwaj et al. 2004]. Changes in personnel, which are very common, locations, resource requirements and resource availability, as well as other constraints set can change. Thus the applications and environment selected for the collaborative processes may no longer be effective or useful due to the changes. Consequently these changes can

also affect the awareness propagation, since the environment in place can no longer support the changes. Awareness about these changes itself must be propagated since that very essential information to help the group prepare and adapt to the changes. As mentioned before awareness also provides the silver lining, if awareness about the imminent changes can be propagated well in advance, then transition due to the changes can be smoother for the group.

5. Unanticipated Requirements, Constraints and Situations
The previous point talks about changes that are planned for and known in advance. However there are many changes that occur unanticipated. These are often a consequence of the collaborative processes themselves. During the course of group work, we may find that certain new requirements arise, or certain constraints are discovered that were not present before. Situations come about as a result of failure of various aspects including the collaborating environments. The severity and impact of these situations can vary. Once again awareness propagation can suffer due to these unanticipated changes. If the environment does not support us incapable of dealing with these situations then awareness propagation may suffer. Once again propagating awareness in some manner about these changes as soon as they occur can help deal with the situations to some extent.

1.3.4 Impact on Effective Awareness Propagation

From the above discussion we see that the plethora of support for collaborative work is welcome but such variety or **heterogeneity** comes with a price. Seeking to collaborate on broader scales, we find that the **lack of interoperability** is one of the main hurdles to effective collaboration among groups. Apart from just the applications, non-uniformity in the infrastructure that supports the applications implies that not everyone can adopt and use the same application set. For instance it would be of tremendous value, if two workers employing separate applications could create, access and modify a document without being concerned if one application supported the other application's document formats. Great strides have been made in enabling applications to interoperate and this is an active area of research.

The focus of this dissertation however is the impact of the heterogeneity on the ability to obtain awareness about the group effort. There must be mechanisms for teams employing a mixture of groupware to obtain the necessary visibility. One can argue that even if every member of a collaborating group does not share the same tool set, awareness of the group's tasks and related aspects is essential. For instance every member of a group co-authoring a paper may not have the same word processor and imaging tools, or work on the same type of platform. However if everyone knew what was to be written in the paper and their respective parts, they could work independently. They would then send their material in plain-text format to the person in-charge of integrating all the pieces. This person could then uniformly format and illustrate the document as per the requirements. This is a work scenario observed frequently. The key here is that **everyone was aware** of what was needed to complete the effort. In fact awareness support can aid in coordination and even alleviate some of the problems caused in spite of the lack of interoperability.

Even if members of a working group share the same tool set, it would be of great value if applications could share information of actions on artifacts (modifications, deletion etc.) that were of common interest. In that way members would be aware of the latest status of the artifacts. Similarly information about the activities of other users such as there where they were located if mobile may also be of value. For instance if a group used a common web-based collaborative editor such as a Wiki, and email to communicate, then the normal course for member A would be to update the document in the Wiki and then email member B the link to it. On the other hand if member B could be notified automatically when A had updated the document then he could act immediately if necessary. B may want to know all the updates of the document made by A or by everyone in the team. Also A could subscribe to be notified when B had received notification of his update. Such capabilities are present in some sophisticated collaborative applications. The issue of interest in this research is, **how could such mechanisms be enabled if B and A are using a different set of applications.**

In a collaboration environment, every application and groupware system can be considered as source of awareness information. When users interact with these applications, information about various aspects of the interaction such as when did the user logon and logoff, what activities were performed and so on can also be part of awareness that are useful to others in the group. This information must be propagated via the environments to other users. These users in turn hare themselves interacting with other applications and groupware, running on different systems. These environments

must be able to receive awareness information generated by others and display it in a manner that is effective to the users working with these applications. In heterogeneous environments there is not guarantee that such awareness can be propagated faithfully as it was generated. This is because awareness information is comprised of not only the actual information content (such as am artifact created by an application) but also meta-information about who creates, when, how large is it, and other details which can be very useful in collaboration. The propagation of this information depends on the specific application. There are no specific requirements for the applications to transfer this information. One may argue that standards used in Internet technologies have been successful in enabling information propagation. This is true when you consider that any email server that follows the SMTP protocol can send and receive email that can be accessed by its users. However different email system vendors have created programs that use SMTP but add much more functionality to their applications such as MS Exchange, Novell's GroupWise and others. Some of these applications can generate HTML based messages with images and voice embedded, some can provide notifications when the recipient(s) access your email, some provide filtering capabilities for spam. On the other hand there are many bare-bones email systems such as pine or elm that can be invoked on a simple text-based display (such as one by the by the telnet program) which cannot display the HTML, launch audio files from the email client, or have capability to display notifications and so on. Different email systems have different attachment specifications and sizes. Some email systems that are specific to a teaching or classroom application allow you send email only to the students registered in the class. This can be a limitation if the instructor wants to email students in his research group. He has to send a separate email, to them. Here is an example of where even if the underlying protocol is the same, the applications can be vastly different. Another example is the HTTP protocol. Many web-based groupware use HTTP for all their communication. However this does not mean that all of them support the same functionality. The issue we are grappling with is how to propagate awareness among these applications, which differ vastly ay the application layer.

Thus the applications and groupware, generate awareness information of different types(related to different aspects), with different characteristics, forms, volume and frequency . The users may not only be incapable to receive the information because they may not have the connectivity. Even what they receive depends on how they are connected. Finally they may not be able to tailor the quality of information they receive. This is because they do not have control on the sources themselves. So how do users receive the awareness information of the appropriate quality? How can they ensure that the awareness they receive is effective or useful since the environment is made up of various information sources the generate information with different characteristics? What will be the overall quality of the awareness information l they get when they combine and assimilate information from all these sources together? How do they know what is available, what are its characteristics and how do they choose the quality based on that knowledge?

These are questions that this dissertation addresses and the above issues from the core of the research problem.

1.3.5 Reasons for the Prevalence of Heterogeneity

Some of the reasons for the presence of heterogeneous environments:

1. The nature of work in this day and age demand us to collaborate with organizations and people who are beyond the boundaries of our own organization. Another motivation is the technological innovation such as the Internet, the World Wide Web, wireless technology, to name a few, have made collaboration possible and cost effective as it allows us to leverage the expertise, skill and manpower of other groups. The issue that is of concern to this research is how efficient is this collaboration. Different organizations have already in place their applications, groupware and infrastructure. Thus when the need arises to collaborate not only does a group have to work with a different group but also a different environment.

2. Collaboration among different organizations may be short –lived or continue long term. Depending on the duration the incentives to adopt the same environment, as other organization may not always be present. A certain organization may be equipped with only certain types of infrastructure. The cost involved in acquiring different applications, and infrastructures may be prohibitive. Furthermore adopting new applications involves changing our work habits and getting the appropriate training. Also we may already be collaborating with others who use the same environments as we do and it is counterproductive to change now. Finally the utility we get with the applications and our environment suits their needs to our work practices.

3. It is not practical in terms of ease pf migration or cost effective to replace legacy systems as well as infrastructure (platforms and networks) that an organization already owns.

4. In spite of state of art groupware applications that are multipurpose and have a variety of functionality to support different needs in collaboration, there are certain niche applications that have to be used depending on the needs of the work. Even if groupware applications can incorporate as much functionality as currently is anticipated, there may be needs of group work that it cannot meet. For this purpose other applications may have to be used.

1.3.6 Motivation for Enabling Awareness in Heterogeneous Environments

The theme of this dissertation is in investigating how awareness can be obtained in groups using groupware applications not designed to work together. The outcome of this research is to propose a solution to obtaining this awareness. We are motivated by the following reasons:

a. The types of project-related information in group work that team members need to be aware of comprises of answering *"Who", "What", "When", "Where", "How" and "Why"*. Such information includes details of the activities that occurred, when they occurred, the people involved; the artifacts created, and can be termed as the *project meta-information*. Irrespective of the applications used, this information, which is generated as a consequence of activities and events by team members, can still be propagated among them. Though applications may not allow artifacts to be shared, such meta-information can be shared. What is needed are mechanisms to enable such awareness propagation.

b. Many technical innovations have been successful in enabling application interoperability, to a large extent. Virtual machines and Java, middleware such as CORBA, use of the HTTP protocol, XML based mechanisms, and notion of Web Services have all enhanced the ability for diverse applications to work together. Our solution seeks to utilize innovations such as above, to bind together groupware applications in an "umbrella-like" framework to enable participants to be aware of the project state.

Both the above considerations motivate the research methodology in creating such an "awareness frameworks", the details of which will be discussed in Chapter 3.

1.3.7 Advantages of Heterogeneity

Before discussing the details of the methodology, we examine one aspect that provides basic motivation for our work. Based on the discussion about heterogeneity it is tempting to suggest that if everyone in a group used a single groupware application system or toolset, which had provision for project-awareness then a framework for awareness, would be unnecessary. For instance a group that uses a system such as Lotus Notes or Microsoft Exchange might not need such a framework. This is true as long as the group is confined to activities that can be supported by the system, every member of the group has access to the same system, and there is no change in the sphere of activities or necessity to collaborate with others who do not have access to the same system. The nature of group work today is such that none of the above factors can remain true for long. The needs of the group change and one system does not fit all requirements. Secondly there is often need to work with members who may be within the same organization but without access to the same system as well as with external members who do not use the same systems. Using a mixture of tools may not only be a necessity but an advantage due to the following reasons:

1. The nature of group work is so varied among groups that one tool does not fit every group's needs. Often we need to use certain specialized tools for specific tasks. Also some applications provide more enhanced features for the certain frequently used functionality. A good example is in hospital care units [Bardram and Bossen 2005] where we find that the staff works together in a variety of modes and using various types of artifacts. Some are electronic, some paper-based, often collaboration involves a short face-to-face conversation or marking up on a whiteboard in the center of the care unit. In such cases it is not possible to suggest that everyone use an electronic groupware application. There have been attempts to include all the functionality into one environment such as collaborative editing, messaging, etc. but as commonly observed in everyday workplaces a mixture of tools are used.

2. The move to build all the functionality within one groupware application may not be practical. This is because some applications are so specialized that it would be difficult if not redundant to duplicate the functionality by building it within the common groupware application designed to provide all functionality. For instance a version control system such as CVS is a highly versatile tool. Integrating such an application within our work environment in a seamless manner is definitely more beneficial than rebuilding version control within the common groupware application. Using a mixture of tools actually improves productivity in many cases.

3. Another aspect of group work is that as far as coordination is concerned, it can be highly coordinated and therefore automated. The use of workflow systems is an example of such groupware. On the other hand, collaboration could be ad-hoc and very often opportunistic i.e. the nature of collaboration could be transient. Often group work falls in between these extremes and the degree of coordination plays an important aspect on the kinds of tools that could be used. Using a mixture of applications is necessary to accommodate this wide range.

1.3.8 Research Question

Having examined the fundamental issues of our research problem, which are heterogeneity and awareness propagation or more specifically effective awareness propagation, we can summarize the research question that is addressed by this dissertation as:

"How can effective awareness be propagated among human users (and software programs) engaged in group work that employ a mixture of applications (tools and groupware) that work over a variety of computing infrastructures of varying capabilities?"

A more concise form of the same question could be:

"How can effective awareness be supported in heterogeneous collaboration environments?"

In other words we wish to find out what the various issues involved in effective awareness propagation in heterogeneous collaboration environment are. Is there a methodology to be able to provide relevant awareness information that is timely, and is in a manner that is most useful to the recipients? The overhead in accessing this information and the distraction in being informed should be no more than absolutely necessary. Furthermore we must ensure that the awareness is propagated to only those recipients who have the have appropriate privileges to receive such information. Last but not the least, can "*awareness about awareness*" be supported since such *meta-awareness* is so useful in group work.

1.3.9 Research Objectives

Based on the research question that was raised in the previous section, the objectives of this dissertation can be formulated as:

1. Investigate the issues that influence effective awareness propagation over heterogeneous environments. There is a need for clear understanding of essential requirements for awareness propagation as well as the inherent problems that heterogeneous environments possess. This understanding is essential to gauge the impact of such environments on awareness propagation.

2. Investigate if it is possible to create a methodology that is based on our understanding of awareness and heterogeneous environments that includes mechanisms to be able to provide users effective awareness in spite of the heterogeneity in the environments they use. In other words is there a methodology to can satisfy the above requirements stated in the research question?

3. If such a methodology can be found, investigate the feasibility of implementing that methodology. Often the issues involved in implementing or engineering a theoretical methodology are complex in themselves and lend the solutions to be impractical. This is a critical requirement because the objective is to devise feasible solutions that can be used in everyday collaboration. The primary problems that have to be tackled stem from the applications and systems that are in current use that were not meant to work together.

4. Devise and demonstrate the methodology that is created.

5. Analyze the outcome above in terms of its effectiveness, capabilities and limitations if any.

1.4 Research Methodology

The steps taken in this research effort towards achieving the above objectives are as follow. The details of the steps undertaken are described throughout this dissertation.

1. Literature review of awareness propagation, with a emphasis on heterogeneity in the systems and applications involved (**Chapter 2**).

2. Ascertain the impact of heterogeneity on effective awareness propagation. This step is essential to gain an understanding of how awareness that has to be propagated using applications and the infrastructure that are varied is impacted by the variability **(Chapter 3)**.

3. Identify requirements for effective awareness propagation taking into account the impact of heterogeneous environments (**Chapter 3**).

4. Create a theoretical basis for mechanisms that can enable effective awareness propagation **(Chapter 3)**.

5. Discuss the engineering issues involved in creating feasible implementation of the proposed theory **(Chapter 3)**.

6. Validate the applicability of the theory and corresponding mechanisms **(Chapter 4).** The effectiveness of the theory and mechanisms must be demonstrated. The details of the validation methodology and experimental framework are discussed in Chapter 4. Briefly, simulations of realistic collaboration scenarios with human users have been used as a means to validate the research outcomes.

7. Analysis of the research effort outcomes and conclusions with regard to the effectiveness, capabilities and limitations if any **(Chapter 5)**. This analysis is based on the validation results as well as the entire research process.

8. Analysis and recommendations on future work with respect to this research effort **(Chapter 5)**.

1.5 Research Outcomes

The outcomes of this research effort as follows. The subsequent chapters discuss each of these outcomes wit all relevant details.

1. **A characterization of the relationship between awareness quality and the heterogeneous environments:** A relationship between the heterogeneity of collaboration environments and the quality of awareness has been proposed. Awareness information characteristics and quality factors of awareness have been defined. Environments are considered at the most basic unit namely the source of awareness information and the medium that delivers it. The relationship among the three ideas, viz. quality of awareness, awareness information characteristics and the sources and corresponding medium of information is the basis for the theoretical framework and mechanisms that are proposed.

2. **Notion of Awareness Attributes:** In the above effort to better understand heterogeneity in environments and awareness, the notion of considering awareness information in terms of *awareness attributes* as is proposed. All types of information that is related to a particular group's collaboration that participants need to be aware of is part of this set. We define this as the *awareness information set* or *"Awareness Attributes"*. Information about the project goals, policies and rules (project Meta-information), information about the various artifacts created as part of group work, information about the activities of participants, their location, are some of the main types that make up this set. A group effort project is characterized by this set of information. The values of these attributes will change as the collaboration progresses. With members of a group using a variety of applications, being displaced in time and space and having to continue working there is much variety in this information set. The use of the terms "attribute" implies there is some "value" to that attribute. Awareness attributes are really a means to name and carry information generated by information sources. Any information can be viewed in terms of some value of some attribute. This concept is explained in detail in Chapter 3. However the notion of attributes helps us denote information as well as add meta-information to it, which is essential to our solution.

3. **Requirements for Awareness Propagation:** The study of the research question reveals a set of requirements necessary to enable awareness propagation. These requirements are divided into two parts, *physical integration* and *information integration*. This research effort has addressed information integration in detail. However the overall requirements have been formulated that apply to awareness propagation in heterogeneous environments.

4. **Generic Awareness Framework Architecture**: Based on the above requirements and literature review of similar endeavors, a generic awareness framework architecture has been proposed. The architecture addresses the physical requirements with four main parts viz. interfaces between information sources (applications, devices such as cameras, microphones, sensors and others)

and the rest of the awareness framework, an awareness propagation medium which is the substrate over which information is transferred, a middleware that integrates the information from various sources and an awareness model that deals with the integrating the information. The details of this framework are in Chapter 3. The focus of this research effort is the awareness model.

5. **Theoretical basis for Addressing Information Integration – The Awareness Model**: In order to address the requirements of information integration from multiple disparate sources, a central logical element called the awareness model is proposed. The awareness model defines the manner in which users and information sources are connected, and provide the users with the ability to select (search and browse) information sources based on their awareness information characteristics. Users can choose to receive information from these sources for extended periods of time. Apart from information from the sources they can find out information about the status of other users who interact with these sources and their state of awareness. A privilege mechanism is used to ensure that only those users with the appropriate privileges can access information sources, meta-information about those sources and information about other users.

6. **Awareness Map**: The awareness map is a any user's view of his or her awareness model. It provides users a complete view of all the available awareness information sources and their meta-information description. The status of other users, the status of applications they are using, their levels of awareness and various aspects that the awareness model keeps track of can be accessed through one's awareness map. Using the awareness map one can search and/or browse through the available information sources, select sources and subscribe to obtain information from a collection of sources by using a mechanism known as the focus. Enhanced awareness about the state can be obtained using the map. A user with appropriate privileges could us the same map to manage the awareness model itself viz. manage user account, privileges, sources and other administrative functions.

7. **Validation Methodology and Experimental Framework**: In order to validate the effectiveness of the awareness model, a validation methodology has been created. Realistic collaboration scenarios with the awareness model have been modeled and the collaboration has been simulated. The actors in the simulation are humans. Thus the human element in the collaboration dynamics is preserved. An experimental framework has been designed to create multiple scenarios with different collaboration environments (applications, systems, actors) and capture data from each run.

8. **Awareness Simulator**: In order to model and run simulations an awareness simulator has been designed and developed. The simulator is a multi-user application. Collaboration scenarios have been created and run with the corresponding data collected and analyzed.

1.6 Contributions of this Dissertation

The above outcomes from this dissertation are instrumental in the following contributions of this research effort. The contributions of this research effort are as follows:

1. **Awareness Propagation in Heterogeneous Environments**
   The notion of a generic awareness framework as a means to integrate various different information sources and enable awareness propagation is one of the key ideas. A simple four-component architecture has been proposed. With a framework such as this information sources can be "plugged-in" and made use of. Though much work is still needed before information sources can be seamlessly and easily integrated into such frameworks. However the contribution here is in the idea of such a framework. This research has made an effort to bring together awareness, an amorphous concept, difficult to capture and communicate, as well as the problems that heterogeneous environments possess, and provide some insight into how that challenge can be addressed.

2. **Tailoring the Quality of Awareness in Heterogeneous Environments**
   The primary contribution of this research effort is in recognizing that it is not enough to just have access to information but the information should be really useful. In other words it must possess the quality desired by the recipient. However in the face of heterogeneous environments this is not trivial. A key component in enabling that is the awareness model, as an enhanced directory of information sources that allow users to tailor the quality of awareness received form various sources.

3. **Improving Collaboration in Groups**
   There is no doubt that better awareness, more specifically effective awareness aids group work. All the contributions listed in this section are towards enhancing awareness and improving group work. However some contributions of this research such as the following, can be particularly useful to collaborative projects:

   a. **Ability to maintain awareness of the group collaborative effort through a unified view:** The awareness model helps users maintain awareness about the state of other users viz. their location, activities, the applications they interact with and various other aspects. This *unified view* of various aspects of the group's collaborative work provided by the framework will enhance the communication, coordination and execution of all project related activities, for all the participants. The ability to take decisions is supported aiding in planning and ability deal with changes in group work is also enhanced. These activities rely on being able to obtain a coherent picture of the project.

   b. **Meta-Awareness**
      The awareness model helps users obtain awareness *about their own level of awareness* for instance, *who else knows what I know*, which gives

23

someone a sense of his level of information with respect to others. It helps users be cognizant of the awareness that other users possess (*what is so and so aware of*). This enhanced awareness we term as meta-awareness.

c. **Opportunistic and Ad-hoc Collaboration Support**: If collaboration within a team is well understood then many aspects of collaborative work including the groupware application support can be tailored to ensure that activities proceed smoothly. Often collaboration is opportunistic and ad-hoc with no pre-determined plan. The opportunity and situation itself presents many incentives to engage in the collaboration and is found to be productive depending on the existing situation. Such collaboration is typically short and task-oriented as opposed to long-term exchanges. Effective awareness is a prerequisite to initiate such opportunistic collaboration. Also due to its ad-hoc nature generally there are no means in place to capture record and disseminate the knowledge about the collaboration and the consequences. Thus its is essential that the awareness about the collaboration event and its details must be disseminated all the concerned parties to whom it is of relevance. With ubiquitous substrates such as the Internet and advancements in handheld computing, such collaboration is possible to some extent. Having an awareness framework and an awareness model that can integrate such a paradigms into a more conventional coordinated group effort is an advantage. As the devices and applications used by actors are connected to the awareness model via the framework, the awareness is propagated immediately, enabling all concerned to participate in the process, which may be very essential to the success of the effort.

d. **Context-Awareness and "Awareness-Contexts"**
Apart from knowledge of someone's activities, having cognizance of their location, environment and other attributes can provide context for our own actions. We may act differently if we are aware of the context with which certain events took place. Context-Awareness in applications [Dey 2001] has gained prominence and wide spread interest. Innovations in ubiquitous and pervasive computing allow one to ascertain the user's context and have resulted in applications such as location-based services and "smart rooms". The awareness model contributes towards context awareness by providing a user information about other's contexts as it conveys to the recipient information about various aspects of others. This helps the recipient infer their contexts of work. Another way is to enhance the usefulness of awareness information by making it more relevant to the user's own sphere of activity. The awareness model and framework allow users to select and tailor the quality of awareness. However the same model could also be used by intelligent agents that could monitor not only activities occurring in the group but also the human actors context and bring to the actors notice information that may be relevant to his/her context of work. A related idea is that of "Awareness-Context". The

awareness map allows users to subscribe to obtain awareness from information sources for extended period of time by creating a "focus". Multiple information sources can be grouped within the same focus. These sources can be unrelated but the user has the ability to group them with respect to some context that he chooses. Multiple such foci could be created. Also the user has the ability to name and define each focus as he chooses. Thus the mechanism of creating foci can help in creating a context around a group of information sources. Sources are not rigidly bound to these contexts and can be moved from one context to another. The ability to categorize the information sources can be very useful especially when one is inundated with a lot of information. Just as objects go out of focus when they are distant and come into focus when they are near, information that is more relevant to others sphere of activity could be brought to the users attention while information from sources least frequently accessed or least relevant to a user will not distract the user. Such functionality can be accomplished with intelligent agents and user-interfaces that are geared towards enhancing user awareness.

e. Historical awareness
The awareness model acts as a transcript of all collaborative activities that occurred. Users can query and browse this to ascertain what happened in the past. They can also find out from the awareness model, the state of awareness of other users in the past. This knowledge of users' awareness in the past can be very useful in many situations where one needs to find out why certain decisions were taken or certain actions carried. It helps to maintain a trace of activities and accountability of users.

4. Information Access Control
The awareness model incorporates mechanisms of information (and thereby awareness) access control essential to collaborative work.

5. Leveraging the utility of heterogeneous groupware systems
The awareness framework will allow groups working together to obtain visibility of the total group effort irrespective of the actual applications and systems being used. Though these applications may not share the ability to directly interoperate by accessing files and artifacts, the framework will allow sharing of information regarding the user activities and other meta-information, thereby enabling coordination and execution of project related-tasks. This approach will allow teams to leverage the benefits of using different groupware applications. Integrating newer applications into group work will not be a major impediment and teams can collaborate with other groups using dissimilar groupware since information needed for coordination can still be shared. Thus the benefits of heterogeneity described in the previous sections, (such as the ability to utilize applications of certain functionality not found in other applications) can be exploited.

6. **Supporting User Mobility**
    The awareness framework and model can be used to support collaboration scenarios where group members are distributed in space and time i.e. not co-located and interact asynchronously (and where applicable synchronously) with each other. Awareness propagation will not only occur among such participants but also among team members who may be mobile. In spite of being mobile, members can be aware of the group effort and others in turn will be aware of various aspects of the noble team members.

7. **Awareness Model in Enabling Workflow:**
    The awareness model may be used in environments that support workflow. A workflow system is used to automate various aspects of a process and may involve humans in the execution loop depending on the process. Thus the workflow engine has to maintain an awareness of various aspects of the ongoing processes in order to effectively accomplish the processes. An awareness model is essential to success of the workflow. If the workflow system uses heterogeneous applications and services and combines them in a service-oriented paradigm [Singh and Huhns 2005] then the challenges are greater. Even if all components that execute the processes are built to interoperate with each other there is definite need for maintaining various aspects of the process awareness. The above benefits that the awareness model and framework provides can be utilized in workflow systems. At the Smart Internet Programming Laboratory [SIPLab 2005] at the Concurrent Engineering Research Center [CERC 2005], West Virginia University (WVU) there is ongoing research in creating a "Context-Aware Workflow-Centric Collaboration Environment" named EkSarva [Reddy, Selliah, Bharadwaj, Yu, Kankanahalli and Reddy S. 2004b][EkSarva 2005]. There are efforts underway to incorporate the awareness model into the workflow environment.

8. **Validation Methodology, Experimental Framework and Awareness Simulator and Frameworks**
    Evaluation in the domain of Computer Supported Cooperative Work has been for long an active area of research. Apart from the difficulties involved in evaluating the functional requirements and non-functional requirements (reliability, performance, security), that are evaluated in other types of systems CSCW systems must also be evaluated to ensure they are really useful in enhancing the quality of work of the group. Though a system may have all the desired features, new users may be less inclined to use a new system if it involves some learning. We often see people using tools that are older just because they well accustomed to using it. Thus getting users to use a system often enough to analyze its impact itself is a problem. Evaluating awareness in CSCW systems is equally difficult. One of the contributions of this research has been to propose a methodology to evaluate the awareness model. The awareness model is created and simulated with human participation. An application named the "Awareness Simulator" has been built to demonstrate the above process. The hope is that the evaluation methodology can be extended beyond the awareness model itself to awareness

evaluation in CSCW systems. This is one area of future work to be embarked on. Chapter 4 discusses validation process.

9. **Understanding and Characterization of Awareness**
   In the attempt to devise solutions to address the research question, a characterization of the impact of heterogeneous information sources and mediums on awareness quality has been gained. Furthermore the notion of awareness attributes has been proposed. This is very useful in characterizing awareness information that is to be propagated. These findings could aid the overall research challenges in the field of awareness. They could promote better design and development of awareness systems in CSCW. The validation methodology and framework have proposed a simple formalism to model awareness propagation (details in Chapter 4). This formalism could be extended and developed to model the flow of awareness information in collaboration. Just as software design can be factored into software design patterns, there is the idea that collaboration itself can be characterized as patterns. Much work needs to be done to completely specify the formalism itself. However if the concept of 'awareness patterns' is realizable, then the formalism proposed in this research could be used as a means to specify such awareness patterns in collaboration.

1.6.1 Impact on the overall Quality of Collaborative Work

It is well understood that awareness is essential to the success of collaborative work. However the more effective the awareness is it enhances the quality of work by providing adequate support to all the fundamental elements of collaboration as discussed in the beginning of this chapter. The list above indicates the manner in which the research outcomes seek to improve the overall quality of collaborative work especially in environments with mixture of applications and infrastructure.

1.6.2 "One-from-All" in place of "All-In-One" Philosophy

In order to enhance the quality of collaboration various research and commercial efforts have provided many wonderful solutions be they massive backend servers for messaging and workflow to the individual devices and user-interfaces that aim to make interaction user-friendly and effectively accomplish work. However on can see a trend or almost philosophy that is prevalent in a large number of such solutions that claim to overcome barriers to collaboration. This is the notion of providing an ***all-in-one*** solution that takes care of every aspect of collaboration through one logical system (at least as far as the user is concerned). However heterogeneous environments are a fact of life. Thus it is seen repeatedly that no one product can overcome every barrier and even if they do take care of the majority of issues, one always encounters requirements that cannot be met. Often the greatest barrier being that new collaborators we wish to work with are using a different system that our system cannot work with. Furthermore to make every aspect of the *all-in-one* solution work, one has to acquire all the components that belong to the product family, which involve cost.

This research efforts calls for a change in perspective to this problem. Rather than looking towards one system for a complete solution, we advocate embracing the heterogeneity. Awareness is essential for successful collaboration but heterogeneous environments impede its effective propagation. However awareness itself is the solution. This is because awareness of various aspects of the heterogeneous environment can significantly alleviate issues caused by the non-uniformity. As opposed to using just one system we advocate use of various systems and applications but while maintaining "*oneness*" through the propagation of effective awareness. Thus a departure from the *all-in-one* mantra to what is a "*One-from-All*" philosophy. As seen from related work in subsequent chapters, this very intuitive methodology (if one chooses not to call it a 'philosophy') has been successfully demonstrated, and it is by no means the intention of the author to claim credit for it. The intention is to merely advocate it as an approach with great promise.

1.6.3 Application Scenarios

The notion of the awareness framework and the awareness model developed can be used in variety of real-world applications. Especially in collaborative efforts that require users to be employ mixture of applications and work with a varying infrastructure. In addition applications where users have to be mobile and have to engage in opportunistic collaboration can benefit tremendously from the awareness support. Typically such requirements can be found in a corporate office environment of various industries (information technology, services such as insurance, law and finance). Also with knowledge intensive industries such as software development, healthcare and many services such as insurance being distributed globally, awareness support is essential in this twenty-four by seven paradigm of work. In domains such as hospital administration and healthcare, military and emergency services awareness support is essential and the awareness framework finds extensive use. It is easier to use the framework as the infrastructure in these domains tends to be more homogeneous (especially military). Another example is in the manufacturing industry and workflow within an organization.
Other applications are in the creation of personal portals such as (Yahoo.com and MSN.com). So far we see that users have access to content provided essentially by these portals. Any customization has to occur with the choices provided by the portal host. However through the use of an awareness model users may wish to create a portal of portals i.e. the information choice can be made out of various portals and the awareness model acts as the integrator of these information sources. Especially in interdisciplinary research we find groups of researchers working together as larger group. The awareness of each sub-groups progress is essential not only manage the project itself, but by the very nature of research information about the current state of another group, their activities, their practices can tremendously enhance one's work practices and methods.

1.7 Overview of the Dissertation

The rest of this dissertation is as follows: **Chapter 2** discusses the background and related work that is present in literature with respect to the types of awareness in CSCW, as well areas most relevant to this work viz. awareness frameworks for awareness propagation and awareness models. The theory, concepts and all relevant details about the main contributions of this dissertation viz. awareness quality, the awareness model and awareness framework for awareness propagation in heterogeneous environments are discussed at length in **Chapter 3**. The awareness model theory is validated along with description of the validation methodology and experimental framework in **Chapter 4**. We conclude this dissertation in **Chapter 5** with a description of the results of the validation and their analysis, conclusions that have been drawn and the manner in which this research work holds great promise for many future research efforts and endeavors that seek to apply it in the real –world.

Chapter 2
Background and Related Work

This chapter describes the background and work related to the research in this dissertation. The objective of this chapter is to present some of the key notions that have inspired this dissertation and form the basis of the research conducted. The entire discussion is divided into two main parts. After a brief overview of the awareness-related research in Computer Supported Cooperative Work we delve into work that had been done with respect to awareness support, especially frameworks that integrate awareness sources. This is followed by detailed descriptions of some prominent awareness models.

2.1 Awareness in General

Awareness in group work has been well researched resulting in a large body of literature. While the information technology perspective has been primarily concerned with designing and developing systems to promote and enhance awareness in groups in terms of information flow, social scientists have been involved in studying the nature of awareness and how it impacts human behavior. Gross, Stary and Totter [Gross, Stary and Totter 2005] [Totter, Gross and Stary 1998] provide a very good comparison of the research from both perspectives. They emphasize the necessity of a unified approach to awareness research and reveal their insights from such an approach. By summarizing a vast amount of literature they include a succinct yet comprehensive background about awareness in CSCW in terms of the origins and foundations.

Awareness research literature reveals that awareness about various types of information that one can obtain in group work have been identified. This awareness is related to people's workspaces, current activities, location, situation, past behavior, and state of mind among others. Consequently various terms have emerged such as *workspace awareness, group awareness, activity awareness, project awareness, situational awareness, informal awareness, social awareness, historical awareness* and *context-awareness.* One of the foremost efforts on providing a definition to *workspace awareness* and *group awareness* in CSCW is by Dourish and Belloti [Dourish and Belloti 1992] whose definition of *"awareness is an understanding of the activities of others, which provides a context to for your own activity"*, has been widely accepted and their work is considered authoritative. Greenberg, Gutwin and Cockburn have [Greenberg, Gutwin and Cockburn 1996] have provided definitions to some of the above types awareness. Schilter, Koch and Bürger [Schlichter, Koch and Bürger 1997] place the above efforts in perspective with the importance of awareness to collaboration itself and Gross, Stary and Totter include the above efforts in their discussion of the perspectives of awareness research [Gross, Stary and Totter 2005].

Though the above-mentioned types of awareness are referred to in various works, they are interpreted with some differences by various researchers and it seems there is no one standard agreed upon set of terminology [Brush 2005]. Irrespective of the terminology and emphasis, there is consensus that awareness with respect to group work can be considered as information regarding, *Who, What, When, Where, Why* and *How* and

this is informal definition is adopted by this author also. There are various approaches to facilitating awareness among groups [Totter, Gross and Stary 1998] [Gross, Stary and Totter 2005]. Some groupware have awareness mechanisms built–in whereas others systems provide frameworks to collect and disseminate awareness information from different applications that are used in group work. Any element of group work, be it an information artifact, an application, or a person can become a source of awareness information. For instance we may be interested in knowing who is currently accessing a certain file and what actions are being taken. This is typical in a collaborative editor application where we know the person who is editing the document as well as the actual changes being made. The type and amount of awareness propagated depends on the mechanisms employed. It is interesting to note that in spite of the vast variety of systems that deal with different forms of awareness and possess different capabilities, three aspects about awareness in group work are closely related. These are *Quality of Awareness*, the *Awareness Information Characteristics* and *Awareness Sources & Mediums.* This observation made in this research effort is a key element of the foundation for the rest of the research presented in this dissertation will be discussed in detail in Chapter 3.

2.2 Awareness Propagation Systems

2.2.1 Awareness Frameworks

A very large body of work exists with regard to various aspects of awareness propagation in CSCW. This includes providing some sort of a theoretical framework for understanding what awareness is and how it is gathered and used. Experts have worked on understanding the role of awareness in groups and how to design systems to support awareness for work groups. Prominent among them are Dourish and Belloti's well-known work [Dourish and Belloti 1992] in studying the impact of the different types of awareness propagation in teams. The authors compare their observations of an active approach to awareness propagation such as *direct messaging* and *role restrictive* information propagation versus passive mechanism such as *shared feedback* mode. The gist of their findings is that the notion of shared-feedback can be very effectively applied to overcome some of the problem associated with the direct approaches.

With respect to *workspace awareness* Gutwin and Greenberg [Gutwin and Greenberg 2002] have conducted very extensive research. They have developed a descriptive theory to aid groupware designers in incorporating workspace awareness. They propose a three-part framework that defines (a) the elements of information that make up workspace awareness, (b) the mechanisms that are used to gather such information that constitutes workspace awareness and (c) identify how this awareness is used in collaboration. They demonstrate the usefulness of their framework through examples. They show that their framework helps one understand how awareness information is used in various collaboration scenarios ad well as how awareness information is produced. This understanding helps one decide how best to represent and place awareness information within an interface, all leading to better design of groupware systems.

The creators of the TeamSCOPE [Steinfield, Jang and Pfaff 1999] system have sought to identify some key needs of distributed virtual teams and the roles of awareness.

They identified that awareness of team member activity; presence, process and perspective were necessary. Based on their finding they have derived guidelines for design of collaboration systems to support virtual teams and created a web-based system TeamSCOPE, demonstrating the principles. Their work provides a succinct summary of the types of awareness needed, the modes of delivery and the mechanisms needed to propagate awareness.

2.2.1.1 AWARE and Java Context-Awareness Framework (JCAF)

A recent rigorous research effort has been part of creating the AWARE systems architecture by Bardram and Hansen [Bardram and Hansen 2004]. The motivation for this research effort comes from the fact that people who are co-located adjust their behavior with respect to others when they are "socially aware of others". For instance based on not only the location but also other cues such as the current time, the current task that the other person is engaged in, and the emotional state (such as mood) one can gauge the severity of the situation that the other person is dealing with and choose to interrupt or not. Even if they do interrupt they can do so in a manner that does not cause undue disturbance and distraction. Also if the need to communicate is not of a high priority than one may decide to contact the other later or even just leave a message. However when members of a group are *not* co-located we have to rely on artifacts both electronic and otherwise, to mediate this social awareness. Whiteboards with list of name, presence in a meeting room, the online calendar and even the amount of paper on one's desk can all act as cues to the context of some person and can provide others the requisite social awareness. To facilitate such a "context-mediated social awareness" as the authors term it, they have created a generic architecture known as AWARE which is built on a Java based framework viz. the Java Context-Awareness Framework [Bardarm 2005]. The framework seeks to provide mechanisms for mobile users employing various types of handheld devices as well as stationary users on desktop machines to maintain a sense of social awareness, which is enriched by the context they are in. Thus apart from information such as location users can choose to set cues about their status, and activity. The environment itself can provide various cues about the user's context and this is gathered by the framework through context monitors and actuators. The authors deployed this framework in a hospital environment where they found that such context-mediated social awareness greatly enhanced the collaboration that occurs among its staff.

2.2.1.2 iScent

Anderson and Bouvin have created a framework called **iScent** [Anderson and Bouvin 2000a] [Anderson and Bouvin 2000b], to support awareness in distributed teams over the World Wide Web. Their framework allows team members' activities to be communicated through an event notification system to each other. Members can subscribe to events and thereby obtain awareness. More importantly they can use the same system to obtain awareness of *who is aware of their awareness* i.e. the notion of *intesubjectivity*. The authors define *intersubjectivity* as the notion of *"I know that you know that I know"*. Software interfaces are used to "wrap" the groupware and tools used by team members and tie them to the awareness framework. This work in particular has inspired this research effort as it demonstrates that different applications can be integrated

into a unifying framework and awareness can be propagated among the users of the applications.

2.2.1.3 HIPPIE, ENI, TOWER & NESSIE

The overall objective of this research effort by Gross and Specht [Gross and Specht 2001] was to provide nomadic users, the awareness of other nomadic users' activities. By incorporating the notion of *context-awareness* the awareness provided becomes more pertinent to the mobile users. For this they used a nomadic information system, (HIPPIE) which essentially provided nomadic users location-aware information services based on the user's location and context. To augment this with an awareness of other users and their activities, they combined HIPPIE with an event notification environment known as ENI or Event Notification Infrastructure. ENI consists of various sensors and indicators to capture various aspects of an electronic environment (such as changes to an artifact) and indicators that displayed this information in various forms (electronically as in conventional browsers as well as through devices such as lamps). The event notification system relays user-generated events captured by sensors to a central server. These events indicate user activities as well other information that the sensors capture. This information is stored as attribute-value pairs. By subscribing to the occurrence of certain events users can be notified when these events occur. By combining HIPPIE and ENI the authors realized a system where users can make avail of location based services as the environment is aware of their location while in turn they can maintain awareness of other users and their context (location) and activities.

ENI itself is one of the key core elements of a large project viz. Theater of Work Enabling Relationship (TOWER) [TOWER Project Website 2005][Gross 2004]. The TOWER project deals with providing group awareness including their activities ad shared working context. This is done by representing users and their activities on shared artifacts as avatars in a 3D world. The TOWER architecture can be combined with existing groupware applications such that awareness of user activities can be viewed via their avatars by other interested users.

An earlier work related to TOWER was the NESSIE project [Prinz 1999] that dealt specifically with creating an awareness information environment with the notion of event notification of user activities. From a review of the above literature one can see that the NESSIE project was the foundation of the ENI architecture described above.

2.2.1.4 The Context-Awareness Toolkit

A very prominent work in creating not just a framework but also development methodology is the Context-Awareness Toolkit by Anind Dey [Dey 2000]. As part of this dissertation he has presented a rigorous formal treatment of the subject of *context-awareness*, and crafted a framework along with a methodology that specifies how context-aware applications can be built to. Using the design process and framework, he has demonstrated that context can be derived from and presented to applications and thereby enhance the users experience greatly while working in such environments.

2.2.1.5 GroupDesk

Fuchs, Pankoke-Babatz and Prinz have created a system [Fuchs, Pankoke-Babatz and Prinz 1995] to support shared awareness in group work by using the notion of a semantic network. They support awareness propagation among the various users by modeling the environment as a semantic network. Awareness about changes in the environments an and other activities are propagated using this semantic network. The notion is that the events propagate between the sources and the interested parties based on the relationship between them, which is modeled as a semantic net. The event propagation system itself was implemented using a CORBA compliant distribution platform, that allows remote access to distributed objects. A prototype of this system viz. *GroupDesk* was implemented where the main metaphor was of shared workspace (desktop) which could be inhabited by a number of users and artifacts. Users could also have private workspaces populated by object that are accessible and visible only to them.

2.2.2 Augmenting Tools with Awareness Capabilities

Apart from awareness frameworks there has been research in equipping tools with mechanisms to propagate information about the user's activities or interactions using that tool. Mangan, Borges and Werner [Mangan, Borges and Werner 2004] provide examples where CASE tools used by software development teams have been enhanced to be used as part of an awareness environment for such teams. The noteworthy aspect of this research is that these tools already exist and this effort augments them with the ability to propagate awareness of a user's activities with respect to artifacts. The original design of these tools did not provide for such capabilities. Especially in software development where awareness is invaluable, this approach is significant.

Another very similar work is being carried out with respect to commonly used applications from the Microsoft Office Suite [Microsoft Office 2005] viz. Microsoft Word and Microsoft PowerPoint. These two applications CoWord [CoWord 2005] & CoPowerPoint [CoPowerPoint 2005] have been augmented to be used in a collaborative setting by allowing multiple users to concurrently create and edit the same Word document and Power Point presentation. Word and PowerPoint do come with many capabilities to collaborate through the use of NetMeeting where documents and presentations can be shared. However this research effort demonstrates that they can be augmented with further capabilities to be used as collaborative editors by multiple concurrent users. By using these applications users can take advantage of the familiar user interface and capabilities. Of course this involves propagation of awareness related to various aspects of the editing process such as information about which users are present in the session, down to the actual object being edited. The above efforts demonstrate that tools can be engineered to propagate awareness and this finding is very useful to our research as will be clear in Chapter 3

2.2.3 Overload, Filtration and Organizational Memory

Key issues related to awareness such as cognitive overload (Fussell [Fussell et al. 1998]) (Kirsh [Kirsh 2000]), privacy and security have been given great attention in various research efforts. In the Moksha system [Ramloll and Mariani 1999] the authors try to go beyond the notion of a user configuring the type and amount of information he would want to receive in a project awareness environment. This approach is taken to account for the highly dynamic nature of collaboration for which pre-planned configurations are found to be insufficient. Along with effective mechanisms to propagate and collect awareness information the issue that is central to their research is how meaningful awareness is obtained in a. Kantor, Zimmerman and Redmiles examine how information in a group memory such as historical information, can be queried through filters to provide to enhance work as opposed to obtaining plain activity awareness of other members in their KnowledgeDepot system [Kantor, Zimmerman and Redmiles 1997]. This notion of obtaining "meaningful awareness" is yet another yet another key concept that is addressed in great detail in this research. Meaningful awareness here has been termed as *effective awareness*.

2.2.4 Applications

Some awareness research efforts have focused in specific application domains in which awareness propagation is essential. They have created frameworks that propagate awareness information specific to the domain.

2.2.4.1 Palantír

Palantír is a noteworthy effort that has dealt with providing awareness in configuration management systems that are employed in software development projects. The *Palantír* system [Palantír 2005] by Sarma and Van der Hoek [Sarma and Van der Hoek 2002] provides recipients, various real-time views of the changes made to artifacts during development process, along with information about the severity and impact. These views are intended to help detect and thereby resolve conflicts during distributed development that conventional configuration management systems cannot address effectively.

2.2.4.2 Gossip

Gossip [Farshchian 2000] is a "awareness engine" that has been built to promote awareness of the product being developed. Propagation of such awareness information is essential especially when the development occurs among teams that are distributed geographically. Central to the research is a product awareness model that encapsulates the state of the product development process and various related aspects at any given time. This information is constantly updated and relayed to the members involved in the development thorough a notification system. The author terms this notification system as an *awareness engine* since it implements an awareness model, which in this case is the product development model. The term awareness model here has been used with respect

to the product development model. The next section discusses awareness models in detail.

2.3 Awareness Models

Some key research contributions in modeling awareness in a manner that is realizable and useful in CSCW are described in this section.

2.3.1 Awareness Models and Spatial Metaphor

In an attempt to characterize the interaction among large numbers of entities in distributed virtual environments Benford and Fahlén propose a spatial model of interaction [Benford and Fahlén 1993] [Benford et al. 1994]. Key concepts that the model defines are *space*, *objects*, an object's *medium*, *aura*, *nimbus*, *focus*, *awareness*, *adapters* and *boundaries*. Using these concepts the model defines which objects are capable of interacting with each other. The notion of *aura* defines the influence of an object over a space around it that can enable interaction. *Focus* denotes the subspace on which an object has it attention, and *nimbus* denotes the subspace around which the object chooses to direct its influence. Awareness or more specifically the level of awareness (quantifiable) is defined as a function of the focus and nimbus between the objects themselves. The objects can manipulate their foci and nimbi by positioning themselves in the virtual space to get more awareness or make themselves aware to others. The model is intended to be used not only in virtual world but also in any CSCW application where a spatial metric (a way of expressing position in terms of distance and orientation) can be identified. The objective of the model is to provide generic mechanisms for conversation management in virtual worlds where conversation and floor control is influenced by the proximity of users and the artifacts they wish to interact with as opposed to some pre-determined scheme. Thus the notion of awareness is central to this model as it is the key enabler of interaction where entities can dynamically react to the presence of others. An implementation of this spatial model of interaction is provided through the MASSIVE project where human users can interact in a virtual world using a variety of interfaces with different capabilities, high-end graphics audio, and text.

2.3.2 Beyond the Spatial Metaphor

2.3.2.1 Model of Awareness for Cooperative Applications

Tom Rodden's work [Rodden 1996] on awareness in CSCW is considered very significant. His work on awareness modeling in CSCW is one of the first attempts to understand and characterize awareness in application domains that *lack a spatial metaphor*. The significance of his work is evident as most CSCW applications in the real world do not necessarily deal with physical spaces. Rodden begins by adapting the same concepts that awareness models using a spatial metaphor define viz., aura, focus, nimbus, awareness and others. The objective is to reason about users awareness of other users and objects in a shared space. The term "space" is not used in the physical sense of the word but is the entire collection of objects that is shared by its users and includes the users themselves. This concept is commonly experienced in shared file systems, or versioning software or even shared desktops. Thus the active presence exerted by users, their

proximity to other users and objects that they share form the basis of their awareness about each other and the objects around them. The concepts such as focus, nimbus of a particular user in the shared space are defined as a function of the presence of the user object with respect to other objects adjacent to the user. Awareness is then defined using these concepts. Definitions of both a continuous as well as discrete forms of awareness is provided based on the various combinations of the presence of users with respect to each others foci and nimbi as well as the overlap in their awareness. Rodden then demonstrates that though this model of awareness is general it can be applied where the geometry of the space as well as the shapes of the bodies in space specialize definitions of focus and nimbus. An alternative way of considering focus and nimbus can be as the geometric fields that objects possess around them. Awareness as a function of overlapping geometric fields of focus and nimbus are expressed. The most significant application of this his general model is to CSCW applications. Rodden uses graph theory and expresses the model's definition in terms of a shared graph. This structure is chosen since most applications in computer science can be expressed in the form of graphs. He demonstrates how users can obtain awareness in cooperative applications such as workflow systems, shared desktops, versioning systems, and shared hypertext.

2.3.2.2 Model of Modulated Awareness (MoMA)

Simone and Bandini propose another alternative to the spatial metaphor [Simone and Bandini 1997]. They use the *reaction-diffusion metaphor*, very popular in describing various natural phenomena from fundamental structures, dynamical behaviors in physics and chemistry to patterns of behavior exhibited by living beings. The objective of this model is in its emphasis on the effects of awareness on the entities that consume it. According to their model, entities inhabit space and these entities exhibit *fields* in space. Fields are the means by which awareness is propagated in space and is dependent on the state of the emitter, the value of the field at the source site and the field distribution function. Similarly a receiver's (entity) ability to receive the awareness depends on the state of the receiver the threshold of sensitivity and the sensitivity function of the receiver. The main feature of this approach is the ability to modulate the awareness by modulating the fields of source and sensitivity the receiver. Thus the model is more powerful in providing entities the *ability to choose* not only what to be aware of but the intensity of awareness of phenomena occurring around them. The model also defines rules about how, awareness information is generated, propagated (diffused) and how it affects the receivers all based on the above concepts. Another important feature is that unlike the former spatial metaphor where entities employed a focus that involves orientation, and awareness is computed based on the interaction of foci and nimbi, here the ability and sensitivity of an emitter and receiver to gain information i.e. become aware is defined by the notion of fields and diffusion. The MoMA model also uses the notion of spaces (awareness spaces) as a means to demonstrate how awareness can propagate both within and across applications to demonstrate the notion of interoperability.

2.3.2.3 The 3-Ontology Framework

The authors of the 3-Ontology framework [Leiva-Lobos and Covarrubias 2002] believe that *context* is an essential aspect of awareness in cooperative applications, which he calls *cooperative awareness* (CA). Their opinion is that traditional approach of the spatial metaphor places emphasis on the notion of presence, shared workspace, mutual orientation, vicinity of artifacts and activities of others. Furthermore technological support for awareness has been predominantly related to event promotion, i.e. propagating events that describe the activities, presences and other aspects of users to other users who are interested in the same.

To make awareness in CSCW really meaningful to the users, he suggests a 3-otology perspective for awareness. This is because the awareness of something is always affected by the context in which we perceive it. Also cooperative awareness is effective when applications make the context explicitly aware to the users in certain occasions and keep it absolutely transparent in other situations. Thus the awareness propagation mechanisms must adapt to the situations. According to his model, the context for any awareness information can be described by three concepts, viz. *event*, *place* and *community*. The model provides an ontology using these three fundamental notions. Any awareness information can be characterized with respect to these three concepts and two modalities of awareness viz. *visibility* i.e. that which is visible and explicit and *transparence*, that which is tacit, implied or known. Just as visibility is essential, transparence is needed to avoid cognitive overload. This helps in becoming aware of some information and its context and relationship to other aspects of cooperative work. The author proves the generality of the 3-Ontology framework by locating with respect to the 3-Ontoogty framework, some of prominent earlier work with respective to various awareness models ([Benford and Fahlén 1993], [Simone and Bandini 1997], [Rodden 1993] and others). An implementation of the 3-Ontology framework has also been created named JAZZ.

2.3.3 Specific Awareness Models

Apart from models that try to provide a general framework for awareness in cooperative work, one can find an effort to model for specific types of awareness such as presence awareness in CSCW. Christein and Schulthess [Christein and Schulthess 2002] propose a general-purpose model for presence awareness in cooperative applications. The model makes use of notions such as *presentity*, *watcher*, *vicinity*, *location*, *state,* and *discriminator* to detect the presence of users and user activities in both physical and virtual spaces. The model is based on the Internet Engineering Task Force RFC 2778 [IETF RFC 2778] that discusses a model for presence awareness and Instant Messaging applications. Through the use of vicinity and presentity concepts, the model defines how people present in the same physical space can be aware of each other, how a user's presence in collaborative applications such as collaborative web browsers can be detected to name a few. The authors have developed a practical Java-based implementation of their model and have also embarked on using further IETF standards such as SIMPLE [SIMPLE 2005] (SIP for Instant Messaging and Presence Leveraging) which is an extension of Session Initiation Protocol [SIP 2005] as a variant to the Java client API in implementing their model. Thus their work is very promising since it provides a very practical and realizable implementation of an awareness model whose benefits can be readily obtained.

2.4 Relation to Current Research Effort

This dissertation is inspired by the above work. An important aspect common to all the awareness frameworks described in **Section 2.2** is that they provide for *three fundamental elements* in some manner or form. These are, (1) The actual awareness propagation mechanism, generally in the form of a scalable event notification middleware, (2) The ability for tools and users to be integrated to this middleware through some sort of interface, and (3) Some sort of model that integrates all the information in some metaphor (such as the semantic net in *GroupDesk*) based on which the events are propagated to other users. It is this aspect of the model that determine who receives what and how much awareness. The focus of our research is on the unifying model that integrates all the information that various applications generate as well as helps users tailor the awareness information they can receive. This model that we term as an *awareness model* is central to the theme of this dissertation and is discussed in detail in Chapter 3.

Though the details of how these three elements are designed and implemented may vary we can say that the three make up the essential core of all the above systems. This observation from the above systems has a crucial element on which this research is based on and thus we acknowledge the above efforts and many others that are not listed as being inspirational motivating factors to this effort. The research work on awareness models described above has been greatly instrumental in shaping the awareness model in this research effort and our work is borrows some terms and concepts. Work in awareness being extensive, the above subset has been cited as it has inspired our work. Though our awareness framework is based on the concepts by the above researchers, and built upon the related work, we will discuss how it is different and goes beyond what has been accomplished.

Chapter 3
The Awareness Model

This chapter describes the central theme of this dissertation. This dissertation deals primarily with the issues involved in propagating awareness in heterogeneous collaboration environments. A deeper understanding of how awareness influences collaboration is helpful in this quest. This chapter begins by providing some insight into how awareness is related to a collaboration system's state, user context and user action. In an effort to better understand awareness the notion of **awareness attributes** is proposed as another perspective to studying awareness. Awareness attributes provide a alternative view to dealing with awareness compared to the general practice of referring to types of awareness, that we examined in Chapter 2. The awareness propagated has to be effective. One key contribution of this research effort is in identifying the relationship between effective awareness and heterogeneity in an environment. The fundamental elements of any environment viz. the **source of information** and **medium of propagation** are responsible for the manner in which awareness is propagated. Identifying this impact in terms of a relationship between heterogeneity and the **quality of awareness** is described in detail. Based on this we are able to formulate the requirements for effective awareness propagation – **physical integration** of systems and **information integration**. A general architecture of "**awareness frameworks**" is discussed next, which essentially facilitates physical integration. From this we are able to infer that an **awareness model** is essential to complete the awareness framework and address the information integration aspect.

All the details of the **Awareness Model** proposed are described next. The primary goal of the Awareness Model is to facilitate awareness propagation among users and systems that are placed in heterogeneous environments. Our understanding of the relationship between sources, medium and quality of awareness helps us devise a model to integrate the above elements in a manner such that users are empowered to tailor the quality of awareness effectively. The awareness model has various capabilities to enhance the quality of awareness being propagated to its users. Another notion introduced is that of an **awareness map**, which is a view of the awareness model. The manner in which the map aids in awareness propagation is essential to the discussion of effective awareness.

The next section illustrates the awareness model concepts through realistic collaboration scenarios involving heterogeneous environments that are often encountered. By augmenting group collaboration with one possible implementation of the awareness model described above, its impact on improving quality of work is illustrated. This next section describes the details of model used. We conclude this chapter by justifying that the concepts are practically feasible and can be implemented. The engineering issues that are involved in any implementation are highlighted and in the final section this research is placed in the context of related work discussed earlier.

3.1 Awareness in Group Work

3.1.1 State, Awareness, Context and Action

We examine some concepts as they relate to group work in groups using heterogeneous systems. We use these definitions to build our awareness model and framework.

The key concepts of interest are, **"state"**, **"awareness"**, **"context"**, and **"action"**. In terms of work being carried out by a group of people that involves some sort of cooperation and collaboration we could define these elements as:

3.1.1.1 STATE

One could consider the "state" of a group's work such as a group project to be at any given time the "snapshot" of all the relevant elements involved in the execution of the project. These elements include the tasks completed so far, current activities of the members, artifacts created, and availability of resources. Other key elements include the meta-information such as the overall goals, tasks, deadlines, timelines, resources required, resource constraints, task dependencies, and workflow. In other words if one considers a group project as an entity that is described by these elements then the state or status of the project would consist of the values of these elements or attributes.

3.1.1.2 AWARENESS

The Merriam-Webster dictionary (online edition) defines "awareness" as "*having or showing realization, perception, or knowledge*" [Merriam-Webster Online 2004]. Various definitions of "awareness" are prevalent in CSCW literature as seen earlier in the chapter on "Related Work"**.** Simply put *awareness of a group's work is the cognizance of the state of the group's work*. Thus when participants involved in the project knows the state of the project then they are said to be aware of the project. However apart from human participants, software programs (agents) could also be made aware of the project status. In many cases this is necessary since programs have to execute tasks without human intervention. When the above state information is made available to any participant or program then we say that entity is aware of the state of the project. However as is often the case members of a group *may not necessarily* be working on the same project but still have to cooperate and collaborate based on the needs for certain periods of time for certain types of tasks. Thus "awareness" *is not* just related to the cognizance of the state of the group work that one is a member of, but one's awareness can be about so many different aspects. For example a member of many different groups will have awareness about the all those groups. The awareness obtained from one group may be useful in another work, or it may be irrelevant. One has awareness of various personal situations, which again can affect their activities with respect to their group.

3.1.1.3 CONTEXT

Experts in CSCW have defined the term "context" in various ways each with a slightly different perspective. Dey has discussed the relative merits of the different definitions and has provided a widely accepted practical working definition [Dey 2001]**,**

applicable in CSCW. Dey defines context as, *"Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves."* Awareness and context can be related in the sense that **"awareness provides context for our activities"** and this is evident in our experience of the influence of awareness. Having awareness about not only the task to be done but also various other elements of group work provides a context to our actions. Just as our awareness of the state of a project creates for us some context of the group work, awareness about various other aspects (as mentioned above in the description of awareness) also add to our context. Apart from awareness of the project's state, other awareness such as environmental awareness (cognizance of factors in one's environment) social awareness (cognizance of the people around us) also provide context, which can influence our project related actions.

3.1.1.4 ACTION

"Action" is what a human or program involved in the project will perform. Actions can be related directly to a group effort that an individual is part of. Or as in the case of overall awareness and the context it provides, actions that are influenced by context can be unrelated. It interesting to note that often actions, which do not seem to have any direct relation to one individual's role in the project's activities, can provide context to other participants and in turn influence their actions with respect to their role in group work. For instance if participant A is aware that participant B is at a certain location and A has a project related task that needs to be accomplished at that location, A could notify B to complete the task if it is possible to do so. In this case awareness of B's going to that location influenced A's actions and resulted in his delegating of his task. B's being at the specified location had in itself no direct relation to his set of project related activities.

3.1.2 The State-Awareness-Context-Action Model

Thus based on the manner in humans work together in a group we could express a relationship among the above four elements involved in group work. For an individual with respect to one group's work which he or she is involved in,

STATE: The *STATE of a project is the snapshot of that project in time.*

AWARENESS: *Cognizance of this **STATE** is the **AWARENESS** of the project.* As mentioned above, our total awareness is much more than just cognizance of the STATE of any one project. But with respect to group work with a particular group we can consider it as cognizance of that group's STATE.

CONTEXT: *This **AWARENESS** in turn provides a **CONTEXT** to any participant for his or her activities.*

ACTION: *CONTEXT is one of the key factors that influences **ACTION** and **ACTION** modifies the **STATE**.*
Thus there is the cyclical relationship among these elements. **Figure 3.1** below illustrates this model.
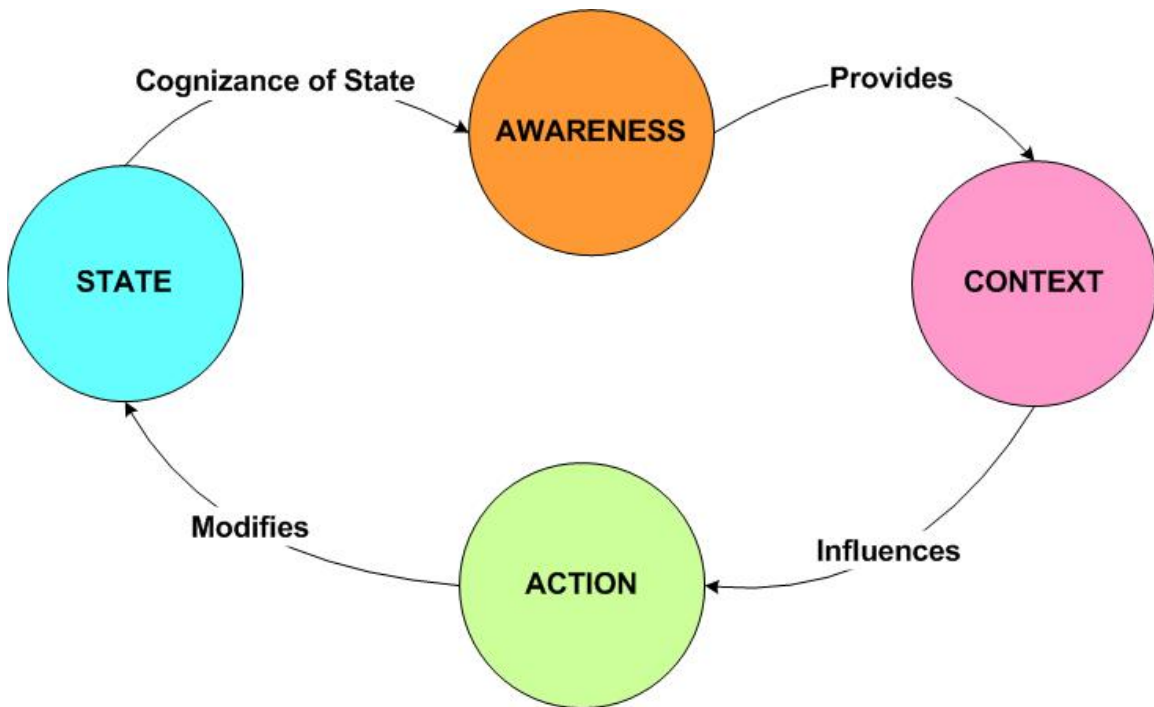
Figure 3.1 The STATE-AWARENESS-CONTEXT-ACTION Model

3.1.3 Applicability of The State-Awareness-Context-Action Model

However based on the above discussion, we know that at any given time, our awareness, the context(s) it provides and the resulting actions that are influenced can have a much larger scope, far beyond the confines of any one group project that we are involved in. Most importantly it is difficult to anticipate and determine how our overall awareness, context and then our actions can impact the group work we are involved in and in turn affect others. This larger scope can be beneficial to the group's work as mentioned above example, detrimental or inconsequential. Thus the fundamental issue is not just enabling propagation of awareness but to enable the prorogation of all *that could be* relevant and most importantly useful to the group as a whole. This is by no means trivial and can be considered one of the great challenges as far as awareness in CSCW is considered.

A beginning must be made to find solutions, and as a starting point one can use the SACA model as a frame of reference even if it is with respect to a specific group's work that the individual is involved in. One's overall awareness could then be the aggregation of many such loops that one is part of. Systems that must involve propagation of awareness could perhaps begin by identifying the loops that their users may be involved in and ensure that awareness being generated in those loops is being propagated to the users of their system. The state-awareness-context-action relationship can be considered as an abstraction of the manner in which awareness affects individuals in group work. Any awareness framework must support the above model elements and the relationship among them.

3.2 Awareness in Group Work

The above model abstracts the manner in which groups work. This is evident from the fact that in collaborative efforts (such as a group project) there is some means (electronic or otherwise) to capture information regarding the various project–related activities as well a repository for the meta-information. At any given time a snapshot of this information is the state of the project. Participants can be made aware of this state, and this provides a context for their activities. Participants' activities are influenced by this context and these actions contribute to changing the project state. Awareness of this new state provides a new context to the participants and the cycle continues. To clearly define an awareness framework that implements this relationship, we need to identify the basic pieces of a collaborating group effort. A framework that supports awareness must include these pieces.

3.2.1 Basic Elements of a Collaborative Group Project

A generic collaborative group effort can be considered to consist of three main elements as shown in **Figure 3.2** below.

Information: This includes all the information about the group project i.e. meta-information as well as the information created as a result of group collaboration. Project meta-information is information about the project. The project's goals, policies, tasks, timeline, resources and other key information elements that describe the project and are essential to the project's success are part of the meta-information set. Also during the project, team members create various information artifacts, either individually or as result of their collaboration with others. These artifacts can be termed as the "artifact information set". One can consider the project "information" to be central to the entire group since all activities revolve around this element.

Participants: Participants are individuals involved in the project. They perform project-related activities to accomplish the goals of the project. As part of their activities they collaborate with other participants. Communication and coordination with other participants is an essential part of their project-related activities. Participants are constantly involved in accessing, creating, and modifying project information.

Systems: Participants use a variety of tools and groupware applications to accomplish their project activities. Groupware applications that support working together could range from plain email, instant messaging, word processors, to advanced version-control systems, sophisticated collaborative designing tools, shared-space editors and workflow systems. Applications could require human intervention and/or consist of software intelligent agents with varying degrees of autonomy. Participants communicate using a variety of applications (email, messaging, telephone, facsimile etc.) over various mediums (computer networks, telephone networks, etc). The ability to collaborate when mobile using hand-held devices and interact with pervasive, ubiquitous computing environments such as "smart rooms" adds another dimension. Thus "systems" could be considered to be the set of all applications being used as well as the devices, computer

hardware, peripherals and networks that form the substrate for those applications to run on.

Figure 3.2 illustrates the three elements as three layers. This is because one can view collaboration as, a *group of participants working together* (**outermost layer - Participants**) *with information* (**innermost layer - Information**) *through a variety of systems and applications* (**middle layer -Systems**).
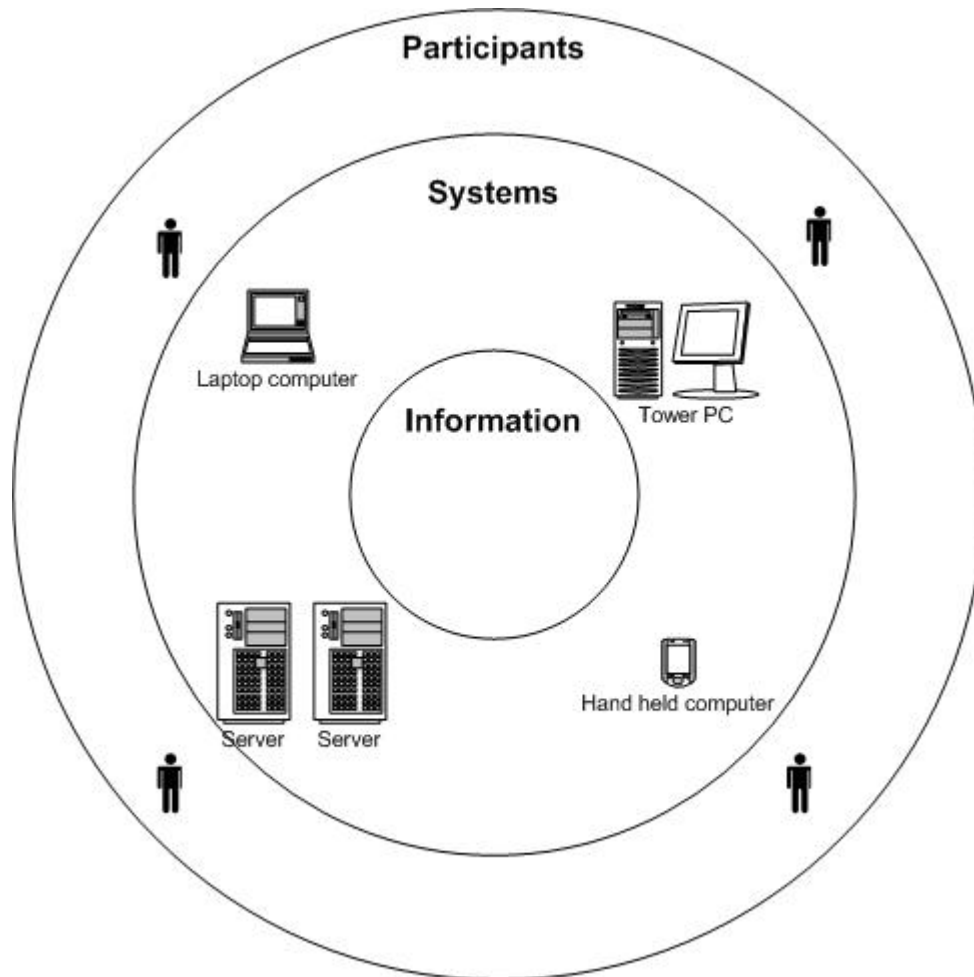
Figure 3.2 I-S-P View of Collaboration

3.2.2 Awareness Attributes

Based on the above description of the constituent parts of a collaborative project we can enumerate the information elements that participants in such a project need to be aware of and the sources that generate them. These elements are our Awareness Attributes can be said to form the "STATE" of the project (the current snapshot of all the project's details) as illustrated in **Figure 3.3** below.
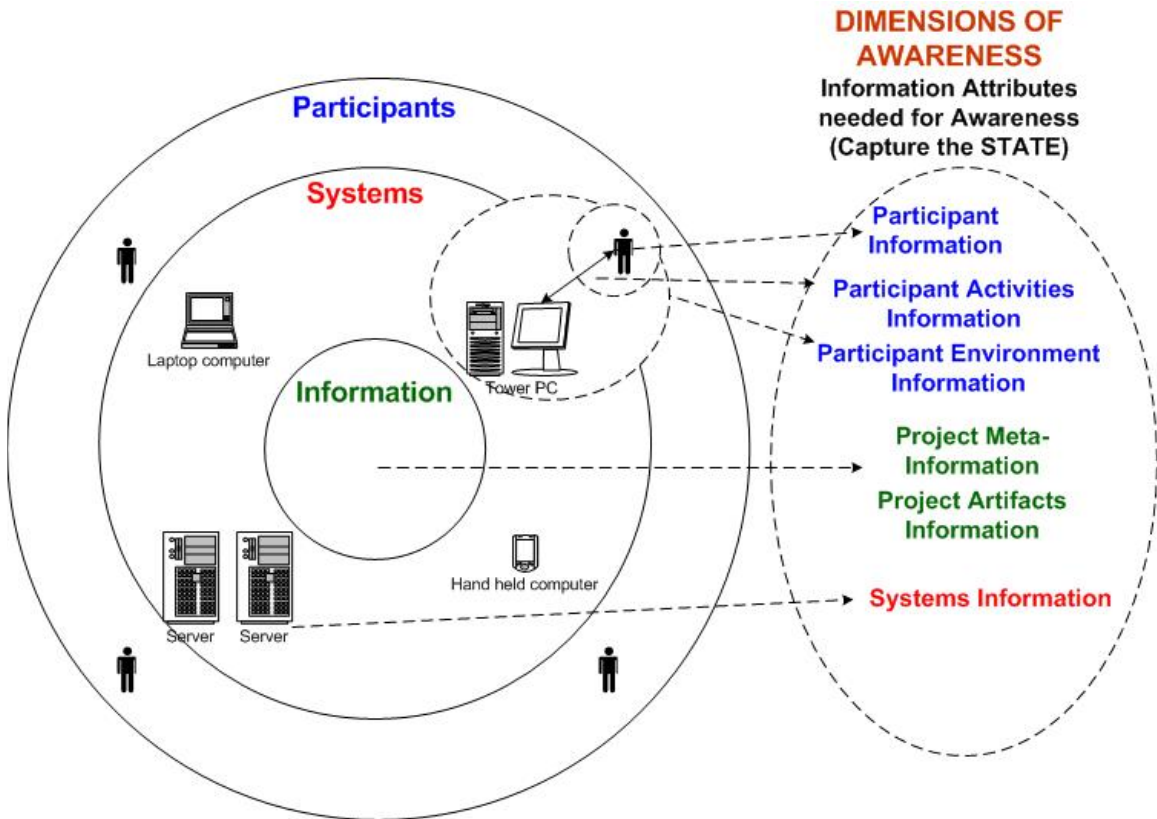
Figure 3.3 Awareness Attributes

1. Project Meta-Information: The project's overall objectives, policies, participant list, participant roles, project tasks, deadlines, timelines, resource requirements, resource availability constraints, dependencies among tasks, and a variety of other information related to the successful execution of the project make up this information set. Participants need to be aware of this information. The project's policies may decide the exact information elements that each participant is privy to, but there must be means for participants to access this information in a timely fashion. This meta –information is subject to change during the execution of the project and participants need to be made aware of the changes too.

2. Project Artifacts Information: Various artifacts are created during the course of the project. Participants constantly access, create and modify information artifacts as part of

their tasks. Apart from information contained in the artifacts, the activities (updates, modifications, deletions) related to them are also of interest to participants.

3. Participant Information: Information about individual participants is anther essential piece. A participant may have one or more roles to play based on the project requirements and information regarding the participant's role, contact information, availability, schedule etc are some of the typical attributes that are of interest to others.

4. Systems Information: Information about the tools and groupware applications used by a team, including details about the software, hardware and networks are relevant. Since participants use these applications to collaborate with each other, knowledge of their capabilities must be part of the awareness information set. With participants being mobile, one needs to know the types of devices that are being used. For instance if participant A needs to notify B of some project-related artifact (file) and is aware that B is currently using his hand-held device as he is in transit, A may just send a short message about the file rather than a large file. Additionally a link to the actual file could be in the message so that B can access the file when he is at a workstation. Such collaboration is possible if the participants are aware of the capabilities of applications involved.

5. Participant Activities Information: Participants need to be aware of the activities of other members of the group. Knowledge of what others are currently doing and what they have done before (a trail of activities) can be of tremendous value to their own work. There must be mechanisms to make participants aware of others' activity. Very often participants' interaction with the applications they use can provide this information. For instance when users work with a version control system, it records the details of who had last accessed the file, what modifications were made and so on. Such activity history needs to be part of the awareness information set.

6. Participant Environment Information: When participants are mobile, awareness of their physical location may be valuable to many project-related activities. Knowledge of participant A's location may enable participant B to delegate a project related-task that A could undertake, thereby taking advantage of A's presence there. Also knowledge of location may help a participant to make use of any location-based services that can help in accomplishing his/her task. Location is just one among the many such attributes of a participant's environment. There must be mechanisms to propagate this information.

When participants of a group project are aware of the above information they obtain a context for their own actions. Participants' actions in turn are propagated to the project STATE as new values of the Awareness Attributes. Team members can obtain this information from the modified STATE and the cycle of awareness propagation and collection continues. Thus awareness of a collaborative effort can be considered as knowledge of the typical awareness attributes listed above.

3.2.3 Benefits of Awareness Attributes

As seen from related works there are various forms of awareness that can be perceived in everyday life. However as there are no specific guidelines and standardized terminology it is possible that terms can be subject to different interpretations [Brush 2005]. This is due to the fact that the current definitions do not specify which of the elements is part of a certain type of awareness and which are not. For instance "*group awareness*" could be considered as the awareness related to the group members' roles, their location, the status of various tasks, current activity to name a few with respect to the work carried out by a group. However one may also call the same by the name "*project awareness*" to mean the same since these elements can be considered essential.
This use of different terminology for the same set of information elements could occur since the members of a project can be considered as a "group". However groups can mean any group such as an online community. Whereas project awareness generally refers to the status of various aspects related to a project such a tasks, timeline, resources etc. However as there is no formal definitions or terminology one may see the same terms being used with varying interpretations. The impact of the lack of standardized terminology in awareness research is that system designers will not be in a position to determine that such and such types of information should be propagated or could be propagated by users of their systems. Further more lack of definitions makes it difficult to infer if there are relationships among the types of awareness (assuming uncovering such a relationships would be beneficial). For example, based on definitions mentioned above, does it mean that the term group awareness is a more generalized form of awareness and project awareness is a specialized from of group awareness? This is not clear.

A complementary approach that is beneficial is to focus on the actual elements of information that make up the awareness information. As often seen awareness supported by groupware systems and awareness systems are described in very specific terms with the elements of information they propagate. The creators may mention the fact their system supports such and such a type of awareness however they are careful to define the elements of that awareness information. These information elements, we term as *awareness attributes*. These are a collection of all the types of elements that need to part of awareness elated to group work. The types are defined but the individual elements are not. Elements of awareness can be categorized into these types. To some extent the approach of mentioning types of awareness attributes and their constituent elements provides a better understanding of awareness itself. Also it helps in designing and developing awareness in groupware systems as these elements are also used while eliciting requirements about what needs to be built. The concept of attributes of awareness information helps in disambiguating the notions of awareness, which in itself can be a complex concept to design for.

3.3 Heterogeneity and Quality of Awareness

Awareness in a collaborative effort can be facilitated by propagating information among the users. However the mere propagation of information does not necessarily make it useful to the members of group. The type and amount of information being propagated, when it is received, the frequency with which one receives information about a certain aspects of group work (especially repetitive events), the amount of overhead and distraction it causes are all essential to determine how useful the information really is to the consumer. These factors are subjective and depend on the individual's current context and perception. These factors could be termed as the "quality factors" of awareness information. Awareness information that is of the appropriate quality to an individual can be called ***effective awareness*** since it is effective in aiding the recipient in his or her work. This is as opposed to information that is received and is of little relevance to the user's sphere of activity. As commonly experienced advances in information technology have mitigated the problems of information access to a great extent. In fact we are constantly inundated by information of various kinds such ticker tapes on television programs to spam sent via email, fax and telephone and the ubiquitous unsolicited "pop-up" advertisements. Apart from that information from online communities such as forums and groups make their way into our email inboxes as electronic newsletters and bulletins. As a result getting to information that is really useful gets bogged in the process of sifting through the irrelevant data that washes across. All this is in addition to the information that we receive that is related to various aspects of our work. Such information itself can be at times overwhelming to manage. In the face of this, the quality of awareness propagated to users is extremely important. The challenge is for us to ensure that information that is really effective reaches the recipient in a timely fashion. More importantly it is necessary to empower end users to tailor the quality of awareness that reaches them. Needless to say the heterogeneity of the environment seriously impacts the ability to deliver such quality information.

3.3.1 Relationship between Awareness Quality and Heterogeneity

It is essential to understand the full extent of the impact of heterogeneity on the quality of awareness. It is obvious that heterogeneity that we refer to is seen in the variety of sources of awareness information and the media that propagate the information to the recipients. How does this impact the quality of awareness? Upon close examination we can see that three aspects of awareness in general are closely related. These are ***Quality of Awareness,*** the ***Awareness Information Characteristics*** and ***Awareness Sources & Mediums.*** **Figure 3.4** illustrates the three aspects and the relationship among them.
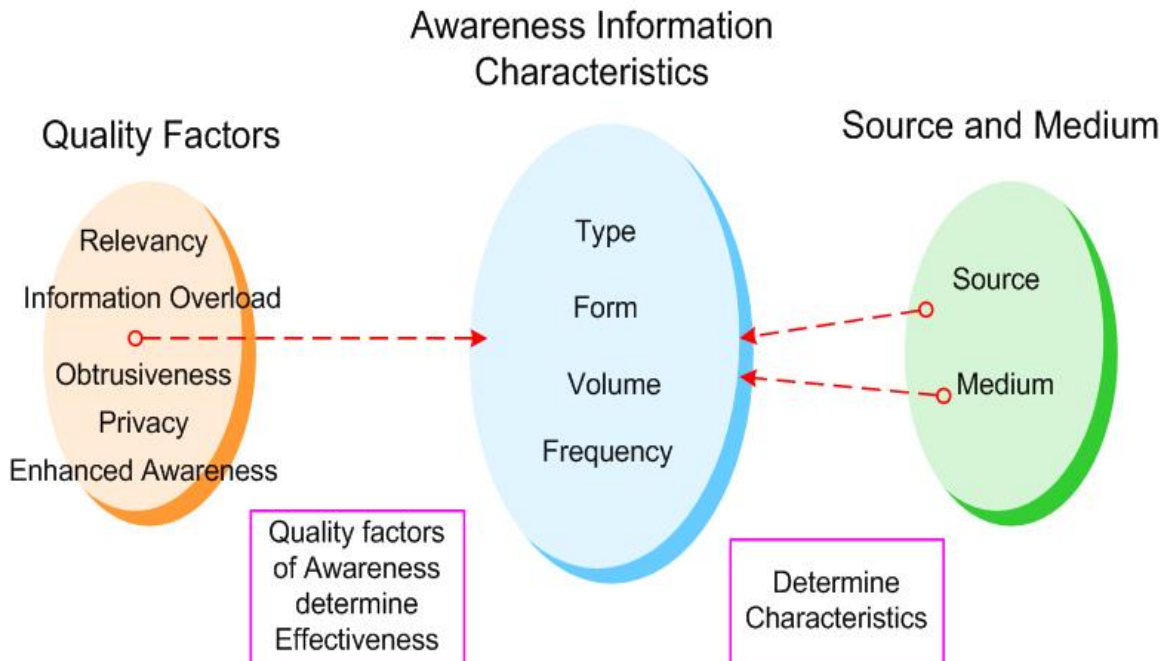
Figure 3.4 Relationship between Heterogeneity and Quality of Awareness

3.3.1.1 Quality of Awareness

The quality of awareness impacts its role in group work coordination and execution. Quality can be characterized by factors such as:

Relevancy: How relevant is the awareness information to my sphere of work, both current and for the future (planning and coordination)?

Information Overload: Am I aware of the appropriate amount? Am I being inundated with too much or not receiving enough? Avoidance of information overload is essential. Often users can be inundated with information in an effort to be kept aware of ongoing group activities. Managing this aspect has been an area of interest in awareness research and various techniques such as subscribing to selective events and filtering techniques have been devised to deal with this problem. On the other hand there are certain types of information that the users must be aware of due to the nature of their work.

Obtrusiveness: How distracting is this information to my current activity? Is it interrupting my current activity? The recipient of awareness information may prefer to be informed in a certain manner. This is similar to the notion of cell-phone which vibrate as opposed to playing a loud ring-tone that can disturb others.

Privacy: Can privacy in the form of access control be established to ensure that only one with appropriate permissions is aware of classified information? Privacy is directly related to awareness and this can have a large impacts on security. A project may have policies on who needs to be aware of certain types of information. Often only a subset of individuals are on a "need to know" basis and the awareness mechanisms should allow such selective access to information. Though awareness and privacy are orthogonal, their impact on security of a group project (system, facility, information etc) can depend on how the awareness is used. Malicious use of increased awareness can compromise

security. On the other hand increased awareness (at the cost of privacy) can enhance security and even safety as seen in video surveillance systems, motion sensors guarding perimeters of facilities, child monitors, and intrusion detection systems programs to name a few.

Enhanced Awareness: Apart from the information content itself there is meta-awareness i.e. awareness about awareness. Some examples are, *"Who else is aware of what I am aware of?"*, *"What is so and so aware of ?"*, *"Who else is aware of my activities?"* and *"I know that he knows that I know"* (intersubjectivity). Such meta-awareness is often essential in organizing access to information and coordination in a group. Furthermore such awareness helps in avoiding the sheer overheard of ensuring everyone has the same information.

3.3.1.2 Awareness Information Characteristics

Awareness Information possesses certain characteristics that describe it. In fact any type of information in general can be described by these attributes:

Type (What am I aware of): Is this about an individual's activity or location, actions on an artifact, conversation transcript, streaming video of a meeting, or a combination of the above?

Form: Is it text, audio, visual?

Volume (How much of it am I aware of): Am I aware of every email exchanged by the group or just a synopsis; is it a long videoconference session? Do I receive a notification for every action taken on an artifact?

Time & Frequency (When and How often do I get that information): In a highly interactive chat session that I am monitoring, do I want to receive every line typed?

3.3.1.3 Sources and Medium

Sources ("Which is the information source?"): Email and IM messages, sensors relaying location, an artifact (actions on it), camera, microphone, user's keystrokes are all sources of awareness information, capturing various aspects of work that other users would be interested in. *Sources* impact the *Type, Form, Volume* and *Frequency.* For instance a sensor-based application will communicate awareness information only when the appropriate stimulus occurs and at the same frequency. High rate of keystrokes can result in high frequency and volume of information relayed.

Medium ("How do I obtain information?"): Wired and Wireless networks, closed-circuit video, telephone (landlines, cellular) are all media. The capabilities of the *Medium* impact *Form, Volume and Frequency.* High bandwidth networks could provide streaming video at much better quality than a dial-up connection.

Quality factors are evaluated from the perspective of the consumer of awareness and how they influence his work. Awareness Characteristics are about the information and so absolute. However the perceived quality of awareness does depend on the characteristics, which in turn are determined by Sources and Mediums. This relationship between the three aspects is the foundation of the solution that is devised to address the heterogeneity impact on effective awareness.

3.4 Supporting Awareness in Heterogeneous Environments

Figure 3.5 below illustrates a typical heterogeneous collaboration environment. We see groups and individuals using a variety of groupware systems and tools of varying capabilities and functionality that work on equally varied platforms, hosts, devices, and networking infrastructure. The users may access these applications using variety hosts from powerful workstations to handheld computers to public kiosks. All these systems and tools are connected the network cloud in the center. This cloud denotes a heterogeneous mix of various types of networks, such as wired and wireless, computer and telephone networks, of different capabilities and connectivity. The single cloud is not to be misunderstood with uniformity. It is a representation of the state of the networked world which is anything but uniform.

Thus the end-user experience is also non-uniform. Users will be able to collaborate if and only if their applications are connected to other applications. The actual movement of information created and used in collaboration flows among systems through these heterogeneous networks. Thus the he quality of information propagation will also be dependent on the infrastructure. But information content that acts as an input to collaborative tasks and the content created as a result of collaboration are one small part of the set of awareness attributes we discussed. Awareness is a much larger notion and thus the mere connectivity of systems is not sufficient to propagate all elements that make up awareness. Awareness information has many elements that are necessary to provide a useful relevant context to the user as seen by the awareness attributes. This illustration depicts a real-world view of the three concentric circles representation of a collaborative project with respect to the connectivity.
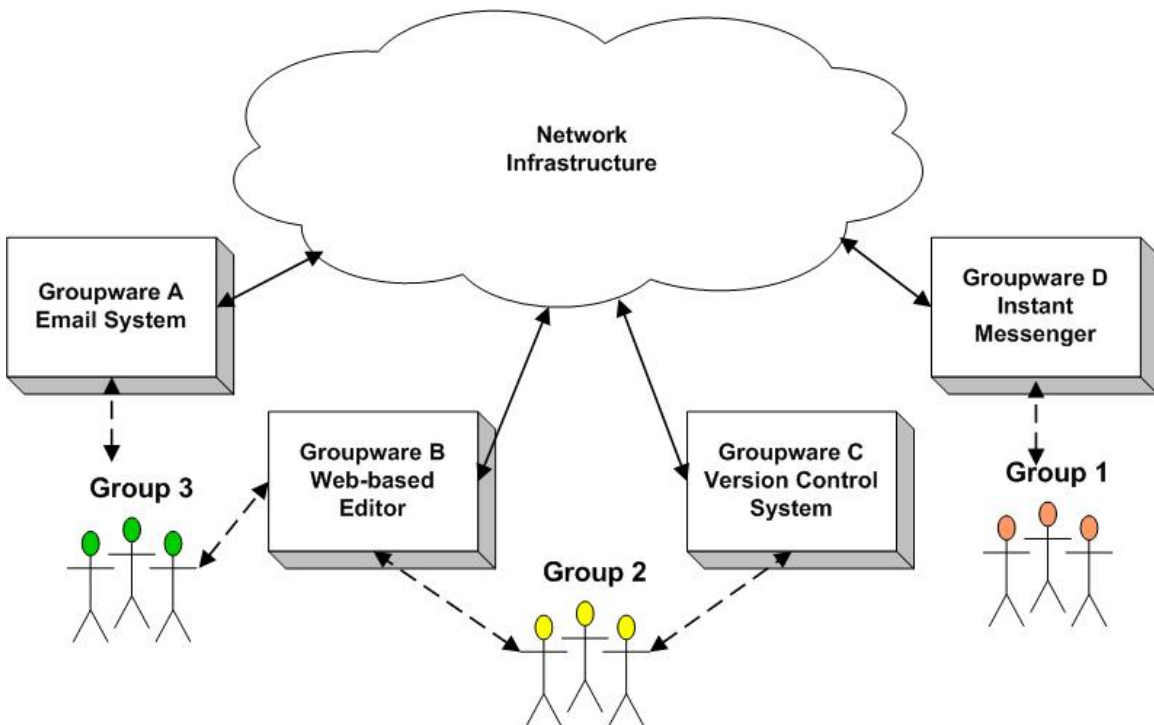
Figure 3.5 Collaboration in Heterogeneous Environments

We use the above view of collaboration in heterogeneous environments as the basis of our solution to support awareness in such environments. To do so we have to identify the requirements of awareness propagation. Having gained an understanding of the role of awareness in group work, the information elements involved (awareness attributes) as well as impact of heterogeneous environments on effective awareness we seek to enumerate the requirements for supporting effective awareness propagation.

3.4.1 Requirements for Effective Awareness Propagation

The fundamental requirement is the ability to integrate various information sources and the media they use to effectively propagate awareness information to the members of a collaborating group. The detailed requirements can be divided into two broad categories viz. those related to the Physical Integration and those related to Information Integration.
The rational for such a division is apparent at the end of this discussion when the requirements are describe in detail. A natural separation is seen that is helpful in the design and development of a solution.

3.4.1.1 Physical Integration Requirements

These are requirements related to the actual movement of awareness attributes across heterogeneous systems and require that systems be integrated to facilitate the same. The requirements are divided into four categories, which are as follows and are described below:
i. Capturing the Awareness Attributes
ii. Representation and Access
iii. Information Flow
iv. Information Access Control and Security Requirements

i) Capturing the Awareness Attributes
 Awareness Attributes and their values, generated by various sources must be captured so that they can be propagated to the appropriate consumers. As mentioned earlier together the awareness attributes of a collaborating group can be said to form the "STATE" of the group.

Capturing Participant Information
Members of the collaborating group should be able to add and update information about themselves. These attributes must be recorded and there must be a means for participants to modify them when necessary. This modification would be dictated by the group's policies. For example a participant may wish to modify preferences that determine aspects of information that he wishes to receive, but may not be able to modify her role in the project.

Capturing Participant Activity
There must be mechanisms to capture relevant details regarding participant activity since it is an essential awareness attribute. For instance an action taken on a file would have to

be clearly captured with details such as the file name, timestamp of the action, the participant who performed it and so on. This information should be recorded as part of the collaboration "STATE". It would consist of attributes, which describe such actions and when these events occur the values of the corresponding attributes would be modified.

Capturing Tools and Groupware Systems Information

Information about the tools and systems being used by project participants is an integral part of the awareness information set. There must be a means to obtaining this information about the devices, platforms, networks and applications used in group work. This acquisition can be voluntary i.e. polling periodically or involuntary by having the systems notify others of their profile when the user performs activities. Also groupware such as workflow systems perform project related tasks automatically. These are essential events in the project that participant's may need to be aware of. Such information is part of the information set that must be acquired from groupware.

Capturing Participant Environment Information

Information about a project participant's environment attributes must be captured. Again this can be done by polling sensors capable of gathering such information or by some involuntary means. These sensors would be on devices that the user would use or be embedded in the participant's surroundings.

All the above requirements indicate that there must be mechanisms employed, which enables the collaborating group state to add these attributes of awareness and during the course of the project obtain their values and update itself.

ii) Representation and Access

Physical Storage

Conceptually the STATE of the project represents all the aspects of the current status of the project including its participants, their activities, the environment they are in and so on. The inputs to the STATE are the values of the awareness attributes. The attributes should be stored in a manner such that their values can be easily queried and accessed in timely fashion by participants seeking information such as the current status of the group effort project (ongoing user activity, user location, tasks completed thus far etc.). Storage is necessary since one may always seek awareness of events that occurred in the past i.e. "historical awareness". For example, activity trails on specific tasks and artifacts. Historical information of events and decisions can be essential in determining how the project has arrived at the current state. Past snapshots of the STATE is beneficial since often we are interested in knowing "why such decisions were taken" and how the present STATE has come about.

Distributed, Synchronous and Asynchronous Access to STATE

Users who are distributed in space and even mobile will access these attributes. Requests from distributed and mobile team members should be handled. Access to attributes will be to ascertain the current state of the project such as current activities of other team

members, their location etc. or state of the project in the past such as a history of actions on an artifact or all previous messages in a thread of discussion. Thus there must be mechanisms to provide for such "synchronous" and asynchronous access of the project state.

iii) Information Flow

Participants need to obtain awareness information from the STATE. The awareness information has to physically be transmitted to the consumers while adhering to the following constraints of time and space. This set of requirements is related to the mechanisms employed to transfer the information as well as access it at the user end.

Time Dimension
Participants may need to be aware of the project's STATE, at the present time as well as in the past. Thus there must be means of transmitting the awareness information immediately in real-time (synchronously) as well asynchronously.

Space Dimension
Since groups will consist of members who are distributed and mobile, the mechanisms of obtaining awareness must be able to provide the appropriate awareness for such users.

Granularity Dimension
This aspect addresses the question "What should one be aware of " or "How much should one be aware of". There must be mechanisms to display the information of the appropriate granularity as desired by the user. This dimension is related to the information integration requirements described in the next section.

iv) Information Access Control and Security Requirements
There is the extremely important aspect of ensuring that all the security requirements are met while integrating heterogeneous sources of information. Sources of information will have restrictions on who is allowed to access the information. Only those users with the appropriate credentials will be allowed access and that too after the authentication. The actual requirements regarding credentials and the authentication procedure will depend on each individual source of information. Security requirements need to be addressed at the physical integration level, as actual "data on the wire" has to be secure, users have to be authenticated as well at the information integration level where access to awareness propagation is controlled. These requirements are seen in the next section. The security requirements will be addressed in the solutions section.

3.4.1.2 Information Integration Requirements for Effective Awareness

The above set of requirements must be met by the physical infrastructure that is responsible for awareness propagation. However the very nature of awareness and more specifically effective awareness imply that great care is necessary in dealing with the actual information generated and transferred to make it really effective. We identify these requirements in this section. Since these requirements deal with collecting disparate information elements that have no relation among them and providing users the ability to derive meaning that is relevant we call this process ***information integration.*** Different types of information generated (email messages, sensor coordinates, streaming video) have no absolute relation, though they may be related within the context of the group's activities. They have varied source specific characteristics (Form, Volume, Frequency). These need to be woven to make the composite picture that is awareness. This integration must occur at the information level. Consumers of awareness must be able to determine quality and control it. This research effort proposes solutions to some of the key requirements identified here. Specifically, the implications of the awareness requirements on information integration can be stated as the following list, which are described in detail below.

- i. Source Context, Group Project Context, User Context
- ii. Unambiguous Representation
- iii. Ability to Tailor the Quality of Awareness
- iv. Establishing Human-Agent Mixed Initiative
- v. User Interface
- vi. Non-uniform access capabilities
- vii. Obtaining Meta-Information
- viii. Integration Process
- ix. Dealing with Change
- x. Extensibility
- xi. Historical Awareness Support

i. Source Context, Group Project Context, User Context
The Awareness Attributes that are captured from the variety of sources, need to be maintained with respect to a common context, which is the context of the group work. The Awareness Attributes are generated by various sources that have no logical and physical relation to each other apart from the fact that they are used in the same group effort. Naturally the formats and semantics of the attributes would be specific to the source as opposed to the project. There needs to be means to view their aggregation as one coherent set with respect to the group as opposed to disjoint attribute-value pairs. This is necessary since team members will be familiar with only the project-specific terms and will not have knowledge of the native attributes themselves. Thus their queries for information will be with respect to the group's context. To retrieve all relevant information based on user's sphere of activity, there must be ways to relate information with respect to the common context, which is the group's work. Thus users can look for awareness information they need from their work perspective as opposed to querying about the source. Using the information the group members will augment their individual context.

For example: Consider a group project where different sets of members use different email services. The members need to maintain awareness of emails exchanged regarding their group project. Normally this would involve everyone adopting the same groupware system such as a web-based online community facility such as Yahoo Groups or MSN Groups. All the users would have to acquire individual accounts and use the particular facility to interact with the group (for instance to send emails to the group on most group-based systems, one needs to login to the groupware). However the members prefer to continue using their preferred email systems but wish to maintain awareness of email messages exchanged in the group. This would mean they have to keep track of other email accounts and CC emails appropriately. If we wish to integrate their email systems together then physical integration is no problem since email systems are well connected. The challenge is to allow emails sent by different accounts to be viewed in the same context as the group's activities. Users should not have to remember individual email ids capabilities. They would only have the knowledge of the groups' members their roles. However they should still be able to browse, search and retrieve messages sent by others. There are other issues such as selectively sending messages to certain recipients as well as the fundamental aspect of enabling different email systems to allow certain information to be shred with others. We will examine those requirements below.

To facilitate the above there must be a transformation (mapping) of the source-specific nomenclature of information generated, to the group's terminology and definitions, as users have no knowledge of the former. One way is by providing meta-tags to the information.

ii. Unambiguous Representation

The awareness attributes must be stored and represented in a clear unambiguous manner to avoid any misinterpretation such that one can obtain the necessary status of various aspects of the project without any ambiguity and in a timely fashion. This is essential so that the correct attributes are queried and meaningful information is extracted about the group work. Since awareness attributes are collected from various sources there needs to be a mechanism to resolve any ambiguities that may arise. This could be facilitated by using some sort of project lexicon, which helps in clearing any ambiguity among attributes. This is essential for not only obtaining awareness but also enacting any workflow based on the values of these attributes.

iii. Ability to Tailor the Quality of Awareness

iii a) Dealing with Information Overload and Distraction

There must be mechanisms in place to ensure that enable users to establish the balance between being made aware of information that is absolutely essential to the user's work however at the same time not being subjected to **information overload**. The *Granularity Dimension* mentioned above addresses the question *"What should one be aware of?"* or *"How much should one be aware of"*. There must be mechanisms to display the information of the appropriate granularity as desired by the user. Depending on their roles in the project some participants may need to be aware of the overall project, and the direction it is taking. Others may need to know only specific details about certain participants, or artifacts. Thus the granularity of what they need to be aware of changes

depending on the project and the participant's role. Mechanisms of awareness must ensure such that participants can get the appropriate kinds of information and are not overloaded but at the same time do not miss relevant information. The user must not be unnecessarily distracted any more than essential to catch his attention of the information meant for him. So the recipient may wish to choose the manner in which he or she is informed of a certain information element. There are instances when the user must be made aware of the severity of a situation such as an emergency. In other words the degree of **obtrusiveness** of information delivery is important.

In order to be empowered to do the above, users must be able to access the awareness characteristics i.e. information about the awareness information. They can then make choices and employ filters when necessary. There must be means to express the Awareness Characteristics by which users can determine and tailor the quality factors using mechanisms such as filtration. Users must be empowered to selectively obtain not only the certain types of information but also tailor the amount of information that they are aware of. In order to do so the awareness information characteristics must be represented. Users will examine these characteristics and take decisions. The ability to select and filter awareness information should be based on the following criteria:

Type: Specify that I would like to be informed of the following types of information.
Form: Specify the form of information delivery, audio, text, image, icon etc.
Volume: Specify how much of the information I want to know.
Time and Frequency: Specify when I should be made aware of the information I have chosen and how often I should be kept informed. For instance if the user perceives that a chat session with multiple users will result in being informed of every line in the chat session, then he may decide to request a transcript or digest of the session from the participants later. This is also related to the minimizing unnecessary distraction.

iii b) Establishing Information Access Control

There must be an appropriate representation of the awareness information to enable one to clearly and easily establish the balance between awareness and privacy according to the needs of the group.

iii c) Enabling Enhanced Awareness

Awareness information representation and mechanisms must be so designed that they allow users to set preferences and whereby enhanced forms of awareness (such as the ones mentioned above) can be obtained. For example one may give privileges to a set of other users so they can be aware of all the activities that this user performs with regard to a certain artifact.

iv. Establishing Human-Agent Mixed Initiative

By integrating the information sources one can be aware of all the relevant awareness information that originate due to various aspects of the group's efforts. Users can voluntarily choose to be kept aware by selecting the information based on its characteristics. The user may also choose to employ intelligent software programs or agents and could subscribe to be notified if certain events occur (such as access to certain

artifacts, completion of certain meetings to view the minutes and so on). Users could have customized agent programs that monitor for such conditions and send these alerts or even periodically notify users of the latest status of certain elements of the project. Thus information can be "pushed" to the participants. Furthermore, depending on the needs of the project the agent programs could be sophisticated enough to take more complex actions on behalf of their users than just simple messaging. These actions would be based on rules created by users. The key issue would be the representation of the Awareness Attributes to empower such agents. Thus agents can provide context-sensitive awareness. Mechanisms to enable such agents to match the user's profile, and work context to available information are required. This can support enhanced forms of awareness such as "intersubjectivity" ("I know, that you know, that I know") among others [Anderson and Bouvin 2000a] [Anderson and Bouvin 2000b].

v. User Interface
Obtaining awareness would begin with searching and/or browsing for it. Users must be provided with a intuitive interface to interact with the awareness information. A complete picture of the available information choices should be provided for the users to select from.

Note that users will be working on different groupware systems and applications. Some of these applications may lend themselves to easy customization to display the additional awareness information that they receive from other heterogeneous systems. Also the host devices themselves vary from high-end workstations to handheld computers with limited display capabilities. Apart from the actual awareness information itself, there must be an interface to allow users work with the access control and enhanced awareness mechanisms that work in the background. In other words users should be able to obtain a complete picture or "map" of the awareness information flow in the group effort.

vi. Non-uniform access capabilities
Team members accessing awareness information do so using tools and groupware with varying capabilities of network connectivity, display resolution and other features. Some common examples of such tools are web browsers, handheld devices, and touch-screen displays. To ensure that they get the information tailored to the systems they use they may have to control the type and amount of information they choose to receive. Thus the ability to select the information based on its characteristics not only helps tailor the quality of information but helps in obtaining in a manner suitable to the individual.

vii. Obtaining Meta-Information
Awareness information as defined by the awareness attributes can include a variety of elements from the very artifact or content being changed to knowledge about the fact that an artifact is being changed. Following the previous requirement, the user's capabilities to receive awareness information may be limited. For example, one may not be able to participate in a video-conference session however one may be able to transmit text messages to the participants. In order to do so the user must know when the session will begin, who are the participants, how to reach them and so on. Thus the "meta-information" about the group effort must be available in some form. Since users'

capabilities to acquire awareness may not support all formats and exact artifacts, there must be means to obtain meta-information about it.

viii. Integration Process

The integration process to include new sources itself must be straightforward. A complicated process with would discourage the use of different tools and groupware. Also users may choose work without the awareness mechanisms just to avoid the overhead of integrating all the sources.

ix. Dealing with Change

Even with an existing project, sources of awareness information undergo change and this is reflected in the attributes describing them. For instance a newer version of a groupware system used by the team may have new functionality. Attributes describing the groupware system may now have to be modified to reflect the new functions used by the participants. There must be mechanisms that are in place to modify the STATE to reflect the change. As change is inherent in any group work [Bharadwaj et al. 2004] a straightforward process is essential for change management.

x. Extensibility

The STATE, which integrates the set of awareness attributes itself must be extensible since new attributes may be added and existing attributes may be modified. It should be able to accommodate new sources of information and integrate them seamlessly into the project context. For example, teams use new tools and groupware for specific tasks. Their interaction with these tools/groupware is an essential part of project awareness and attributes generated from these sources must be added to the existing set.

xi. Historical Awareness Support

Awareness of various aspects of group work that occurred in the past i.e. *historical awareness* [Nutter and Boldyreff 2003] is often essential and there must be means to retrieve such information. This is the conventional definition of *historical awareness*. We extend this definition to include the knowledge of not only what happened in the past in terms of the group's collaboration but also what was the awareness of the group in the past. In other words it is often useful to know *"Who was aware of such and such information?"* in the past. Very often we wish to trace why a certain decision was or was not taken, why some course of action was followed, why wasn't anyone aware of the impending problems and so on. Having a transcript of the state of awareness in the past can help us answer these questions. This transcript would keep snapshots of the awareness state in the past.

In essence the above requirements (for both physical and information integration) address the notion that members of a collaborating group need to be **appropriately aware** of the project **whenever they need** and **wherever they are.** Thus integration involves information transformation, relating different types and adding meta-information description to enable the above features. It is much more than creating databases, labeling and storing data. We propose a model as a basis for integration. This provides a common logical framework thereby decreasing the complexity in the integration process.

3.4.2 Addressing Awareness Propagation Requirements

The solutions devised to address awareness propagation in heterogeneous environments must address both the physical integration requirements and the information integration requirements. We introduce the concept of an **awareness framework** as a means to address both aspects.

The notion of an **awareness framework** has been inspired by past research in creating awareness systems that integrate various devices, hosts and applications. Most awareness systems research has been with respect to completely homogeneous systems i.e. awareness propagation within a single groupware system with its own client applications that convey the information appropriately. The systems we reviewed in Chapter 2 essentially try to propagate awareness in heterogeneous systems. Even then we see that most of those awareness systems have dealt with environments that are fairly homogeneous with a good understanding of the different kinds of source, the types and formats of the data they generate, the capability of the mediums that actually propagate the information as well as the capabilities of the client system. However there are some common elements that can be abstracted from all of them.

Close review of awareness systems in Chapter 2 reveal that awareness systems in essence have components that deal with the aspects of collection of awareness information, propagation of that information, and access to that information by other users. In addition there is a logical central entity that is analogous to the notion of the "state" and provides mechanisms to store the information and the meta-information required in an appropriate representational format so as to be accessible by the users. This research effort aims to distill these architectures and then extend them to create a **generic awareness framework architecture that** can be used to integrate a variety of sources. Most importantly the generic architecture proposed here is meant to empower the user with the ability to tailor the quality of awareness received. Addressing this and other requirements above are the motivating factors to the solution.

3.4.2.1 Awareness Frameworks for Awareness Propagation

Figure 3.6 below illustrates a high-level view of the **awareness framework** architecture**.** We see the same heterogeneous environment as illustrated in **Figure 3.6** above. We see distributed teams and individuals that use a mixture of tools and groupware to support their collaboration. This is in keeping with the philosophy of the awareness framework which seeks to act an as integrating umbrella framework over the existing environments. The components that make up the awareness framework are:
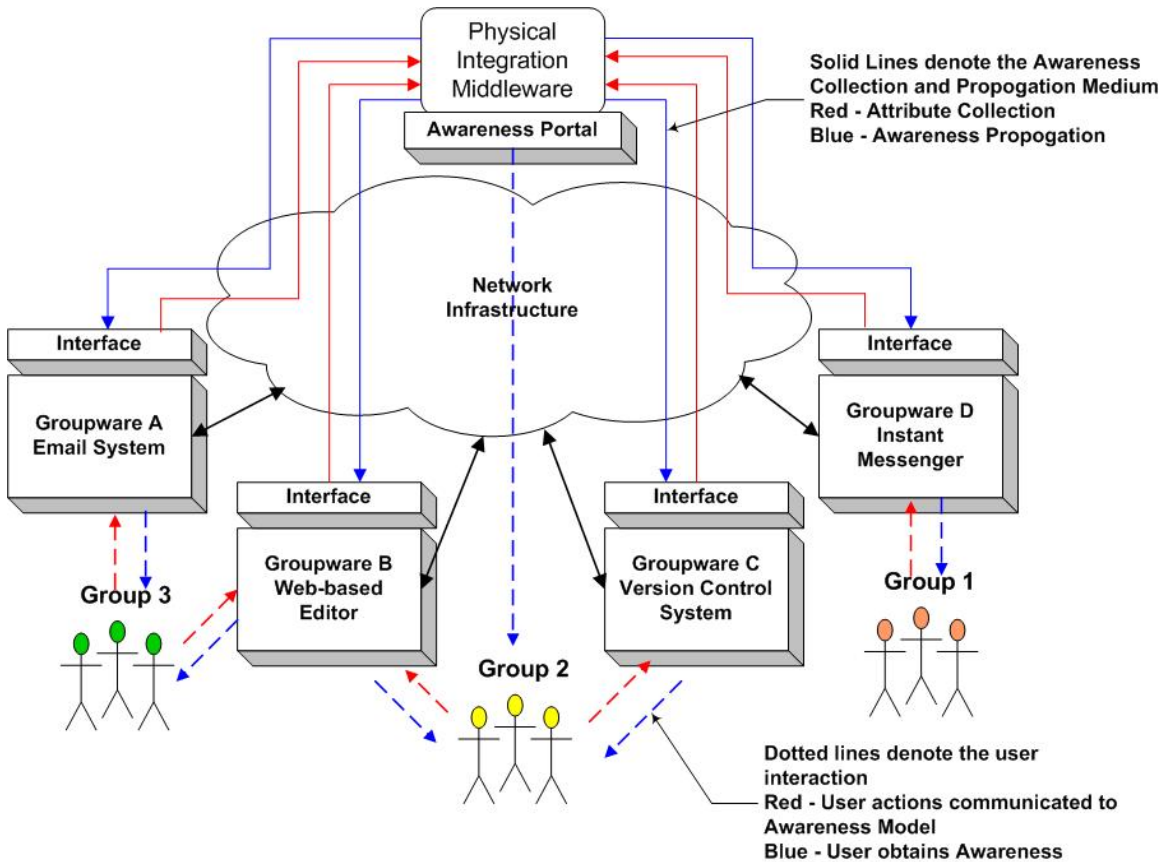
Figure 3.6 Awareness Framework

a) The Tool/Groupware Interfaces

In order to completely leverage the utility of various tools and groupware that are used in everyday group work, there must be mechanisms to integrate them so that team members can be aware of the project in spite of the diversity in tools and groupware. User interaction with the applications is an essential **awareness attribute** along with the others. Information about user activities such as creation and modification of artifacts, communication among team members (either in the form of email, chat or conferencing sessions) collaborative editing sessions and other activities need to be communicated to the rest of the awareness framework. Each application or groupware system used is a source of awareness information. Actors or users in the collaboration interact with these sources based on the needs of their work. The awareness attributes generated by these sources must be propagated to other actors via the awareness framework based on their needs. In order to do so the key component is the **interface** between the tool/groupware and the awareness framework. The Tool/Groupware Interface performs the task of collecting and communicating the user interaction to the rest of the awareness framework. Henceforth we will refer to this component as the **interface** for brevity. This interface determines what awareness attributes are shared. It determines when the attributes are shared. For instance the interface may transmit the information to the rest of the awareness framework when an event such as a user action. Or the interface may allow

itself to be polled by the awareness framework. Extending the example of an email systems that is part of an awareness framework, let us consider the case when the awareness framework has to be notified when user A sends an email to user B. The interface detects such an action and propagates the information about that action to the rest of the awareness framework. This frees the user A from having to CC and BCC the message to the members of the group that should also be receiving the email. Of course for this to work, user A may have to initially configure some options in the awareness framework that will make sure that the email information goes to only those users that he has chosen from amongst the group. Furthermore the user must indicate that this message is to be propagated to the awareness framework recipients based on the preferences he has set as opposed to the their messages that he may send (such as personal messages). This essential aspect is noteworthy. This is because **the same application can be used by a user in other spheres of activity**. For example, we use the same email application to send messages related to work as well as our personal messages. The next issue is if the interface will be at the email client or at the email systems server. For instance some email systems may allow information from its server to be shared in the awareness framework as opposed to allowing the client to send it directly. This way, the email system can exercise greater control on what is propagated and it also keeps the client thinner. The best example is a web-based email service where the client is just a browser and is available freely. These are some of the important choices. The interface may also allow the user to set other preferences with respect to the awareness framework such as configuring automatic notifications.

Furthermore the interface will determine how information from the rest pf the awareness framework is displayed in the tool/groupware client that the user is working on. This is because the very same applications and groupware systems that act as sources of awareness information, can also display the awareness information received from other sources. These are some of the key issues regarding the interface design. There are many technical details as to how to create interfaces for the variety of applications. As each source is different this is a very important aspect of the awareness framework research. The focus of this research is on the information integration and we assume that such interfaces can be built. Related work does show many efforts having successfully extended tools and groupware clients and integrating them.

b) Physical Integration Middleware

In order to integrate the variety of sources there is a requirement for an integrating middleware. The tool/groupware interfaces for each type of source application in the framework will be able to communicate using this middleware. This communication will be possible because the interfaces can conform to the specification of the of the middleware that allows the interoperation of awareness attribute information. Good examples of very successful middleware are the Common Object Request Broker Architecture or CORBA [Schmidt 2005] [OMG 2005] and the recent popularity of the Web Services Middleware for Service Oriented Architectures [Singh and Huhns 2005]**.** The middleware can be "light" where it communicates information of small volume i.e. information attributes that are essentially meta-information. The sources themselves are connected as shown by the network cloud in **Figure 3.6**. Actual content be it text, streaming video, images, voice can still be accessed using the existing network

connections. However the awareness attributes can be propagated to the rest of the awareness framework through the middleware. On the other hand the middleware itself can also be used to send the high volume content.

c) The Awareness Attribute Collection and Propagation Medium

In order to physically connect the awareness information sources (tools/groupware/sensors) to the rest of the awareness framework there needs to be a medium. Awareness attributes propagate over this medium. There is no need for a separate medium; the middleware that connect the various sources can utilize the existing network connection. The ubiquitous Internet and public telephone system are two examples of all pervasive medium that are part fop the network cloud. Adding to that is the rapid deployment of wireless technology including broadband wireless services. In fact this part of the awareness framework is the most developed and ready to use component among all. In **Figure 3.6** above, the arrows that connect all the information sources to the middleware represent the medium. The arrows pass through the cloud to indicate that the medium is very much part of the existing network infrastructure and there is not need for any dedicated connectivity. However the medium must have a very degree of availability and accessibility since it conveys essential information, which is the meta-information. Participants in a group project who may be collocated or distributed and even mobile depending on the organizational structure and needs of the project can easily communicate. The awareness model is the logical central entity that collects and maintains this information. The component that enables this communication is the "Awareness Attribute Collection and Propagation Medium". Henceforth we will refer to this component as the **Medium** for brevity.

3.4.2.2 Awareness Model for Awareness Propagation

The above components address the physical integration aspect of awareness propagation. However information integration requirements must also be met to make the awareness framework complete. How is the information integrated and managed? There is need for some logical central entity that does so. These requirements are met by the key component that is the *awareness model*. An awareness model is an entity that defines all aspects of the awareness propagation in certain group's collaboration. It defines the actors involved in the collaboration, the information sources, the characteristics of information generated by each source, the capabilities, of the sources in terms of information generated and ability to receive awareness information. In other words it defines the connectivity among the sources, and thereby establishes the relationship between the actors and what they can be aware of. It also provides the actors with a complete view of all information sources available along with the characteristics. Furthermore an awareness model defines the channels of information flow i.e. it allows the group to set access control on the information generated. A collaborating group will have roles assigned to its members and may also have restrictions on who is eligible to receive what information. These rules can be set in the awareness model. Also it allows users to obtain *awareness about awareness*, i.e. meta-awareness or enhanced awareness that was discussed earlier. These enhanced forms of awareness can also be enabled by a privilege mechanism similar to the one used for access control. For example, an

67

employee may find it extremely useful to know who else among his peers is aware of his activities. Another example is when an employee is not allowed to find out if his superiors are accessing information about his activities. This is a case where they need to monitor him without his knowledge.

Based on the above description the term "model" is appropriate for the logical central entity that defines all the above aspects of awareness propagation. Awareness models as seen in Chapter 2 have been used essentially in the virtual worlds to define the interaction and behavior of entities in such three dimensional spaces. Our use of the term awareness model is inspired by the same concept i.e. a means to define the propagation of awareness information among entities which influences their interaction. The awareness model addresses the issue of appropriate means to represent the awareness information and integrate it. It also provides a means to manage the information flow as well and is the key entity in the information integration process. The notion of awareness model is the primary focus of this dissertation and the next section describes the details of this component. **Figure 3.7** below illustrates the complete awareness framework with all of the above-mentioned components.
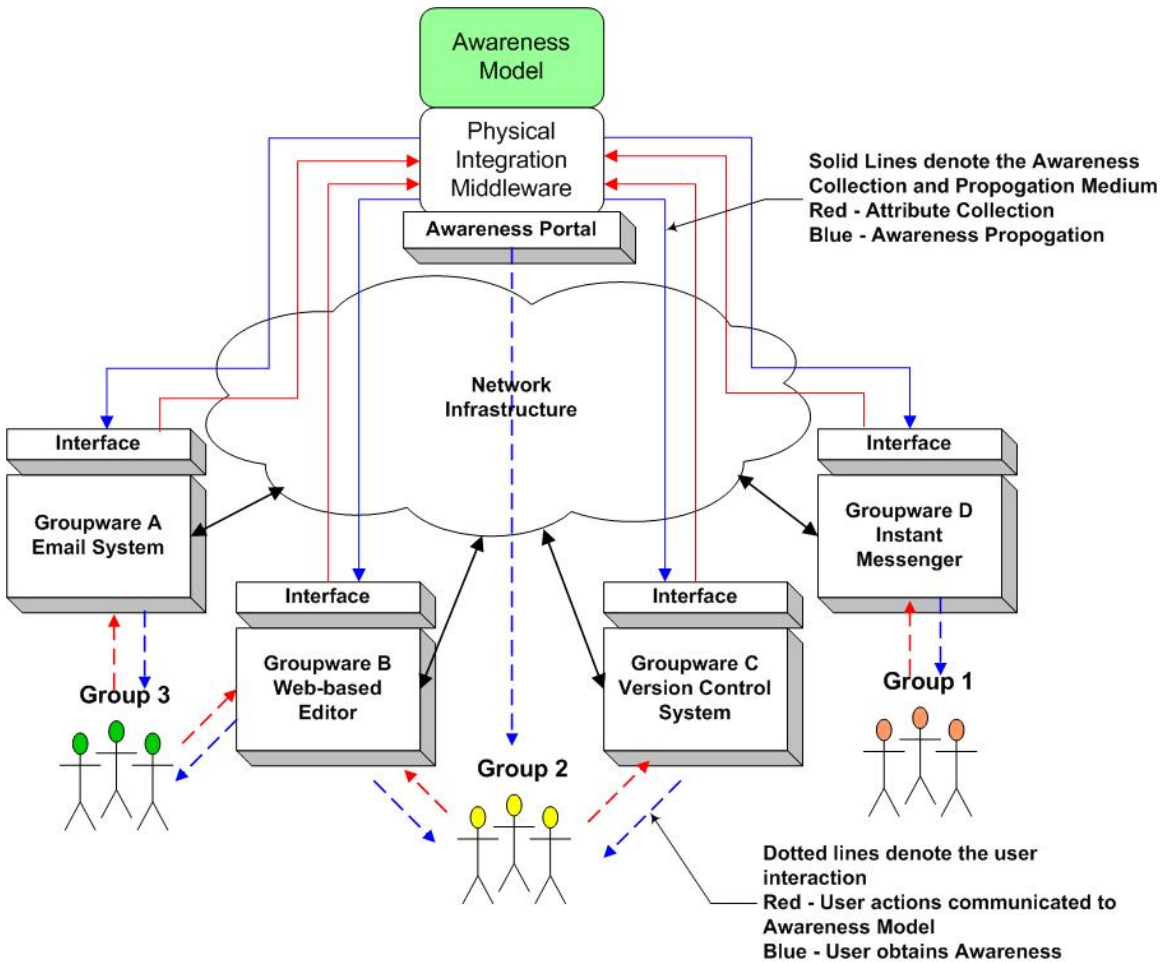
Figure 3.7 Complete Awareness Framework (with Awareness Model)

In **Figure 3.7** we observe a team using different groupware applications. The applications are integrated with the Awareness Model using the Awareness Attribute Collection and Propagation medium denoted by the solid lines. The individual groupware systems themselves communicate with the Awareness Model using the Tool/Groupware Interface (labeled as "Interface"). Users interact with the groupware and the dashed lines denote this interaction. Awareness Attribute collection i.e. information flowing into the Awareness Model is marked by red while Awareness Propagation from the Model is denoted by blue. For instance, user interactions with the groupware (red dashed lines) are communicated by the Interface to the Awareness Model as awareness attributes (red solid lines). Users can obtain awareness about the collaborative effort directly from the Awareness Model through a portal. However it would free the users from launching a separate application (such as a browser) to reach the portal if they could obtain the awareness information directly through the groupware applications themselves (blue dashed lines). The amount and type of awareness conveyed by the applications would of course depend on the capabilities of each type of application.

3.4.3 Security, Performance and other issues

From the discussion on the awareness framework in the previous section, it is clear that some crucial issues in the successful working of the awareness framework have not been touched upon. Prominent among them are the issues of **security** and **performance**. We will defer the discussion about these issues until the awareness model is described in detail along with an illustration and then examine the above issues closely.

3.5 Awareness Model and Awareness Map

This section describes in detail the Awareness Model component of the Awareness Framework. The motivation for the awareness model and a brief description of the model in terms of what it does for awareness propagation was made in the previous section. The primary goal of the awareness model is to address the information integration requirements that were presented in the previous section. The awareness model provides a logical framework for unifying various awareness information sources into a composite view. Information sources can be integrated in a straightforward manner. The model is adaptable to change and is extensible. It empowers users with the ability tailor the quality of awareness by providing information about the information sources. In essence it acts a very *enhanced directory* of awareness information sources.

The awareness model presented is based on a few concepts and constructs used in the related research with respect to awareness models in virtual worlds. The terms are defined here with different context as compared to the conventional definitions of awareness that applied to three dimensional virtual world entities. Three key terms used are ***Source, Medium*** and ***Focus***. We use the relationship between heterogeneity and quality of awareness described earlier as the basis for our definition of these concepts. Every application or information source is connected to the awareness model and user interaction with the information source is propagated to the awareness model through the application/tool/groupware client interface. *Actors* also referred to as *users* also need interact with the Awareness Model directly to find out all the available sources and the privileges they have as well as to modify their preferences. They can do so via any application that allows them to interact with the awareness model, or use a separate portal to the awareness model as shown in the previous section **Figure 3.7**.

Each concept related to the awareness model is described below in detail. The concepts are: *source*, *medium,* the *connectivity* among sources and media, the *information content* generated by the sources, the notion of *focus*, the *source superset*, *privilege mechanism* for access control, the *meta-information description*, *enhanced forms of awareness*, *notifications* as an essential awareness propagation mechanism and the *awareness map*.

Before each of these is examined in detail it is important to keep in mind the users or actors can interact with the awareness model through their view of the model. Each user's view of his/her awareness model is called the **awareness map.** In order to be truly useful applications must permit the users to view the awareness model and interact it with. Each awareness map is user specific and the details of a generic awareness map will be discussed in detail below. Briefly, it is the map of awareness information sources available along with all the privileges, preferences and meta-awareness that the user can view from the awareness model. The awareness map could be a graphical or a text-based depending on the display capabilities the user possesses. Through the awareness map, users can obtain meta-information about sources and the medium that are used to propagate information generated by the sources. They can also choose what sources to receive information from as well as tailor the quality of information being obtained by choosing different formats, frequency and volume of information being generated if the source provides such capabilities. Such interaction is possible due to the meta-information description about the sources. Users can also obtain enhanced forms of awareness and through the privilege mechanism privacy can be established. **Figure 3.8**

illustrates the awareness model concept. Meta-information associated with the focus and source concepts is indicated as a list on the illustration and is discussed in great detail in a subsequent section devoted to it.

The basic concepts and constructs of the generic awareness model are discussed first. The manner in which these concepts can be implemented together to form the generic awareness model features are discussed next along with how users can interact with such the model.

Figure 3.8 The Awareness Model Concept

3.5.1 Basic Concepts of the Awareness Model

3.5.1.1 Source

A *source* is any entity that provides information. A source is generally an application that generates information about various aspects of some user's interaction with it. This information includes the information content generated as a consequence of the interaction, such as a file that is created and written to. A source can also be a sensor that is capturing various aspects about the environment it is placed in and the information it generates are the values of these attributes such as temperature, pressure, humidity, or detection of movement. A source can be a camera or a microphone and the information it generates is the video and/or audio that is captured by it. Thus a source is any element that is capturing and propagating information about some user activities, locations, artifacts, or environment. In other words source is that entity that generates the various **awareness attributes** that were described at the beginning of this chapter. The type, and format as well as the volume and frequency of information generated by an information source will depend on the particular characteristics of that source as well as the external stimulus it receives. If users interact with an application very frequently then it is bound to generate information at the same frequency.

For example, an email service's server can be source of information that allows propagation of all email activity that occurs in an email account, i.e. if user U_i has an email account on the email service and if the email server is a source then it will propagate information about the email created by that account as well as email received by the account. The amount of information propagated depends on the particular source as well as the interface that connects the source to the Awareness Framework. A source may allows propagation of all the details of an event such as sending of the email including the actual contents of the email itself or it may just propagate the fact that an email was sent by the user account owned by U_i at a certain date and the time, to so and so. Thus only meta-information may be propagated.

The characteristics of the information generated and capabilities of a source are described by meta-information to the user who can choose if he/she wishes to receive information from this source. The manner in which users choose to receive information form certain sources and related details are discussed in the concepts that follow this notion of source. This meta-information could include details such as a unique identifier of the source, its name, a description, the address such as a URL or URI of the source, the type of information it generates, the amount (volume), the form, how often, how can the source be accessed and the preferred medium of the source i.e. what is the recommended mode of accessing the information it generates and the corresponding quality of delivery and many others. The meta-information fields will depend on the source itself. It is this information that allows users to pick and choose the sources they want and thereby determine the quality of awareness they will receive. The information content generated by the source can also have meta-information describing it. This aspect will be discussed in the next subsection.

3.5.1.2 Source Owners and Common Sources

Just as sources can be of various types, the relationship between a source and the user interacting with it can also be of the following types

Source Owner: A user or actor can be the sole owner of a source of information. An email account or an instant messaging account is a source and is owned by the account holder. Information generated by it can be attributed to the owner.

Common Sources: Sources can also have multiple owners, such as a common calendar, which everyone can modify, or electronic in-out board, which records who is in the office and out or an electronic message board. In this case the owner for the source could be "all" or a group with the members of the group specified. When someone receives awareness from those sources then he receives information due to the interaction of all the users who interacted with that source.

There are also sources, which may not have any owners such as a sensor that detects movement or temperature. Of course in this case one could designate that everyone owns the source. Thus the notion of ownership of source is greatly dependent on the type of the source.

The notions of who is the owner of a source is important when we see later on how enhanced forms of awareness features and privileges operate based on them.

3.5.1.3 Medium

The information generated by a source is propagated via some medium. The medium can be mixture of various types of networks (computer and telephone) with various capabilities (broadband, dial-up and others) and made up of various physical media (fiber, twisted pair) or be wireless. The characteristics of the medium dictate the quality of awareness information. Along with the description of the sources the awareness model will consist of description of the preferred medium that users should use to access the source. For example, we often see sites that stream information, providing users various choices of the same content but optimized for different bandwidths. Generally content for low bandwidth connections are of a poorer display quality. This will help the user determine how long it takes to access the information, as well as the quality of the content. Thus meta-information about the medium is essential to empowering the user.

3.5.1.4 Connectivity to other sources

By its very definition the awareness model consists of information that defines all the sources that are part of the model and how they can be accessed. In other words it provide a complete map of the connectivity of sources including the meta-information of the capabilities of those source and the media used. Thus the awareness model at the very basic sense consists of a connectivity model. However just being connected is no sufficient. We shall see that a privilege model explained below is also necessary to be super imposed on this connectivity model to make the awareness model truly useful.

3.5.1.5 Information Content Schema

The actual information generated by each source is termed as information content. As described above in the description of the source the actual content depends on the

source. For example considering the email account as a source of information, each email generated would be considered the actual information content. The information content is described by meta-information that described various characteristics about it. For example for the email generated, the meta-information would be a description of all the attributes of the email message such as from, to, subject, timestamp, attachments and so on. Thus the meta-information fields describe the information generated by the source. Users can make use of this information to decide the relevancy of the information to their current context, and future activities. In this way they can choose to obtain awareness from it or configure agents to monitor when certain information that is useful to them is created.

As discussed in the section describing the awareness framework, the awareness model is not meant to be a data bottleneck; rather it is analogous to a lens over the awareness capabilities of the group bringing into focus the pertinent aspects. Users may have direct lines of communication with the source for actual data transfer. This means that actual transfer of information content need not necessarily occur through the awareness model, and the middleware. Awareness information itself will be transferred through the framework (model, middleware, medium and the interface), however the information content can be transferred over the connected networking infrastructure. If the middleware used is able to transfer the information content then there it can do so. However there is no such requirement.

3.5.1.6 Focus

This top-level concept represents the focus (the "focus of attention") of an awareness consumer. The notion of focus on some object or entity implies that there is continuous awareness of it. Having focus mans that our attention is concentrated on that entity, analogous to a "spotlight" on an object. We use these concepts in defining our awareness model focus as a construct in the awareness model that allows a user to maintain a constant state of awareness of one or more information sources. A user can create a focus element and add sources to it. If the sources are active i.e. generating information then this awareness is propagated to the user provided the sources are in his /her focus. So one method of obtaining information from an active source is to add it to a focus element. Conversely by removing a source from a focus the user will cease to be aware of its information. This action is analogous to "shifting focus" or "tuning out" a source. Sources could be in a focus and not be active, which means they are not generating any information and consequently the user will not receive any information. From these other sources that the user could access but are not currently part of his Focus, the user can still be aware of their characteristics because of their meta-information description.

Apart from being a means to indicate which sources the user wishes to receive information or not, the focus serves another important purpose. Just as the conventional notion of focus provides concentrated view and often a context, the focus in the awareness model also provides a context. A Focus is a unified view of all its sources and the corresponding events and interactions occurring due to them (e.g. person's actions, communication, actions on an artifact). For instance the Focus could provide awareness about an activity in terms of email messages exchanged by the users (sources) or awareness about people entering a room, modifying an artifact and leaving assuming a camera was capturing the activity or the application used to change the artifact was

transmitting the information. In fact both sources are available the user could both the camera and the application as information sources to his focus and this focus would provide him with a unified view of the activity. Of course the user must have the capabilities to view the camera images and the appropriate client to receive notifications about the changes. However the focus as a construct helps him maintain a context of that activity. As time changes the user can maintain this context. This is very useful especially with heterogeneous sources we lose track of which sources are generating information about the same context. A focus as a construct helps maintain this unified view. Thus the Focus construct helps us create and maintain contexts of awareness related to our work as part of a group. Also just as our focus shifts over time, the Focus construct can change when sources are added and removed from it. The type and number of sources within a Focus can change.

There is one aspect about the focus construct in the awareness model that deviates from the analogy of the human focus we have used so far. Users can have ***multiple foci***. The notion of having multiple foci may seem contradictory to the act of focusing. However this is useful when one considers that the Focus concept is also used a means to maintain context of certain aspects of group work. Something humans do very well is to maintain multiple contexts and seamlessly switch contexts as necessary. We also use our knowledge from some aspect of our life in other areas, or more specifically we use our awareness of some context in other contexts. Often this is used to optimize our effort, allocate resources and gain some benefit, both in the short and long run. For example, if we have to go from our office to another location to get some work done, and if we know that certain tasks can be done along the way we use this knowledge to accomplish those tasks; this is in the short run. On the other hand I may have awareness that in the next month we should meet with our clients to resolve certain requirements and make a mental note that some of us need to meet with them. When a colleague informs me of his/her schedule for the next month and that he will be absent, I would try to use that information to immediately schedule the meetings sooner or find a replacement for the colleague or some other alternative. Thus proactive behavior is often triggered and aided because of our awareness of seemingly unrelated issues. The above examples are quite simple compared to situations in group collaboration where involvement of various people, tasks, information, schedules, resources and constraints make it difficult to maintain awareness easily of al aspects. Having multiple Foci in the same view can help alleviate the complexity to some degree. Users could have multiple foci, each focus providing information about different aspects of the group's work.

Similarly users could have the same source of information in multiple foci. Again if a focus is used to denote a common context then we often see that the same source of information could be useful in multiple contexts. Of course though the source is the same the information it generates at different times may be relevant to the different contexts that the user has defined.

The above description of a focus is an such that all the information from a source is received of that source is in a user's focus. Filters which can be configured any users receiving the information may be employed on each source to allow only certain information to be received.

3.5.1.7 Meta-Information Description

Meta-information attributes describing awareness characteristics qualify the focus, its sources and medium and are essential to determine and tailor quality. The actual set of meta-information that could be associated with each of the above elements will depend on the types of information source and mediums involved. An example of such a description will be presented at the end of this section.

3.5.1.8 Source Superset

For each user U_i the set of all information sources that the user can access as part of his or her view of the awareness model is called the *source superset* for that user. These include all the information sources that the user can add to a focus and continuously obtain information from as well as all the sources for which the user can obtain meta-information. It is named the *superset* since every source for that particular user is included. Each user's source superset is different and is defined by the policies of the collaborating group and the roles the user plays in that group. The privilege mechanism defines which sources can be added to a focus. For the rest only meta-information is available. For example user U_i may be able to view the meta-information about two email accounts belonging to other users in his source superset. The meta-information could include the name of the owner, the email id and so on. However he may have the privilege to add only one of the email accounts to his focus. This would mean that every email generated by that account would be accessible by U_i. Sources outside his source superset are not accessible and invisible to him. **Figure 3.9** below illustrates this concept. In our description of how the awareness model is validated, this concept has been implemented and is part of the graphical user interface to the awareness simulator (client user interface). Detailed description and more images of the same are provided Chapter 4. The source superset represents one level of and hierarchy. Finer levels could be enforced depending on the needs of the group. However the awareness model is capable of facilitating such levels or planes of awareness.

As mentioned earlier sources are owned by users (one or more). Thus users can also use obtain information about others users from the source superset. As part of the source superset a user (user A) must be able to find out the status of another user (user B) whose sources are in user A's superset. User A can also find out information about user B such as what are all the sources that B can be contacted. The current status of user B in terms of which source user B is currently active at (logged on or using), or which current sources he is receiving information at; for example: the user B is online on his Instant Messaging Client. The status of user B will be based on the sources that user B is using, and the information propagated by the source and its interface. For instance with sources like email accounts it is not only useful to know if the user is logged on but more useful to know actually when the user sends a message or accesses a message. This is true for email since one could be logged on and be doing nothing of interest. This is more typical of sources employing asynchronous communication On the other hand with sources that employ synchronous communication, the online status indicates that the user is active at that source. In other words one user (user A) can retrieve information about all the sources the other user (user B) is interacting with i.e. the sources that he or she is generating and receiving information from. To be able to find out such information, user must have privileges, and the sources owned by user B must be in user A's superset. One

can choose to add them to an existing focus or create new ones provide the first user has the appropriate privileges to add them to a focus. Upon ascertaining their status one can interact with them. The above discussion assumed that users could ascertain information about users whose sources that are in their superset only. However there may be sources a user owns that are not in the superset, in which case the requesting user may receive a "no information found" response. On the other hand the awareness model may permit very restricted meta-information to be accessed for all users along with their current status and similarly for all sources, very similar to a public telephone directory. This is more an issue of policy than that of design of the awareness model. However it is more likely that access to user and source information will be restricted. Ascertaining status of users can be manual when some user requests information or can be enabled through automatic notifications that are generated when users' change their status. Notifications will be discussed in detail below.

The above features are part of the notion that the awareness model is a directory of information sources. In fact querying to ascertain status and contacting them based on their availability is a very basic and essential functionality.
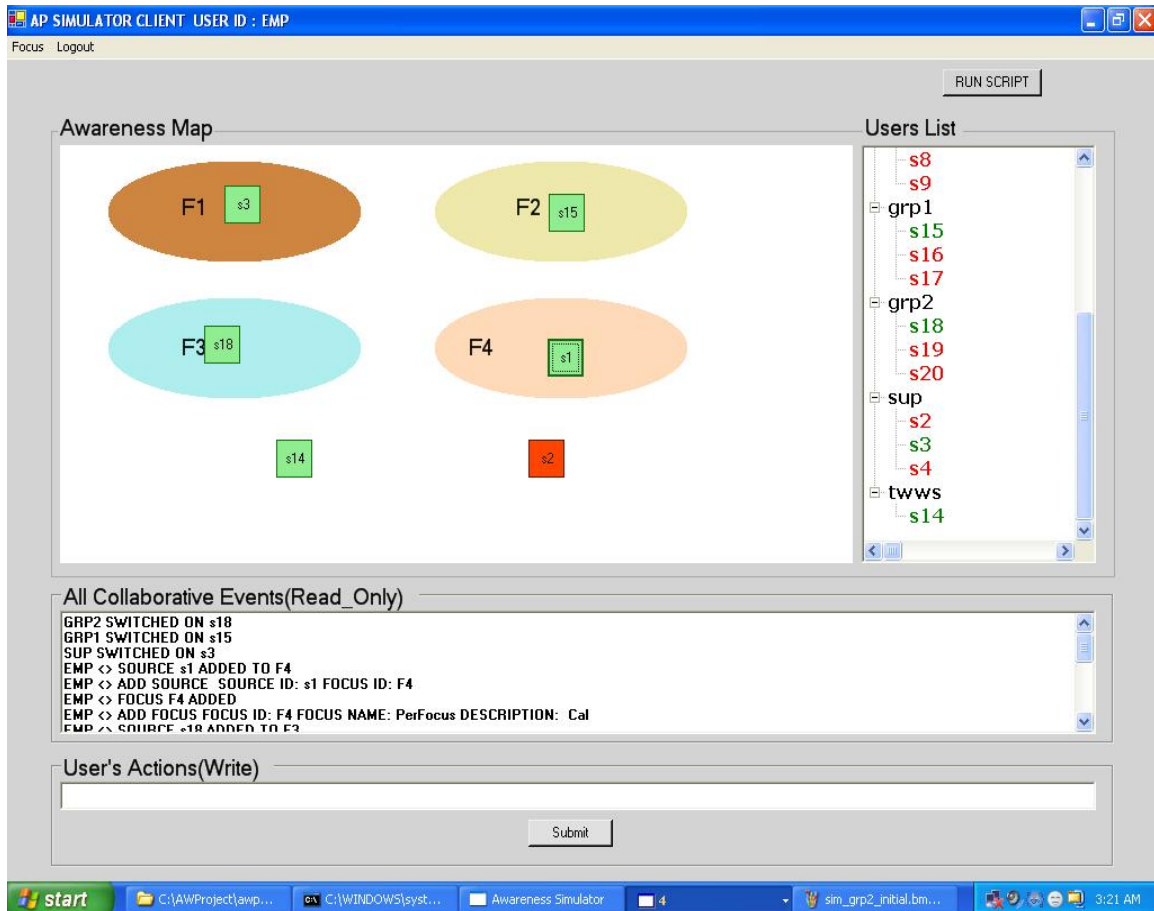
Figure 3.9 Awareness Map

3.5.1.9 Awareness Map

The user's view of all sources, media, and foci in his source superset with corresponding meta-information is his "Awareness Map", a concept inspired by [Gross, Wirsam and Graether 2003]. A simplified view of the awareness map notion is in **Figure 3.9**. As mentioned above the user's *source superset* is the set of all sources that the user has some form of access to. The user could obtain meta-information about some of the sources and or obtain information content generated by the sources. We could think of the source superset a directory of information sources for that user and could be displayed in the right most side of the awareness map. The sources from which the user can obtain information content form the left hand side panel of the awareness map. The user can interact with these two panels. This is one possible manner a graphical user interface could be designed for an awareness map. The source superset would display the list of all sources and through user could obtain the meta-information of each source by selecting it. A tree view could be used to display the source superset where parent nodes are users who own the sources and the child nodes are the sources they own. This is one possible representation of the source superset concept. This view can be called the "User-Source List Window" and indicate meta-information about the source such as its properties, which users are interacting with it and so on. Some types of sources have one and only one owner of the source, for example email accounts of individual users are owned by them. Information generated by the source can be attributed to the owner. Others can have multiple owners.

Users will be empowered with the ability to search and browse for various information source that wish to obtain information from. This is part of selecting the appropriate type of information source. If they have the privilege of obtaining the information then they could do so. Otherwise they may get only meta-information from those sources. All these would be part of the user's source superset. However the ability to search and browse may extend to other sources not in their source superset. In this case even limited meta-information may be provided and the users may have the ability to request the appropriate role that is in-charge that access to such information sources be allowed.

Users can interact with the awareness map to invoke various features. They can create foci, add and remove sources to foci as well as delete foci they are no longer using. Information sources can be represented by special icons that are representative of that particular type of information source. Foci could be shaped as circles or ellipse to indicate a spotlight effect. In the **Figure 3.9** foci are represented as ellipses and sources as square icons with the source identifier. Use of appropriate colors, changing sizes, blinking as well as effects such as relevant foci coming into view on the screen while others diminishing in size or moving out of the screen can be used to attract the users' attention without undue distraction. On selecting an element such as a source that is displayed in the map, one can zoom-in on the element to get all the meta-information. This information includes details about the element such as the source characteristics, medium characteristics and the characteristics of the content it generates. The meta-information associated with the awareness model elements will be described below. The map also provides an ability to ascertain awareness about awareness such as, who else is aware of what I am aware or? Who is aware of me? *intersubjectivity*, and so on. This ability can be very useful in various aspects of group work and typically in coordination

as well reduction of overhead involved in keeping everyone on the "same page". Users could use the map to subscribe for notifications they wish to receive based on events that can occur throughout the collaborative group process. If support for intelligent agents is provided by the awareness model, then users could use the same interface to configure such agents. These agents would aid in addition of context-sensitive awareness sources based on the user's profile of work. The superuser or corresponding role of the group that has privileges to manage the awareness model and it users could use the same awareness map but customized for the "administrator". This view would allow the superuser the capabilities to add, modify and delete users, sources of information, create access control policies limiting what users can access can be established with this view and so on. A super user would have unrestricted view of all users and their Source Supersets.

3.5.1.10 Customized Sources and Awareness Map

Users interact with applications. These applications (which are also sources) can be used to display the awareness map apart from their normal functionality, which is to provide the information content that they are designed to access. In this manner if the user switches between applications and devices he/she will always be able to access the awareness model in some form. The manner in which the awareness map is displayed of course would depend on the capabilities of the individual application and host. Note that to maintain context of one's collaborative activities and the information being obtained it is necessary to be able to obtain the same awareness map in all the sources that the user interacts it. However the user may wish to customize the awareness map in certain sources. For instance if I interact with a particular source (such as another email account's email client) I may wish to deliberately see certain sources of information in the awareness map it displays. In other words when I use this email client I wish to be kept aware of information from some sources. These sources may not be part of the awareness map displayed by my other applications. Thus each application could be customized to display its own awareness map. As mentioned above even with this feature it is always useful to be able to get a complete picture of all my awareness maps, the foci and corresponding sources in each of them to help keep the user aware of the overall awareness he is getting. This distinction of the same awareness map on all applications or individual maps on each application has some implication on enhanced forms of awareness that will be clear in the following sections.

3.5.1.11 Privileges

The connectivity aspect in the awareness model can be augmented with a privilege model that defines all the types of privileges each user possesses with respect to the awareness information sources. In this manner an access control mechanism can be established. For example the notion of source superset is a very coarse grained implementation of access control. Even within the superset we can envisage two kinds of privileges. From some sources the user can acquire information content generated by the source i.e. the user has what can be termed as "information content privilege", From the other sources, the user can only obtain meta-information such could be termed as "meta-information privilege". Continuing with the email example above; the user can view every email generated by one account while on the other hand the user can only get meta-information about the other account. Consequently he cannot get any information

generated by the account unless the owner of the email account specifically addresses a message for this user. The latter privilege viz. "meta-information privilege" is similar to looking up someone details and contact information in a directory or a profile. While the former gives you the ability to not only get their profile information but also access actual information generated. Having information privilege on a source implies that one has to have meta-information privilege too. Though some basic meta-information such as the id of the information source and connectivity details may be available, detailed meta-information that is available to other users may be deliberately withheld. For example there can be situation where one may be allowed o receive data from a source and asked to use the data to perform some tasks, however many details of the source such as its origin and the description of this data and what it pertains to could be withheld. The above privileges demonstrate an "all or nothing" option as far as information is concerned and is coarse-grained in its access control capability. Finer levels of control can be established where users are allowed to access information based on certain attributes of the information itself among other things. The objective of the research effort is not to propose 'the awareness model but to emphasize the necessity of 'an awareness model' that should possess such capabilities. Certain roles in the collaborative effort would have the ability to set privileges for all users. This is similar to the notion of a superuser in a system. Users may be allowed to set certain types of privileges too. The demarcation of the user-privilege would be based on the requirements and policies of the collaborative project.

3.5.1.12 Enhanced Forms of Awareness

The awareness model can be used by its users to find out various aspects of the state of their awareness as well as the awareness of other users. This we term as ***meta-awareness*** or enhanced awareness. In group work its is often useful to know the following with respect to the awareness level of other group members:

What I See is What You See (WISIWYS): The ability to find out who else is aware of the same information source(s) that I am aware of can be very useful. For example is someone should be aware of some information and is not it could be brought to his/her notice. On the other hand if we find out that information is being made aware to someone who should not be privy to it then we can take appropriate action. To ascertain if such propagation is taking place this ability to find out who else is aware of what I am aware of is necessary.

Second-Order Lookup: Let us consider user A is able to view source B in his awareness map. Source B belongs to user B. If user A wishes to find out what are the sources in user B's awareness map, then he can do so using the Second Order Lookup feature on user B. In other words this is similar to obtaining the awareness map of a user's source, which is in your awareness map. Hence the term Second Order. Such a feature may be extended to as many orders as desired. For example one may wish to view the awareness map of a user C whose source is in the awareness map of a user B and so on. If each source has a customized awareness map then the Second order would be done on a source. However if a user has the same awareness map on every source then this feature would be invoked on the user.

Reverse Look-up: This feature can be considered the inverse of the Second-Order Lookup feature. In this case a user A can find out who are the users that have his sources

in their foci i.e. who is aware of information being generated by the sources that he owns or interacts with. This feature would be invoked on the particular source

The awareness model is able to implement such forms of awareness very easily because it already possesses the information required to enable such features. The awareness model has the list of all the users, their sources, their awareness model views i.e. what are their foci and the corresponding sources in them. The privilege mechanism used to restrict access can be extended to enable the above features. To be able to use any of the above features, users must have the appropriate privileges. The awareness map would allow user to use the various features through the interface it provides. Thus the existing information in the awareness model and the privilege model can be used to facilitate the above features.

3.5.1.13 Historical Awareness Support

As described earlier our definition of historical awareness extends the conventional definition to mean the state of everyone's awareness in the past as opposed to just what occurred in the past. The awareness model can be used to record the series of actions that each user takes during his course of interaction with the awareness model. Users can be provided with the capability to search and browse through not only their past interactions with the awareness model but the interactions of other users (provided they have the appropriate privileges). Thus one can ascertain by finding out what foci and sources each user had in his/her awareness map in the past, their state of awareness. Of course the awareness model record or transcript can also help them they can find out what actions other users performed, what prompted them to do so (other actions that might have initiated such actions) and the consequences. This is more in keeping with the conventional definition of awareness.

3.5.1.14 Search and Browsing Information Content

Apart from the ability to search and browse the awareness model for relevant information sources and select them to add to a focus, users can also search for information content generated by a source. This search /browse capability could be provided by the awareness model. Users can select a source and then search and or browse the information content generated by it. For instance one can search through various folders of an email account for messages received, messages sent and so on. This is possible if the application that displays the awareness model has the capability to display the information content, also the source and its corresponding interface itself must allow such searching/browsing and retrieval. The issue of security in accessing this information content cannot be overlooked and any such access must conform to the security policy of the information source. Again the interfaces to the awareness framework determine if such access is possible and if so how.

3.5.1.15 Notifications

Various forms of notifications can be provided to users. Notifications could be about when certain users access the awareness model. If user A adds to his/her focus an information source that user B is interacting with, then user B may be notified of that addition. Similarly when someone removes a source from his/her focus the user

interacting with that source could be notified. Notifications could be generated when a certain user logs onto a source or logs off from the source (such as the notification sent by MSN or Yahoo Instant Messenger to the other members of a chat group). Another example is when certain users who are logged onto a certain source perform some action. This is commonly experienced in email clients where it is not only beneficial to know that someone is logged onto their email account but to actually be notified when they send a message or access a message. Thus the notification could be based on the user actions in addition to their status online. The types of notification sent as well as the cause for that notification again depends on the types of information sources used in the collaborative effort as well as the support that the interfaces provide for propagating such information. Depending on the interfaces and sources notifications could be based on certain specific user actions that other users may be interested in being aware of. A user may be notified if someone is performing a second-order lookup or reverse lookup on his sources. Once again the privilege mechanism could be used to permit or restrict the notification abilities that users can avail of. Thus the abilities depend on the sources and the needs of the collaborative effort.

3.5.1.16 Default Application Behavior

An extremely important aspect is that information propagation need not follow the awareness model. Users may wish to send and receive information to other users according to their personal preferences and needs. The users may be involved in the collaboration or may be external. For instance user A may want to send an email to some user B but privately and cannot do so since upon using the source (his email account) the message get sent to others as well (due to the awareness model settings where his email account source is in the foci of users he does not intend to send the message to). Note that a user's email account may be his/her personal account, which is also used in the group collaboration. Even though the user may use the same source (application, host) in the collaboration not all information generated may be for the group. A user should able to communicate with another user even if this user's information source is not in that user's focus. This behavior is natural since there may be situations where one has to communicate with others and if the awareness model settings prohibit the communication then it may not be possible to do so. In this way one can communicate with someone even if source they use is not in the intended recipient's focus. The recipient would receive this message. Also this action being completely independent of the awareness model would not follow its settings. Thus even if the recipient has chosen not to receive the message by not placing this source in his focus he would still receive it. It is up to the recipient to make ensure that such messages do not come to him if he chooses not to receive them. The Awareness Model should not take over the user's source but only take information meant for it.

For such selective propagation two approaches can be followed:
1. The first is to empower the user to customize the source itself and only allow certain information to reach the awareness model. Once the information reaches the awareness model it follows the privileges set. For this approach to work mechanisms can be employed at the application-interface to the awareness framework and awareness model.

User-based filtering of email messages is an example. Though generally applied to incoming messages the same could be applied to outgoing information. An example would be: If the Microsoft Hotmail server has a filter allowing only messages with certain attributes to go to the awareness model, then a user email account whose message matches those attributes such as a particular subject line would propagate to the awareness model. Others would not.

2. The second approach would be to establish more fine-grained access control and privilege mechanism in the awareness model itself. Using this mechanism all the information generated by a source gets propagated to the awareness model but only the selection and filtration gets done at the awareness model level.

3. A third would be a hybrid of the above two approaches where the information is routed based on settings at both levels. This is an example where both the user and the awareness model act simultaneously and could be termed as a "mixed-initiative". As mentioned before adjusting the boundary between the two is an interesting research question. In any case, the application interface, medium and security policies of the sources as well as the organization(s) hosting and using the Awareness Model would greatly dictate the final shape of the solution.

A related issue in either approach is who would be allowed to set the privileges. The users themselves or someone would dictate the privilege setting. Again one possible hybrid approach could be followed with users having certain privileges on the source side while the privileges on the Awareness Model side could be set by the group or organization and that would be based on the requirements of the project and its policies.

3.5.1.17 Use of Intelligent Agents

So far our discussion of the awareness model has been with respect to human actors (users) manually interacting with the awareness model. Human interaction can be augmented with intelligent software agents who can act as monitors of awareness on behalf of their human users. In one sense the notification mechanism is an example of an automated agent exhibiting very simple intelligence. Based on the user preference the notifications are sent to the user when certain events occur with respect to the awareness model. However agents can be configured to perform more complex tasks on behalf of the human user. Agents could match ongoing events such as information being generated from certain sources, with the ongoing activities of the user on whose behalf they are acting and suggest to their human actor that he/she add those information sources to his/her foci. Other suggestions could also be made based on other users activities in terms of addition of sources to their foci, other users online status and so on. Extending the notion of focus that is already employed in the awareness model, intelligent agents may be automatically configured to "bring into the user's focus of attention" some information that it believed to be of relevance while "taking out of focus" the other information sources. In terms of an awareness map this could be graphically displayed as foci symbols getting larger in size, information sources being highlighted, changing color or blinking to gain the human users attention. Other information source could diminish in size This implies that the agents should not only be aware of the activities and events with respect to other users but they must be aware of the their owner's sate of awareness (his foci and sources) as well as his current activities. The challenge is then to effectively capture represent a users context in terms of not only his preferences but also activities in

a manner that intelligent agents can make use of them. To a certain extent the notion of awareness attributes and the meat-information description can aid such behavior. A larger research issue is to effectively establish a human-agent continuum where the either human and or agent can work together with no conflict among the behaviors. For instance a simple example is when the human can override the agent's choice or behavior. Establishing this human-agent *mixed-initiative* effectively to enhance the collaboration is a research challenge [Reddy, Selliah, Bharadwaj, Yu, Reddy S. and Kankanahalli 2004a].

3.5.2 Generic Awareness Model Features and User Interaction

This section brings together the above concepts to demonstrate the features that are desirable in a generic awareness model. The manner in which users can interact and benefit from these features is discussed below. It is intended to be a summary of all the awareness model capabilities described so far.

1. The awareness model provides mechanisms to define the connectivity of all the different information sources used in a collaborative effort. It defines the users their roles and their relationship to the different information source in the collaborative effort as part of the awareness model.

2. The awareness model also provides meta-information about all the sources as well as their connectivity capabilities (medium of information transfer). This meta-information describes the information characteristics, which help users select the appropriate information sources based on their needs as well as the capabilities they have to access the information. Thus they can tailor the quality of awareness they receive. This meta-information is termed as awareness information and can be realized ion terms of awareness attributes. The ability to select various information sources can be through searching and or browsing the awareness model for all the information sources.

3. The information sources are of various types and can be owned by one user or multiple users. Information generated by the source is a consequence of some interaction of users with that source. This information content generated can be accessed by users employing applications that are capable of accessing and displaying such content. The users are interacting with these applications which are in turn themselves sources of information for other users. Note that the information content is different from the awareness information which is the meta-information about the content. However information content can also be considered as part of the awareness information set (awareness attributes).

4. The information that propagates to the awareness model and the rest of the awareness framework depends on the type of sources as well as the interfaces that the sources implement to connect to the awareness framework. The design and capabilities of these interfaces is crucial in determining exactly what type awareness information is propagated as well as how often.

5. Information content can be accessed directly from the sources while the awareness information (meta-information about the content) is transferred through the awareness framework. However the awareness middleware may also be used to transfer the information content provided it is capable of doing so.

6. Users can interact with the awareness model through their view of the model viz. the awareness map. The ability to ascertain what information sources are available and then select information sources based on their meta-information attributes is possible through the awareness map. Searching and browsing for information sources can be provided as part of the awareness map. Various other features such as obtaining enhanced awareness, historical awareness and ability to configure preferences for notifications, and setting privileges are also accessible through the awareness map. Through the user-source list directory users can ascertain the status (online, offline, busy, and others) of other users and their applications (sources). The awareness map can also be used by a superuser or other appropriate role that manages the entire awareness model for all users. Actions such as addition of users, sources, and their management can be facilitated through the awareness map. The applications used in the collaboration process (which are also sources themselves) can display the awareness map in some form. Users can maintain the context of all their state of awareness in when the same awareness map information is available in all the applications they use. They can also customize the individual awareness maps in each application to display certain information sources they choose and still be able to obtain awareness of all their current contexts of work and the associated awareness sources.

7. Apart from the connectivity model a privilege model can be used to restrict users' access to creation information sources. One coarse-grained access mechanism is to define for each user a set of all sources that the user can access called the source superset. This source superset consists of all the sources that the user can access information content from as well as those from which the user can obtain meta-information. The meta-information of the sources creates a personal directory of information sources. Privileges can be used to create finer-grained information access control by restricting access based on various attributes of the information source and content. The ability to set privileges itself can be considered as a privilege. Users who are given the rights to set certain types of privileges may do so.

8. A focus is a construct that can be used to denote a set of information sources from which the user can continuously obtain information form. If a source is in some user's focus then all information generated by the focus can be received by the user. In other words the focus is a mechanism to subscribe to information from a collection of sources for an extended period of time. As the sources generate information the user is notified of the information being generated. The user can access the actual information content using the appropriate application. Users can create foci to denote the context of work they are involved in. Users can create multiple foci each with multiple sources. The same source may also be part of more than one focus. These are

useful for managing different contexts of activities and the associated awareness that the user has with respect to them.

9. Users can obtain the various forms of meta-awareness by invoking features such as *"What I See Is What You See"*, *"Second-Order Lookup"* and *"Reverse Lookup"* which were described previously.

10. The awareness model itself can be searched and or browsed to reveal information about various interactions that the user may have committed in the past. Furthermore users may also have the privileges to search and or browse actions by other users. Most importantly this ability can provide *Historical Awareness* of not only various aspects in the past but also the state of awareness of various users in the past.

11. Apart from searching the awareness model for information source to select, users may be allowed to search a source for the information content it has generated. This provided the source and its corresponding interface can provide the awareness model the capability of acting as an information content retrieval portal.

12. Use can configure and obtain notifications from the awareness model based on various events that occur such as actions by other users as well as when certain types of information are generated. The types of notifications that can be generated depend on the types of sources and the interfaces.

13. Applications (information sources) with respect to the collaborative effort and the awareness model can also be used independently of the awareness model. Users can continue to interact with applications without having information about the interaction and the content generated be propagated to the awareness framework. This behavior would be as if there was no awareness framework present. The issues lie in deciding how this would be implemented, viz. at the awareness model level or if the users would be given the ability to choose what information gets propagated. A hybrid solution seems the most useful for a majority of collaborative efforts.

14. Apart from just human actor interacting with the awareness model, intelligent software agents can also act on behalf of the users to perform various tasks with respect to the awareness model.

3.5.3 Meta-Information Description Illustration

An example of a set of meta-information attributes that qualify some of the key awareness model constructs is presented below. The attributes listed below illustrate how users can benefit from some of the generic awareness model features listed before. The set of attributes required for each model would greatly depend on the collaborative effort. **Figure 3.8** illustrated the meta-information that could be associated with a focus and source.

i. Focus meta-information:
description: About the focus i.e. what awareness is this focus providing. The user who creates the focus can define this attribute.
focus_identifier: A unique identifier as there could be multiple foci. This is also user-defined.
start_time and *end_time:* Indicate the duration for which the focus was active. These attributes are particularly useful when one wants to ascertain how long a certain user was aware of certain asoects of the collaboration. This is how historical awareness may be obtained.
Source & Medium List: A list of sources and corresponding preferred medium (if specified) that are present in the in the focus for example: *(source$_1$, medium$_1$), (source$_2$, medium$_2$),…(source$_N$, medium$_N$)*.

ii. Source meta-information:
In addition to its own *Description* and *ID* fields, each source entity has:
start_time: The time the source comes within the current Focus.
end_time: The time the focus excludes it. These fields can be queried to ascertain to find out how long this source was part of someone's awareness. Used in obtaining historical awareness.
Awareness Map Foci List: Each source element may have multiple foci it is obtaining awareness from such as (Focus$_1$, Focus$_2$…. Focus$_N$). This is possible when you consider that any application is a source. When a user interacts with that application it will display a view of the awareness model to the user. The user may choose to configure the awareness map displayed on this source in a particular way that is different from his awareness map on other sources. Thus each source would have its own list of foci. Thus such as customization could be implemented. This concept will be discussed further in the features section below.
Information Content: This describes the information generated by the Source. Its attributes are:
Type: About the Information (natural language and/or keywords).
Form: Text, Audio, Video stream and others.
Frequency: How often is the source generating the information. Could be in the form of discrete notifications or continuous stream.
Total Volume: How much has been generated so far (or recorded) or what is the total size of the information content that can be accessed.
Content: The application users are interacting with will display the information content that is designed for them, for example and email client can easily display the email that a user receives. However there may be client applications that display the awareness map

that can also allow users to obtain information designed for other clients. For example, is user A is using an application which displays the awareness map, and user A selects a a particular source in that map and wishes to view the actual content generated by it he may be able to do so provided the application he is using has the capabilities. This capability depends on the application as well as the interfaces to the awareness model. How would the current application know what exactly to display, this could be done by providing the user the ability to search and brose the source's information content. Just as one can search through the contents of our email "inbox" to retrieve the messages that are tired in our email account, the email account being a source. Depending on the capabilities of the application that is displaying the awareness The actual information being generated, according to a content-specific schema depending on the source. For example, actions taken by sources, actions on artifacts, video streams, email messages and chat sessions would all be content.

ii. Medium meta-information:
Corresponding to each Source is its Medium. In addition to its own *description* and *identifier* fields, each Medium entity has:
Medium Specific Characteristics: A set of attributes about the specific medium. For example, the network characteristics for a wired or wireless computer network, cellular phone network or closed circuit television network would be of interest since they would indicate how much of information could be delivered and in what manner.

The previous section describe the overall concepts involves in the awareness model. The noteworthy aspect is that there is no one single awareness model that fits all collaborative efforts. The aim of this dissertation is not to propose a single awareness model as the solution but introduce the notion that an awareness model s necessary for awareness propagation and to define the characteristics such an entity.

The actual architecture of an awareness model may vary depending on the needs of each collaborative effort. Furthermore the awareness model could be implemented using different methodologies and technologies. For instance one could us e a Relational Database Management System or use a hierarchical structure such as a tree to define the model and implement it using XML. The latter could be stored as an ASCII XML file. The RDBMS method has been used in the examples explored in this research.

To acquire a better understanding of the details of an awareness model we examine a realistic scenario where awareness propagation is necessary in a heterogeneous collaboration environment in the next section. This is one possible design of the awareness model for this particular scenario. The description will include how this design addresses some of the key requirements. The corresponding implementation and assumptions made will also be discussed.

3.6 Awareness Model in a Collaboration Scenario

We choose typical activities that occur as part of collaboration in office environments as a candidate scenario where the awareness model can be effective. Office environments demonstrate various aspects desirable for our study. Individuals have different roles and are engaged in various tasks. Some individuals have multiple roles and have to switch between them as required. Groups are formed depending on the needs of work. Individuals may belong to more than one work group. Users may be mobile. A variety of applications can be used. Even within the same organization people may choose to use different applications based on the requirements of their work, capabilities, availability and their preference. Furthermore individuals and groups within one organization may have to work with groups across organizations. This collaboration may be short-term or for an extended period of time. An office setting in an organization has all elements that contribute to the heterogeneity that we are faced with. Domains such as healthcare, military, and design and engineering to name a few could have been chosen. However it is often seen that in such domains nearly all individuals and groups tend to use the same set of applications creating homogeneous work environments. The same groupware is often used. Very often the groupware is specialized for the domain enabling certain specific functionality. Consequently everyone is empowered with the same type and level of awareness that the homogenous environment propagates.

It can be argued that groupware is employed even in office settings. Though true, it is often seen that groups using different groupware need to collaborate. This can be either within a large organization or across organizations. Thus the awareness model addresses the larges issue of awareness propagation with different groupware systems and individual applications.

3.6.1 Office Collaboration Scenario

Figure 3.10 illustrates the Office Collaboration scenario. The actors involved are a **supervisor** and **employee** co-located in the main office. They are equipped with workstations and have broadband access to the Internet as well as their own office Intranet. Additionally they may be connected to any dedicated networks with other organizations as required. Both of them use email, instant messenger and have access to telephones for their communication. Both are engaged in various activities. We focus on the employee. The employee is responsible with having to participate in a document review process with a group, which is located remotely. We name this document review group as **Group 1**. The employee is also concurrently engaged in a collaborative editing task with another remote group viz. **Group 2**. For the purpose of this scenario we represent all the members of each group apart from the employee as one entity, the group itself. In reality every member of the group could be located remotely and we could include each individual actor in our scenario.

The employee and Group1 collaborate using a Video Conferencing Infrastructure (VCI). This is a traditional video conferencing system where each user connects to a server using a dedicated client over ISDN lines. Every user can view images and listen to the audio from every other user. A distributed document review can be conducted using such a system since it provides the same benefits of a face-to-face meeting, which is often

the means for a document review. All the elements of the video conferencing system are abstracted as one entity we term as the infrastructure since this level of abstraction is sufficient. Based on the functionality provided by the VCI users may also exchange text messages, and have "sidebar conversations". They may share documents and make presentations, take e-polls and so on. Microsoft's LiveMeeting, a web-based conferencing system provides such capabilities. We assume our VCI is basic with live images, audio and text messaging. The document review process itself is fairly simple. It is assumed that the members of the group have received the document to be jointly reviewed and have joined the meeting with their recommendations. The actual process involves going through each page, with each member making their suggestions, any negotiation necessary takes place to resolve conflicts. Upon everyone's acceptance the recommendations are recorded by one member (the editor) and the group moves onto the next page.

In the collaborative editing process, each use connects to the Collaborative Editing Infrastructure (CEI) that consists of a server that manages the concurrent editing process. Users can see the cursors of other editors on their collaborative editing client. They can also exchange text messages in a chat-room like user interface to discuss their work. The employee is connected to both these groups using dedicated client applications over broadband networks. Thus he receives the best possible quality images and audio.
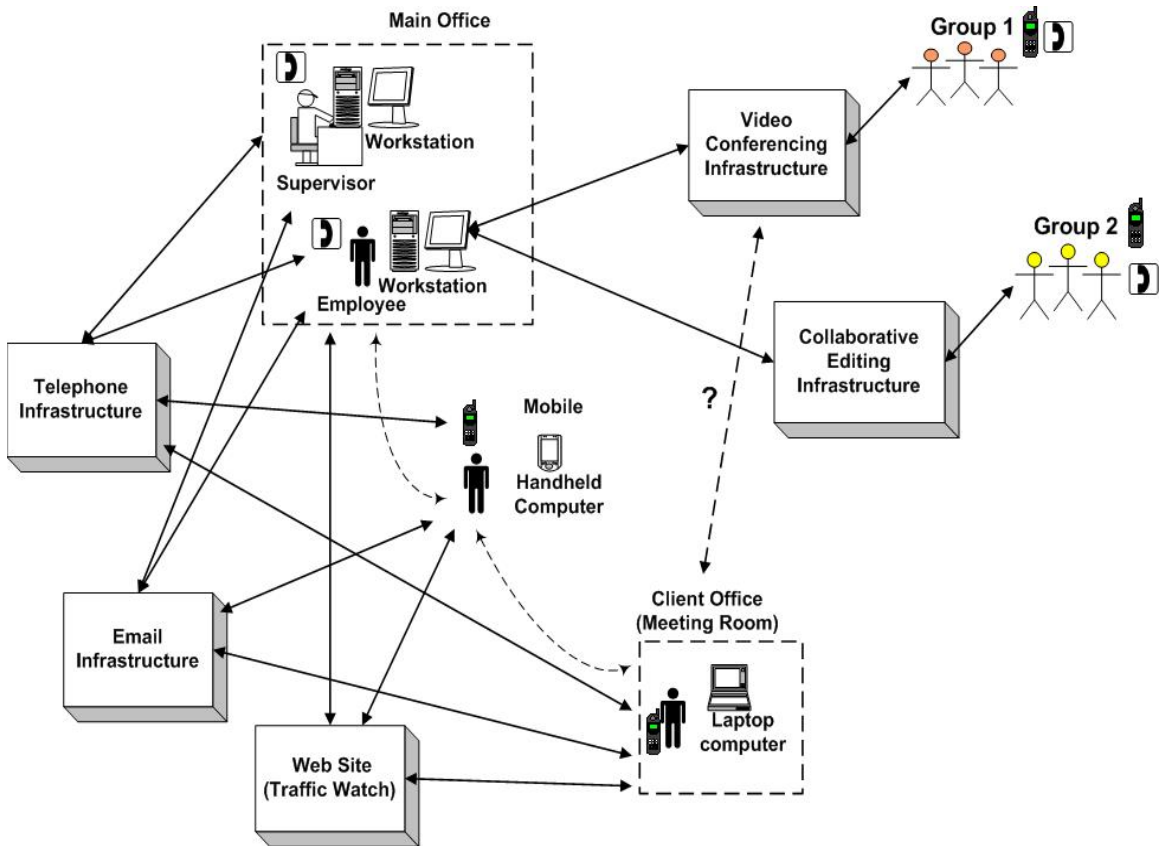
Figure 3.10 Office Collaboration

Let us assume that the employee has unavoidable prior commitment such as attending a meeting at another client's office. This implies that the employee has to be away from his workstation and at his client's office. He also has to spend some time traveling between the two sites. This means that he can no longer access Group1 and Group 2's information as he could in the main office. Thus the heterogeneous environment impedes his ability to maintain awareness of the current status of his work. Even though he may not be able to contribute to the groups' entirely his awareness of their work can help him make suggestions via email messages, instant messages or even telephone calls. While mobile the employee has to be away he can access his hand-held computer (PDA) and a laptop computer. To maintain awareness he must be able to connect to the either group's VCI and CEI servers. However his ability to access the VCI and CEI depend on various factors. Both the servers must support access from wireless mobile clients. As hand-held devices are used the applications must support appropriate interface that can make interaction user-friendly while supporting all the required functionality. Most importantly, wireless network access must be available to connect to those servers. As is often observed even in meetings people often use their laptops to record information from the meeting. However they also exchange emails, and chat with others using various applications. While attending the meeting he employee may choose to use his laptop to maintain awareness of the groups' current status even if he cannot see the actual documents being reviewed or edited. This is an example of meta-information being used when the complete information id not available. In this case as before, the employee must have network access to be able to connect to the VCI and CEI. Though network access itself may not be the issue as the employee is engaged in another concurrent activity, he cannot interact, as he would do so at his office workstation. Thus he must be able to maintain awareness by casually gazing at his screen. His image cannot be viewed and his responses also cannot be audible, they may at most be infrequent short text-messages. Thus connectivity is not such an issue, but the ability to interact is constrained by the work situation he is in. The key aspect here is that his collaborators in both Group 1 and Group2 would greatly benefit from knowing that he is aware of their activities but at the same time they are aware of his limitations as far as his ability to interact. This knowledge shapes their interaction and work processes. For instance they may send him more messages knowing he can only read those as opposed to viewing the entire document since that would be too distracting. They may hold off on some decisions and come back to it once he is able to interact fully and so on. **This heightened level of mutual awareness of each other's context is what the awareness model aims to provide.** All the above is possible provided he can access the public network using his hand-held and/or laptop hosts and the VCI and CEI permit such access via public networks. The ability to be connected is a key requirement for any awareness propagation and we examine an interesting aspect of the state of connectivity and some types of applications in the following paragraph.

As seen in **Figure 3.10** there are some infrastructures that we can maintain excellent connectivity irrespective of where we are and what hosts we use. This is due to the ubiquitous nature of the particular infrastructure. The best examples of such infrastructures are the telephone network (landlines and cellular); email servers (web-based email) and websites from we acquire information (such as a traffic watch website that the employee uses to alert him of any delays on his often-traveled routes). The latter

two are due to the Internet's ubiquity. The employee can use these services irrespective of host device capabilities (hand-held device, laptop computer or office workstation). The multiple lines from each host to each infrastructure denote this. Apart from minor differences in the amount of information and quality of communication due to the actual device's screen and network connectivity the user's ability to work is not greatly affected. On the other hand his connectivity and accessibility to the video conferencing and collaborative editing infrastructures is not assured and depends on the host device, the networks, the software client capabilities as well as the server systems. The dashed line with the question mark denotes this. Thus there is a gap in the awareness propagation with regard to these applications. If the aim is propagate not only awareness of activities but a heightened level of mutual awareness of context such as mentioned above, there needs to be a means to bridge the gap that is seen as a result of the heterogeneity. Users must be able to ascertain not only what types of information sources are available but their capabilities as well. This will allow them to connect to those sources and tailor the quality of information they wish to receive from those sources.

3.6.2 Office Collaboration with the Awareness Model

The awareness model seeks to bridge this gap. **Figure 3.11** illustrates the notion of connecting all these applications together using an awareness framework. The awareness model is the central entity that integrates the information from these sources. To make this possible the applications must be able to connect to the awareness framework via interfaces. These interfaces determine the amount of information that can be propagated to the awareness framework. The awareness model itself acts as the directory for all these information sources. Users can ascertain which sources are available, the type of information they provide and the characteristics of the information (type, form, volume and frequency). This allows users to choose information sources to receive information from. It also allows them to be aware of features such as "*Who else is aware of what I am aware of?*" "*Who else is aware of what I am doing?*" " *Can I be aware of what he/she is aware of?*" and so on. These features are a result of the privileges that the Awareness Model provides for its users.
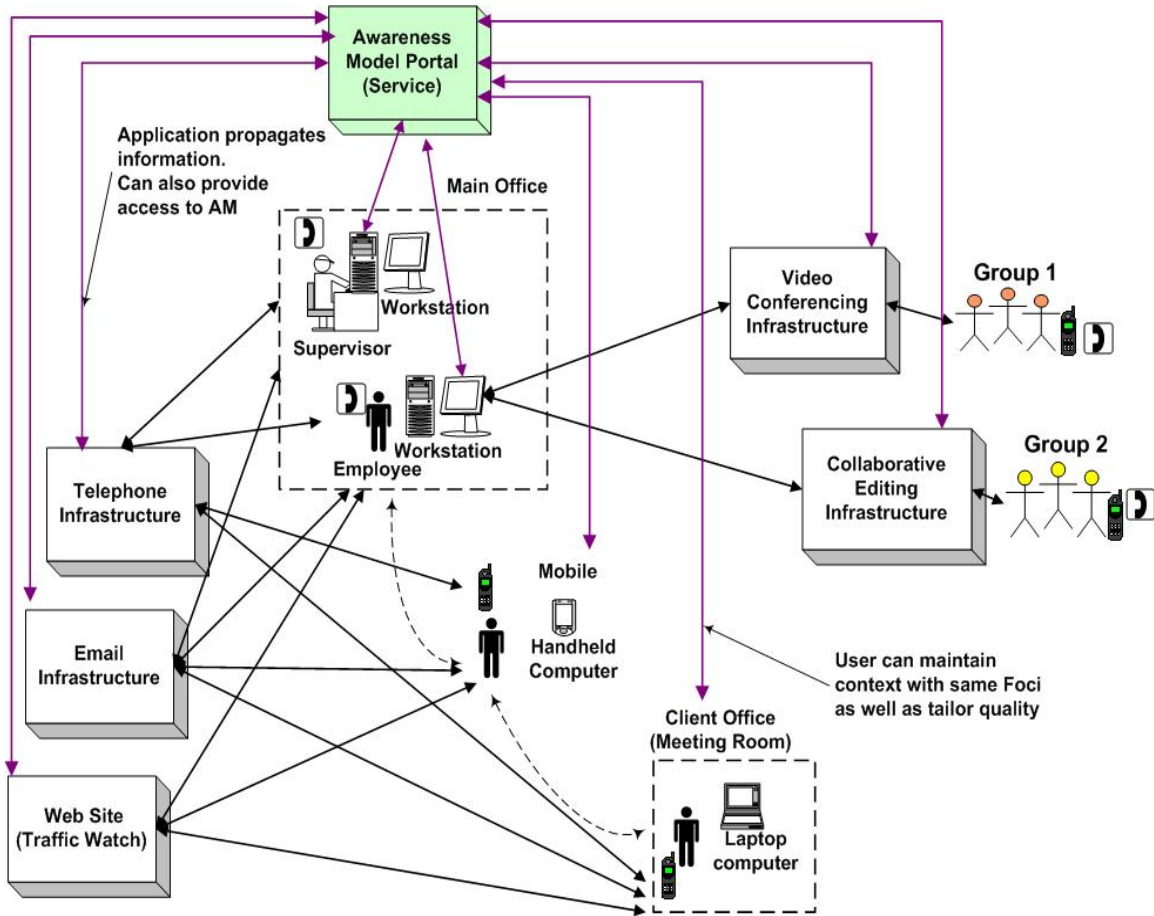
Figure 3.11 Office Collaboration with Awareness Model

In our example, we assume that every application infrastructure used by the actors is connected to the awareness framework and awareness model. Actors have a view of the entire spectrum of information sources available. This spectrum is obtained by using the awareness map, which is nothing but each user's view of the awareness model. Based on the privileges set, each user's view may be different. The privileges themselves are set according to the policies of the group and determine who has access to what information. For example the employee may not be allowed to see all sources of information that the supervisor can access, however he would be allowed to access information from sources such as shared artifacts that he needs to work with. When the employee is at his workstation he can access the video conferencing and collaborative editing servers as before. When he switches to his hand-held computer and later his laptop he can still find maintain his awareness of the activities by selecting these sources on his hand-held and laptop computers. Not only does he select the source he selects the characteristics of the information to be delivered. As an example while using these hosts he may select only the text-based updates of the activities as opposed to the full images and audio. Groups 1 and 2 are also aware of the employee as a participant in the group work process since they have access to the Awareness Model too. Most importantly they are also aware of the fact that he has switched from his workstation to his other hosts. This information is

useful as it keeps them cognizant of his interaction capabilities when they work together. For instance if we know someone cannot access the full image and only see text based messages we may provide a better description of the issues via the message. In the absence of the Awareness Model the employee may have to convey that he would access the VCI and CEI through other hosts and have limited capabilities. Thus apart from acting as a directory of information sources the Awareness Model also helps in reducing the coordination overhead that lack of awareness can cause.

Assuming the supervisor has privileges to monitor the current status of the employee, he is now aware that the employee is out of the main office but is now using other applications and hosts. If the supervisor wishes to contact the employee he knows how he can get in touch with him. More importantly being aware of the employee's status may allow him to reach him in a manner that is not distracting or too obtrusive. For example if the employee is in the meeting with the client, the supervisor may not call him on his cell phone but send him am instant message. The supervisor can gather this information from his Awareness Map. Thus the ability to propagate awareness irrespective of the applications and environment is the key contribution of the Awareness Model.

3.6.3 Some Key aspects of the Awareness Model seen in the Scenario

The above scenario is composed of the users, and the heterogeneous environment, which imply they have ability to be connected and share information in a certain manner. We want to demonstrate that having the awareness model would allow one to accomplish the same tasks and the characteristics as before in a more efficient manner. Among its many capabilities some key aspects demonstrated by the model are:

1. A *focus* is a mechanism to subscribe to information from a collection of sources for an extended period of time. As the sources generate information the user is notified of the information. Users can create *foci* to denote the context of work they are involved in.

2. Apart from *focus*, which has the active sources providing information, the user should see all sources from which it is eligible to receive information. This is part of the *source superset* concept.

3. A user (**user A)** must be able to find out the status of another user (**user B**) irrespective of whether the B's sources are within A's foci. A can find out the status of any user in terms of all the sources the user can be contacted at and the current source he is receiving information at for example: the user B is online on his Instant Messaging Client. In other words this will retrieve information about all the sources she is using i.e. the sources that she is generating and receiving information from. One can choose to add them to an existing focus or create new ones provide the first user has the appropriate privileges to add them to a *focus*. Upon ascertaining their status one can interact with them. Again all the above is provided by the source superset.

4. A user should able to communicate with another user even if this user's information source is not in that user's focus. One can notify someone even if they are not in that person's focus. The receiver the option not to be bothered with the notification. This is the notion of a communication that is independent of the awareness model. In fact this is the manner in which we do communicate now in the absence of an awareness framework. Only difference being that when integrated to an awareness framework, one can ascertain the status of the recipient before initiating a communication which is not registered in the awareness model (for example calling him on his cell phone) or sending him an email but specifying to your email application "not to send this via the awareness framework's rules". thus the awareness framework does not get this information however the recipient still does.

Both the features 3 and 4 above are part of the notion that the Awareness Model is a directory of information sources. In fact querying to ascertain status and contacting them based on their availability is a very basic and essential functionality.

The **Table 3.1** below provides some situations in group work where awareness is needed. It compares how this awareness is propagated in the presence of an awareness model as opposed to when there is none. Our validation of the Awareness Model seeks to demonstrate this behavior. As mentioned before we assume that applications can be connected to an awareness framework and they can provide each user with a view of the awareness map viz. each user's personal view of the awareness model. Depending on the application's capabilities the awareness map may be an actual graphical view or even text-based indicator of the user's capabilities to access information source, his current foci, the list of all other users, the sources they use and their status. As changes occur to those sources the user is notified of them while logged on to the application. For example even while using an email client the user may be notified of someone wishing to contact him as the client may be connected to the awareness model and thus have the capability of being notified. Also the table below describes various situations where human users manually search and retrieve information using the awareness model. This we term as the "Pull-Mode" However intelligent agents can be configured to do this on behalf of the human user i.e. to notify the user of changes that may be of relevance to him. This is the "Push-Mode". There is also the case where both human and agent may work at the same time. In every case we need a common information model off of which such changes can be detected. This common model is the awareness model.

Table 3.1 Specific Cases in the Office Collaboration Scenario
(For the sake of readability **Table 3.1** is continued over several pages)

Collaboration Requirement and Required Awareness Support	Goal/Tasks with respect to Office Collaboration Scenario	Without Awareness Model	With Awareness Model
1. Maintaining multiple work contexts when changes occur. Users change applications, hosts and locations.	In our scenario this involves continuing work with the two groups even though the employee cannot access the information in the same form as he did at his workstation. He changes locations and hosts. For example from the main office to client meeting, from meeting to lunch. (High speed n/w to dialup to PDA and so on.)	The user has to keep a mental note of the context of his work activities. He must also know where and how to connect to (assuming the video conferencing and collaborative editing infrastructures allow access from other hosts). Though it is not difficult to maintain context of current work, it is difficult to remember all the sources the user is working with i.e. all other work activities and contexts each time you change hosts and applications. If he needs to access information about other work activities that are dormant it will not be available.	The user has a view of all his work contexts via the Awareness Map irrespective of the changes. As one change device/application and location: Create the appropriate focus and add the source to the focus. Can be enabled by "Push-Mechanism" i.e. User's agent automatically detects the change and synchronizes both on the source side as well as the employee side and updates the employee's view OR "Pull-Mechanism": User adds the source and synchronizes. In either case we need a common information model off of which such changes can be detected. This common model is the Awareness Model. Since the user has a directory of information sources, irrespective of the level of activity he can access information about other work activities.

Table 3.1 Specific Cases in the Office Collaboration Scenario (continued)

Collaboration Requirement & Required Awareness Support	Goal/Tasks with respect to Office Collaboration Scenario	Without Awareness Model	With Awareness Model
2. Support for meta-information to discover and select information sources.	Employee wishes to determine if he can connect to the Video Conferencing and Collaborative Editing infrastructures from his other hosts. He browses the information provided by each source to see how to connect to them.	Not possible. Must contact someone separately to find out if he can connect to the sources. Lack of connectivity will not only mean inability to communicate but also lack of meta-information about the sources available.	This is possible by searching and/or browsing the meta-information about each source he is allowed to access in the Awareness Model through his Awareness Map.
3. Support for group decision process that can involve negotiations	Ability to communicate one's suggestions and be aware of others suggestions when working with either group. Need for everyone to get on the "same page".	This is possible if there is some form of connectivity to the infrastructure of each group. This connectivity should be provided irrespective of the user's host and application used.	Just as in the case e case where the Awareness Model is absent the infrastructure must allow different hosts with varying capabilities to connect.

97

Table 3.1 Specific Cases in the Office Collaboration Scenario (continued)

Collaboration Requirement & Required Awareness Support	Goal/Tasks with respect to Office Collaboration Scenario	Without Awareness Model	With Awareness Model
4. Dynamic Planning and Coordination (Just-in-time)	Employee is on the road to meet his client. There is a disruption on the freeway. His traffic watch website sends such alerts to him; however he needs to get this on the road. Upon receiving this he can contact the client and inform him of the delay. This can help the client proceed or engage I other activities till they meet.	To successfully schedule dynamically the user must be aware of both pieces of information at about the same time. The issue is similar to the maintenance of current context. The employee has to keep track of the website and connect to it each time he switches. This is assuming the website automatically alerts to its subscribers. Assuming he is in "Pull-Mode" where he has to check the website then, the following problems can occur: May not explicitly check the website for delays or, when checking the website he may yet have information about the meetings.	With an Awareness Model and Map he has the current contexts of interest on each host. Can easily add the source to his focus and retrieve information. An awareness model agent on the employee's behalf can automatically synchronizes the latest information with his hand-held. Thus any change is immediately propagated allowing him to take a decision about when he can make it to the meeting. This he can convey to his clients.

Table 3.1 Specific Cases in the Office Collaboration Scenario (continued)

Collaboration Requirement & Required Awareness Support	Goal/Tasks with respect to Office Collaboration Scenario	Without Awareness Model	With Awareness Model
5. Efficient allocation of resources	One group is waiting on another group for information Awareness of the amount of time and nature of delay can lead to the other group's better using its time till it resumes work. Similarly competition for scarce resources can be resolved.	The group that is going to cause the delay must explicitly inform the other group about the delay and how long it takes. Or the second group must make an effort to monitor the first group's progress. Even finding out how to monitor status and whom to contact can be difficult since there is no information available about the information sources. The extra coordination and effort to propagate such information (by either group) often gets overlooked. This results in inefficiencies.	If the second group determines that it needs to be aware of the first group's status all it need to do is find out from the Awareness Model the source(s) of information that first group uses to record its status. It can add this source to its focus. For example this could be a project management application (Microsoft Project). Once this is added to the second group's focus, as soon as any changes are made to the current status of the first group, the second group is notified of it and can plan accordingly. More importantly the second group is also receiving information from other sources and can use this in conjunction with that information to plan and coordinate other tasks.

Table 3.1 Specific Cases in the Office Collaboration Scenario (continued)

Collaboration Requirement & Required Awareness Support	Goal/Tasks with respect to Office Collaboration Scenario	Without Awareness Model	With Awareness Model
6. Opportunistic Collaboration by selecting relevant type of awareness source. This is unanticipated as the supervisor chose to take advantage of the second client's presence. Also Limiting the amount of distraction caused.	Another client visits the main office. Supervisor wishes to update him of their work. He needs to ascertain current progress from his employee but is not aware of his current status. He must contact employee and ask him for a progress update that he can pass onto the client. As part of the update the employee wishes to send the document he is currently editing with Group 2. He instructs Group 2 to send the supervisor a copy directly.	Supervisor does not know what the employee is doing and has to contact him. He checks the calendar and finds out that the employee is away at the client office for a meeting. He prefers not to call his cell phone as he may be in the meeting. Would like to text-message him but is not aware of his contact details and if he is currently logged on.	Supervisor's awareness map shows employee's profile and current status. He is available via cell phone/email/IM/etc. with corresponding contact information. Supervisor contacts by text-messaging him on his instant messenger. This is done without unduly distracting or disturbing him while he is in the meeting.

As employee knows the status of the artifact that is being created with Group 2 he decides to send the document to the supervisor. He instructs the Group No. 2 to send the copy to the boss. Group 2 already knows the boss source details in its source superset and just forwards the document.

The supervisor receives the email and the document. Group 2 informs the employee that the |

			document has been sent. If the collaborative editor itself is used to send the document (assuming it has the capability to send copies of documents via email, then this information is automatically propagated to the employee since he is aware of all activities in the collaborative editor) All these activities can occur concurrently with minimal distraction and effort on part of the employee who is in a meeting with clients.

Table 3.1 Specific Cases in the Office Collaboration Scenario (continued)

Collaboration Requirement & Required Awareness Support	Goal/Tasks with respect to Office Collaboration Scenario	Without Awareness Model	With Awareness Model
7. Improving the quality of the end product. This benefit is unanticipated.	When one group is aware of the activities pf the other group, some information obtained helps in modifying certain characteristics of the second group's work product. This can lead to improvement of quality. Often such behavior is unanticipated and is a consequence of the fact that groups are aware of other groups activities	Unless it is anticipated that one group needs to be aware of another group's activities no explicit mechanism is provided to propagate awareness of activities.	Since the Awareness Model provides the complete directory of information sources and their characteristics, if necessary groups can add other groups' information sources to their foci. In our example Group 2 may be able maintain awareness of the video conferencing work group, Group 1. Assuming there is some dependency between the two groups' work they may use their awareness to take a decision that impacts their work product based on the Group 1's work. As the employee is aware of both groups he is automatically aware of with the decisions taken by Group 2 as reflected by their interaction through the collaborative editor infrastructure. The extra overhead necessary to update everyone is avoided. Thus it supports group decision processes.

Table 3.1 Specific Cases in the Office Collaboration Scenario (continued)

Collaboration Requirement & Required Awareness Support	Goal/Tasks with respect to Office Collaboration Scenario	Without Awareness Model	With Awareness Model
8. Recording decisions and maintaining a decision trail along with an "Awareness Trail". Why someone did or did not act in a certain manner can often be answered by knowing who was aware of the fact.	During the course of work with each group he employee would like to search the recorded minutes of previous meetings with each group. Certain decisions were taken based on the people involved at that time. This knowledge of who was involved and what they knew can be useful.	No such direct support exists. Unless minutes of previous meetings are compared with a view to finding out the level of awareness of the participants such information is not available.	The Awareness Model can be extended to record events in a "Transcript" module. This Transcript not only provides a log of all events that have occurred but information such as who were the participants involved, what was there level of awareness about specific issues and so on. Such information can be inferred through queries. For instance if everyone's awareness map at a particular time is collectively viewed one can infer the collective awareness of the entire group. Can use this transcript to also search on past messages from the sources.

3.6.4 Awareness Model for Office Collaboration Scenario

This section describes the awareness model created for the Office Collaboration Scenario described above. The Office Collaboration Scenario awareness model was created and used in the validation process to validate the theoretical concepts of the awareness model. The validation (described in Chapter 4) was done using simulations involving human interaction. An Awareness Simulator application was created which enabled humans to assume roles and act out the scenario. The awareness simulator created an awareness framework where the model was the central entity. The awareness simulator consists of a client, which can be used to simulate the behavior of any role and interactions of the role with the corresponding information sources (applications that the user is employing in collaboration). The awareness simulator ties all the clients (information sources) to the awareness model just as a real awareness framework would but in addition performs various functions such as logging all activity, which can later be analyzed to study the awareness model's effectiveness. The awareness model that will be described in this section was used as the candidate for those simulations. The details of the Awareness Simulator application and the manner in which the simulations were carried out are part of Chapter 4. Here the design of the awareness model used will be discussed.

This awareness model design demonstrates the manner in which the different collaboration related activities can take place as described in **Table 3.1** above. Many of the generic awareness model features are seen in this design. As mentioned before, the awareness model is a generic concept. Each collaborative effort will have a specific awareness model designed to cater to the needs of that particular effort and will exhibit the corresponding features. Consequently choice of technology to implement such awareness model features will also vary. All the generic awareness model features were not implemented in this example and key concepts were implemented. This research emphasizes the need for an awareness model (an awareness framework) with certain desired character5itis and does not enforce any one implementation methodology over the other.

A relational database was used to implement the awareness model. The MySQL [MySQL 2005] relational database management system (RDBMS) was the choice made. The desired features that were to be validated were implemented in the form of tables in the database. Users assuming roles interacted with each other in enacting the collaboration scenario and the awareness model provided the awareness support for the scenario. The tables in the database consisted of all the necessary information required such as role information, sources information, privilege information and others as will be clear from the description below. Users' interaction resulted in queries being made to these tables to retrieve the information. Based on the tables, users were shown the appropriate sources in their source supersets, they were notified about status of other users and the sources they were working on, they were allowed to perform actions such obtaining information about sources and enhanced forms of awareness. Most importantly they were allowed to select the information sources that are visible in their awareness map and add them to foci they create. Information generated by those sources is now available to the users interested in them and appears in their client. Interaction with their own sources was simulated by messages they entered and sent to the awareness framework. This simulated the user interactivity using different applications but with

respect to the awareness framework and model. The database schema is as shown in **Figure 3.12** below.

run_info_t	
run_id	VARCHAR(30)
scenario_name	VARCHAR(45)
scenario_desc	VARCHAR(200)
we_desc	VARCHAR(200)
run_desc	VARCHAR(200)

active_user_t	
user_id	VARCHAR(20)
source_id	VARCHAR(20)
active	VARCHAR(5)

second_order_t	
parent_user_id	VARCHAR(20)
child_user_id	VARCHAR(20)
second_order	VARCHAR(5)

user_focus_t	
user_id	VARCHAR(20)
focus_id	VARCHAR(20)
focus_name	VARCHAR(25)
focus_desc	VARCHAR(200)

user_t	
user_id	VARCHAR(20)
password	VARCHAR(10)
last_name	VARCHAR(25)
first_name	VARCHAR(25)
role	VARCHAR(15)

source_t	
source_id	VARCHAR(20)
source_name	VARCHAR(45)
source_desc	VARCHAR(200)
preferred_med	VARCHAR(25)

user_src_t	
user_id	VARCHAR(20)
source_owner	VARCHAR(20)
source_id	VARCHAR(20)
cp	VARCHAR(3)
ip	VARCHAR(3)
focus_id	VARCHAR(20)
reverse_lookup	VARCHAR(3)
wisiwys	VARCHAR(3)

event_log_t	
timestamp	DATETIME
message_type	VARCHAR(30)
user_id	VARCHAR(20)
source_id	VARCHAR(20)
to_user_id	VARCHAR(20)
to_source_id	VARCHAR(20)
recv_user_list	VARCHAR(255)
focus_id	VARCHAR(20)
focus_name	VARCHAR(100)
focus_desc	VARCHAR(150)
active	VARCHAR(5)
action	VARCHAR(5)
message	VARCHAR(255)
reply	VARCHAR(25)
cp	VARCHAR(5)
ip	VARCHAR(5)
source_name	VARCHAR(100)
source_desc	VARCHAR(200)
preferred_med	VARCHAR(20)
awareness_map	VARCHAR(255)
src_super_set	VARCHAR(255)
rev_lookup	VARCHAR(255)
wisiwys	VARCHAR(255)
rational	VARCHAR(255)

Figure 3.12 Awareness Model for the Office Collaboration Scenario

Each table in the database serves to exhibit key features of the awareness model and is described as follows. Some tables in the model were initialized with data that represented the user information, privilege information, and source information among others. The database also consists of tables are necessary for recording information from the simulation. Each table is described below with a description of its fields and initial data.

NOTE on the Generic Database Schema

The tables that represent the Office Collaboration Scenario can be considered as generic if any other scenario has the same awareness propagation requirements. However group collaboration varies significantly and each example can have specific requirements. So an awareness model created will have the corresponding information and hence a different schema. For example, one may wish to express more information about their sources (in the Office Collaboration example, we have chosen very limited information), thus different number and types of fields may be needed. This changes not only the schema but also the programs used to query the database. However the awareness model concepts (as illustrated by the table schema for the Office Collaboration Scenario) remain the same irrespective of any scenario. The table schema for this example, can be considered as an implementation of a basic awareness model that can form the core and be extended or modified to include more information and thereby exhibit more versatility. Currently the author's research efforts are in devising means where users can adapt such a generic core to other collaboration scenarios.

NOTE on Database Creation and Initialization

Each run has the same initial data so the database is created and initialized automatically through a script that populates it with all the required data. This script is executed in the Awareness Simulator when the user wishes to start a new simulation run.

run_info_t

This table is used to store information about each run of the simulation. Before the simulation run commences this table is manually populated with information since each run differs. All the other tables are initialized through a script.

run_id: Used to uniquely identify each run.
scenario_name: Name of the scenario for example, "Office Collaboration Scenario".
scenario_desc: Short description of the scenario.
we_desc: Description of the Work Environment (Explained in Chapter 4).
run_desc: Description of the objectives of the run and any features.

run_id	scenario_name	scenario_desc	we_desc	run_desc

user_t

This table contains information about all the different members involved in the Office Collaboration Scenario. This table is accessed when a client program is launched and the human user logs onto the Awareness Simulator with a corresponding member role. In the scenario though Group 1 and Group 2 are groups of people, we assume that they are represented by a designated 'point of contact' (POC viz. grp1 and grp2) for simplicity. In our scenario we found it was sufficient to represent multiple users by one entity in the case of the Group1 and Group 2 since in the scenario all members accomplished the same objectives. From a simulation perspective it helped since we did not have to look for more participants and train them to use the simulator. The 'all' is a user_id, which represents all the users who can access a certain source such as a common office calendar. In our example both the employee (emp) and the supervisor (sup) can access the calendar. When a certain source is said to be accessed by 'all' it means by everyone who belongs to 'all'.

Also we have a traffic watch website (with user_id 'twws') which is accessed by the employee to keep track of the traffic and road conditions. We have an automated agent program that generates traffic alerts. To start the traffic website alert service we need to launch a client. For this purpose a separate role was created and entered in this table.
Please Note: The word 'role' above is used to mean a role in the collaboration scenario. The word is also used as a field in the table. Here it is used to mean the type of user accessing the awareness simulator and can either be a regular user or an administrator with privileges to add and delete sources and other roles. However the administrator functionality is yet to be implemented.

user_id: Unique user name credential.
password: Authentication credential.
last_name: Last name.
first_name: First name
role: Indicates if the client is just a user or other (for instance an administrator with greater privilege)
Since both the 'all' and 'twws' do not have corresponding human users such as the employee and the supervisor, the name fields are set to 'na' (not applicable).

Table 3.2 Awareness Model Example: user_t Table Initial Data

user_id	password	last_name	first_name	role
emp	emp	Smith	John	user
grp1	grp1	Handey	Jack	user
grp2	grp2	Williams	Jill	user
sup	sup	Cook	Thomas	user
all	all	na	na	user
twws	twws	na	na	user

source_t

This table contains fields that describe every information source in the example. Please note that every application involved is considered a source of awareness information. Even if users use an application to access information from others for example, if user A uses his email account to access the email sent by user B, user A's email account is a source because user A can use the same account to send email. Furthermore if user B is interested in knowing if user A has accessed the email sent, then upon user A's access to the email, his account can notify user B. Thus user A's email account becomes a source. This table is accessed during the collaboration when any user needs to ascertain information about a source and its characteristics. Very limited amount of information about the sources has been shown in this example. However this is the table that would consist of the meta-information for each source and its medium and that would enable users to select and tailor the quality of information. Hence it can be set up to provide as many detail as necessary. In our example users can select and deselect sources and add them to their foci, based on the source description.

source_id: Uniquely identifies each information source (awareness source).
source_name: Name of the source.
source-desc: Describes the source characteristics, an example of the meta-information.
preferred_med: The medium with which the source should be accessed for best results i.e. the preferred medium of the source. For example, a streaming video server may suggest that its files may be accessed via broadband as opposed to a dialup modem. IN our example we used the field as an example of meta-information.

Table 3.3 Awareness Model Example: source_t Table Initial Data

source_id	source_name	source_desc	preferred_med
s1	Calendar	Office Calendar accessible by all office members. Hosted on the office's web server and accessible via the WWW to office members only.	Broadband
s2	Supervisor Email Account	The supervisor's Web-based email account.	Broadband
s3	Supervisor Instant Messenger Account	The supervisor's Instant Messenger account.	Broadband

s4	Supervisor Cell Phone	The supervisor's cell phone	cellular network
s5	Employee Email Account	The employee's Web-based email account.	Broadband
s6	Employee Instant Messenger Account	The employee's Instant Messenger account.	Broadband
s7	Employee Cell Phone	The employee phone	cellular network
s8	Employee Workstation Video Conferencing Client	The employee's client with which he connects to the Video Conferencing Server. Supports high quality video, text messaging and file sharing.	Broadband
s9	Employee PDA Video Conferencing Client	The employee's client with which he connects to the Video Conferencing Server. Supports text-based messaging only.	broadband wireless
s10	Employee Laptop Video Conferencing Client	The employee's client with which he connects to the Video Conferencing Server. Supports high quality video, text messaging and file sharing.	Broadband

s11	Employee Workstation Collaborative Editor	The employee's collaborative editor client with which he connects to the Collaborative Editing Server. Supports WYSIWIS editing and text messaging for multiple users.	Broadband
s12	Employee PDA Collaborative Editor	The employee's collaborative editor client with which he connects to the Collaborative Editing Server. Supports text messaging only for multiple users.	broadband wireless
s13	Employee Laptop Collaborative Editor	The employee's collaborative editor client with which he connects to the Collaborative Editing Server. Supports WYSIWIS editing and text messaging for multiple users.	broadband
s14	Employee Traffic Watch Website Account	The employee's Traffic Watch Website Account. Alerts employee to current traffic and road conditions.	Broadband

s15	Group 1 Video Conferencing Client	The group's client with which they connect to the Video Conferencing Server. Supports high quality video, text messaging and file sharing.	Broadband
s16	Group 1 Email Account	Group1 POC's email account	Broadband
s17	Group 1 Cell Phone	Group 1 POC's cell phone	cellular network
s18	Group 2 Collaborative Editor	The group's collaborative editor client with which they connect to the Collaborative Editing Server. Supports WYSIWIS editing and text messaging for multiple users.	Broadband
s19	Group 2 Email Account	Group2 POC's email account	Broadband
s20	Group 2 Cell Phone	Group 2 POC's cell phone	cellular network

user_src_t

The table forms the core of the awareness model for this example. The fields define what each user views in his or her source superset, along with the respective privileges. It records the foci that each user places the source in. It sets the privileges based on which users can obtain enhanced forms of awareness as described below.

user_id	Source_owner	source_id	cp	ip	focus_id	rev_lookup	wisiwys

The fields **user_id**, **source_owner** and **source_id** are related in the following manner.

For each user Ui in the **user_id** field, it indicates all possible user-source combinations, which are in the **source_owner** and **source_id** fields that Ui can contact and/or access information from. Those user-source combinations from which information can be received can be added to a focus to receive information continuously. For example, as shown in the initial data below, the user **sup** can access information from the sources **s5**, **s6** whose owner is the user **emp**.

If Ui needs to contact and/or receive information from any user Uj who is using a particular source Sj, the user Ui must have corresponding to his own user_id an entry in the source_owner and source_id fields. For example, If Ui is the user-id of the consumer and Uj and Sj are the user-source combination of interest then Uj and Sj must be entered in the source_owner and source_id fields corresponding to Ui. Additionally he must have privileges to be able to contact and or receive information. Only if a privilege is allowed the user may contact or receive information (depending on the privilege) otherwise the user may only get basic information about the user and source (**from the source_t** table) but cannot do any more.

In **Figure 3.13** below, we see a screen shot of the awareness simulator client for the employee (emp) role. The **user_src_t** table is the main table, which is used to populate all the information in this awareness client. The smaller window (**Users List)** to the right is the **source superset** for the employee. This is populated with the list of all the users and the sources owned by them and is in the form of a tree. Thus it is populated using the first three fields (**user_id, source_owner, source_id**) of the user_src_t table. The employee can see his own sources in the list. He can use this source superset tree to obtain information about a source including the privileges he has on them (**Contact Privilege CP** and **Information Privilege IP** discussed next), view the status of the source and its owner to see if the source owner is active (indicated by a green font) or inactive (indicated by red font). Upon change in status the employee is notified of the same with change in color. The larger window in the center is the **Awareness Map** with all the sources indicated as squares, again with their active and inactive status. The employee can use the same source superset and invoke the *reverse lookup* , *wisiwys* and *second order* functionality to obtain enhanced awareness. To simulate the activities that the employee undertakes in interacting with various sources (applications) the human user assuming the role of the employee uses the input textbox (**User's Actions**) below to enter messages. These messages indicate the activity such as sending an email, or an instant message, making a phone call, reading, taking to a person in the same room and

so on. The strings are entered in a certain format to indicate the type of message. As this client is part of the simulator the user is allowed to enter not only messages for the scenario but also can record observations and notes for the analysis. The collaboration Events window displays all the messages meant for this user. For example, in **Figure 3.13** we see that the employee has many sources (s1, s15, and s3) within different foci that are represented by the colored ellipses. Information generated by these sources (events are represented in the form of messages entered by the users who own these sources in their respective User's Actions input box) are propagated to this user viz. the employee and are displayed in the Collaboration Events window. Even the messages the employee enters are echoed in that window. Also notifications are displayed in the window apart from being shown in the Awareness Map and the Users List in the form of color changes. Details of the awareness simulator and user interaction are provided in Chapter 4.
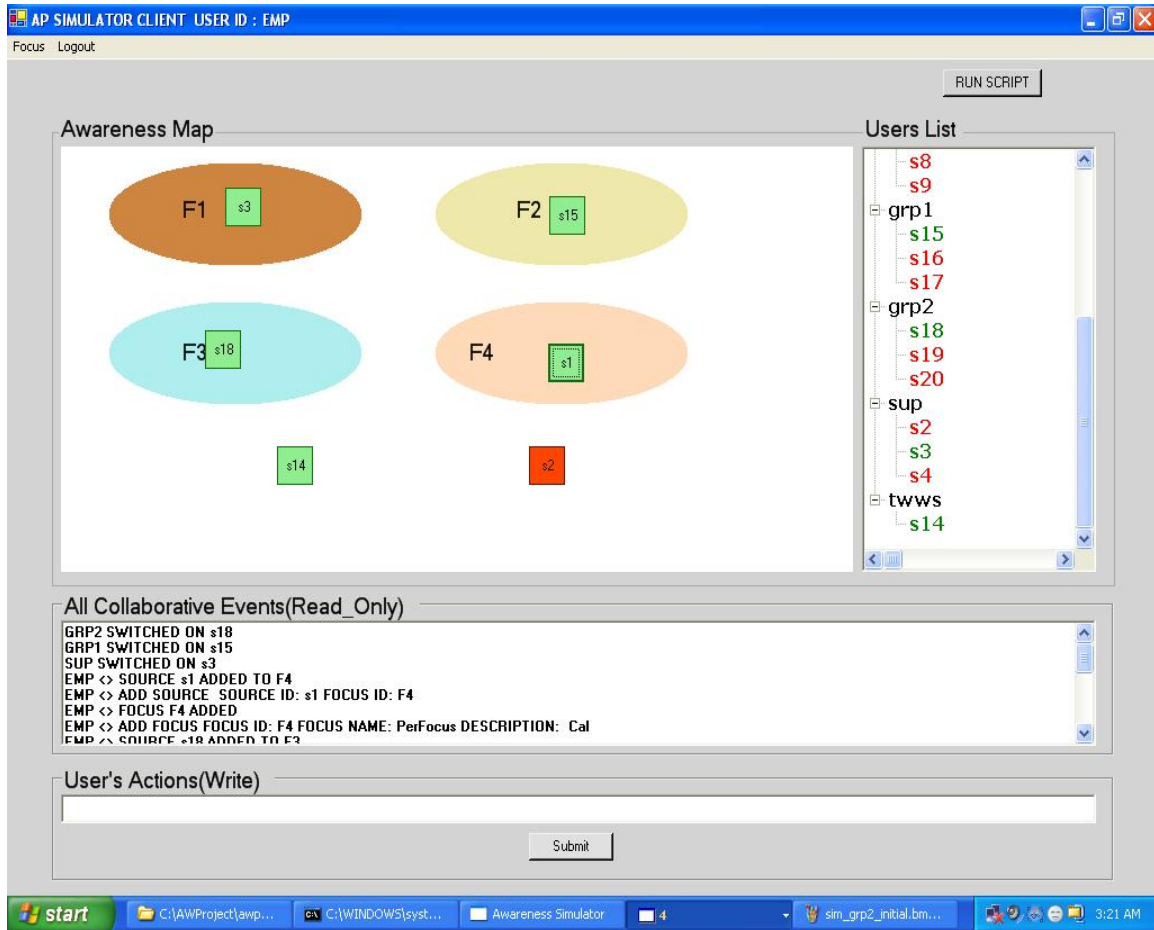
Figure 3.13 Awareness Client (Employee's View)

Privileges

Two kinds of privileges have been implemented. **Contact Privilege (cp)** and **Information Privilege (ip),** which are as follows.

Contact Privilege (cp)

To enable any user to make available his contact information to other users selectively we use contact privileges indicated by the **cp** flag. Just as one needs to have someone's contact information to contact that person (such as having their phone number or email id), having a cp on someone's source means the user can contact the other user at their source. Please note that every application used as part of the group's collaboration, is considered a source of awareness information. For example email accounts, IM accounts, cell phones etc. Thus user A has to have contact privileges on user B's email account to send him an email.

Information Privilege (ip)

The information privileges are indicated by the '**ip**' flag. Information Privilege is applied to any source that generates information of interest to the recipient. If Ui has **ip** privileges on a particular user-source combination, any and all information generated by that source is received by Ui. A group calendar, a web site generating information to subscribers, or even an email server or email account which has mechanisms to propagate all emails generated by the account (as is seen in online groups and communities) can be assigned **ip**. For instance if you send email to an online group you are a member of then your email automatically gets sent to all the members. This is a mechanism to enable users to make available the information content they generate selectively to other users.

The fields **'cp**' and **ip**, where applicable are set to **"yes" (Have privileges)**, **"no" (No privileges)** and where not applicable based on the types of source they are set to **"na" (Not Applicable).** As mentioned above only if the flags are set to "yes" any action can be taken, Otherwise only basic information (from the source_t) can be obtained. **To actually receive the information the user-source combination of interest must be in a receivers focus**.

In **Figure 3.13** we see that the employee can view all the sources on which he has CP or IP or both in the Users List windows to the right. He can view all the sources for which he has the Information Privilege (discussed below) in the large **Awareness Map** window in the center. Due to space constraints in the graphical user interface the employee's own sources are not seen here but from the data in the table we can see that every user has Information Privilege on own source. Information about the Status will be discussed with the "active_status_t" table.

The above table demonstrates the concept of "Directory of Users and Sources" which is essential for awareness.

Focus

The **focus_id** field indicates the unique identifier of the particular focus that the user adds the source to. Only those sources with Information Privilege can be added to the focus. Therefore the values that this field can take are "nofocus" (when the source has Information Privilege but is not added to a focus), a valid focus_id string given by the

user (when the source is added to a focus) and "na" (when there is no Information Privilege on the source). This also helps in retrieval of same information using just a straightforward query rather than having to check if the user has any Information Privilege on the source.

The assumption **in this implementation** is that a source can only be added to one focus at ay given time. This is for the sake of simplicity and the awareness map user interface is designed correspondingly.

In **Figure 3.13** the employee can create foci (using the menu above) and drag and drop these sources he wishes to receive information from into the foci. The moment he does so the source owner receives a notification that his source has been added to the employee's focus. All information generated by that particular source is received by the employee and if he wishes not to receive anymore he can drag the source out of the focus he has created. Foci that are created can be given names and identifiers when they are created. They can also be deleted.

Reverse Lookup

If user U_j and source S_j are in any user U_i's focus and if the reverse lookup flag (**rev_lookup** field) is set to **"yes"** then User U_j can find out if user U_i has the combination U_j-S_j in his focus. Essentially it answers the question, *who all can get information from me or in whose focus am I in?* The values are **"yes"**, **"no"** and in when not applicable **"na".** In our scenario we assume that if another user cannot access information from your source, by definition he cannot add it to his focus and hence rev_lookup field will have a "**no**" value.

What I See Is What You See (WISIWYS) Privilege

A user U_i who sees a source S_j owned by U_j in his Information Privilege set (the set of all user-source combinations on which he has Information Privilege) can see who are the other users who can see this particular source if and only if he has the WISIWYS privilege. Upon invoking the WISIWYS function on the source (through the awareness map) he will see a list of other users and information telling him if they have included it in a focus or just have it in their Information Privilege set.

Common Sources

A source can be used by more than one person. For example the Group Calendar with **source_id** set to 's1'. Multiple users can update their schedules on a calendar. Depending on the access control policy the calendar may be readable by all at any time. Or certain people may have read-only access while some others have selective read-only access where they are not privy to certain meetings scheduled. Similarly the write privileges may be selective and user specific. In our scenario sources exhibit typical behavior. For the group calendar application that is readable by all and has multiple users as its owner, "**all**" is entered in the **source_owner** field. Thus it is displayed only once in each user's awareness map even if it has multiple users who can be its owners and is also displayed only once in the source superset window. In other situations where a source can be used by multiple users, but NOT all the users, each user-source pair has to be recorded as always. Reverse Lookup will not be applicable and there will be a **"na"** in that field. The

above assumptions are made for this scenario but can be changed depending on the scenario.

Table 3.4 Awareness Model Example: user_src_t Table Initial Data

user_id	source_owner	source_id	cp	ip	focus_id	rev_lookup	wisiwys
sup	All	s1	na	yes	nofocus	na	yes
sup	emp	s5	yes	yes	nofocus	yes	yes
sup	emp	s6	yes	yes	nofocus	yes	yes
sup	emp	s7	yes	no	na	no	no
sup	emp	s8	no	no	na	no	no
sup	emp	s9	no	no	na	no	no
sup	emp	s10	no	no	na	no	no
sup	emp	s11	no	no	na	no	no
sup	emp	s12	no	no	na	no	no
sup	emp	s13	no	no	na	no	no
sup	sup	s1	yes	yes	nofocus	na	yes
sup	sup	s2	yes	yes	nofocus	yes	yes
sup	sup	s3	yes	yes	nofocus	yes	yes
sup	sup	s4	yes	yes	nofocus	yes	yes
emp	all	s1	na	yes	nofocus	na	yes
emp	sup	s2	yes	yes	nofocus	yes	yes
emp	sup	s3	yes	yes	nofocus	yes	yes
emp	sup	s4	yes	no	na	no	no
emp	twws	s14	na	yes	nofocus	na	na
emp	grp1	s15	yes	yes	nofocus	yes	yes
emp	grp1	s16	yes	no	na	no	no
emp	grp1	s17	yes	no	na	no	no
emp	grp2	s18	yes	yes	nofocus	yes	yes
emp	grp2	s19	yes	no	na	no	no
emp	grp2	s20	yes	no	na	no	no
emp	emp	s1	yes	yes	nofocus	na	yes
emp	emp	s5	yes	yes	nofocus	yes	yes
emp	emp	s6	yes	yes	nofocus	yes	yes
emp	emp	s7	yes	yes	nofocus	yes	yes
emp	emp	s8	yes	yes	nofocus	yes	yes
emp	emp	s9	yes	yes	nofocus	yes	yes
emp	emp	s10	yes	yes	nofocus	yes	yes
emp	emp	s11	yes	yes	nofocus	yes	yes
emp	emp	s12	yes	yes	nofocus	yes	yes
emp	emp	s13	yes	yes	nofocus	yes	yes
grp1	sup	s2	yes	no	na	no	no
grp1	sup	s4	yes	no	na	no	no
grp1	emp	s5	yes	no	na	no	no

grp1	emp	s6	yes	no	na	no	no
grp1	emp	s7	yes	no	na	no	no
grp1	emp	s8	yes	yes	nofocus	yes	yes
grp1	emp	s9	yes	yes	nofocus	yes	yes
grp1	emp	s10	yes	yes	nofocus	yes	yes
grp1	grp1	s15	yes	yes	nofocus	yes	yes
grp1	grp1	s16	yes	yes	nofocus	yes	yes
grp1	grp1	s17	yes	yes	nofocus	yes	yes
grp2	sup	s2	yes	no	na	no	no
grp2	sup	s4	yes	no	na	no	no
grp2	emp	s5	yes	no	na	no	no
grp2	emp	s6	yes	no	na	no	no
grp2	emp	s7	yes	no	na	no	no
grp2	emp	s11	yes	yes	nofocus	yes	yes
grp2	emp	s12	yes	yes	nofocus	yes	yes
grp2	emp	s13	yes	yes	nofocus	yes	yes
grp2	grp2	s18	yes	yes	nofocus	yes	yes
grp2	grp2	s19	yes	yes	nofocus	yes	yes
grp2	grp2	s20	yes	yes	nofocus	yes	yes

Modifying Privileges
All awareness model privileges and flags (CP, IP, Reverse Lookup, Second Order seen later) can be changed dynamically based on the needs of collaboration. In reality users may themselves chose to make available their information or users with the appropriate authority may do so such as a supervisor or policy maker. To do so, a user must be able to view a summary of all his sources and privileges that he and others have on those sources. To keep one's user-source combination from being seen in another user's awareness map, one needs to simply delete the particular user-source combination from that user's row. In our scenario we keep the Awareness Model privileges viz. CP and IP fixed and have yet to implement the ability for a user to change his/her own settings.

Default Application Behavior
An extremely important aspect is that information propagation need not follow the Awareness Model. Users may wish to send and receive information to other users according to their personal preferences and needs. The users may be involved in the collaboration or may be external. For instance user A may want to send an email to some user B but privately and cannot do so since upon using the source (his email account) the message get sent to others as well (due to the Awareness Model settings). Note that a user's email account may be his/her personal account, which is also used in the group collaboration. Even though the user may use the same source (application, host) in the collaboration not all information generated may be for the group. The awareness model should not take over the user's source but only take information meant for it.
For such selective propagation two approaches can be followed:
The first is to empower the user to customize the source itself and only allow certain information to reach the awareness model. Once the information reaches the awareness

model it follows the privileges set. For this approach to work mechanisms can be employed at the application-interface to the awareness framework and model.

User-based Filtering of email messages is an example. Though generally applied to incoming messages the same could be applied to outgoing information.

An example: The hotmail server has a filter allowing only messages with certain attributes to go to the awareness model.

The second approach would be to establish more fine-grained access control and privilege mechanism in the awareness model itself. Here everything gets propagated but filtered at the awareness model level.

A third would be a hybrid of the above two where both are done to some extent. This is an example of a mixed-initiative where both the user and the awareness model act simultaneously. Adjusting the boundary between the two is a interesting research question. In any case, the application interface, medium and security policies of the sources as well as the organization(s) hosting and using the awareness model would greatly dictate the final shape of the solution.

A related issue is in either approach who would be allowed to set the privileges. The users themselves or someone would dictate the privilege setting. Again one possible hybrid approach could be followed with users having certain privileges on the source side while the privileges on the awareness model side could be set by the project organization.

In our Awareness Model we try to implement a hybrid approach that is simple. The CP and IP are currently set by a super user in-charge of the group collaboration according to group collaboration policies. Individual users cannot change them. However to mimic the activities of a user being able to propagate awareness independently of the awareness model using any source we use a special type of message called the "personal message", which has a certain format. This message is disregarded by the awareness model and goes directly to the recipient thus simulating default application behavior. This simulates the action of communication and information sharing and propagation independent of the awareness model. An example of such would be sending a personal message to one's family member using the same source. If ensuing responses arrive they will not affect in any way the awareness model since that recipient (family member) is no way associated with the collaboration.

Notion of Groups

The above privilege mechanism could be extended even within a single Awareness Model where even while using the same source the user could send messages to different groups based on their current context of work. So though recipients of both groups have IP privileges on the user's source of information they do not both get the same messages. Mechanisms such as filtering would be employed to route only the pertinent messages to the appropriate group. Thus the Awareness Model would be enhanced. Of course another alternative is to have sub-Awareness Models within the main model or even create completely different models for different groups. There is no restriction since the source is an independent entity.

second_order_t

Second Order Focus
This is a concept where user U*i* (indicated in the **parent_user_id** field) can find out all the child foci of a user U*j* (**child_user_id** field**)** that is in his current focus. To enable this functionality, we have the second_order_t. This describes which user has 2nd order privileges on which other user. In our scenario the assumption made is that second order is relevant on any user on whose source one has IP privileges. Thus we do not see grp1 and grp2 have an entry for the sup. The values allowed are "yes" and "no".

For example, the supervisor (sup) can see the awareness map of the employee. This is the policy of the office. So all the foci and the sources that the employee has in his awareness map will be visible to the supervisor. This is indicated by the 'yes' value in the **second_order** field for the sup and emp combination in the first row. The reverse is however not true as the employee does not have the second_order field set to 'yes' for the supervisor. Based on the scenario it does not seem typical for the employee and the groups to have such privilege over each other and hence we see the field set to the value 'no'. However it can be changed if necessary. The initial data is shown in the table below.

Table 3.5 Awareness Model Example: second_order_t Table Initial Data

parent_user_id	child_user_id	second_order
sup	emp	yes
emp	sup	no
emp	grp1	no
emp	grp2	no
grp1	emp	no
grp2	emp	no

active_user_t

This table provides users with a notification of the current status of other users by indicating which sources they are currently active on. The fields **user_id** is meant to indicate the source owner and the **source_id** field holds the source identifier. The **active** field keeps track of the status of the status of the particular source. This information is displayed in the awareness client. In **Figure 3.13** one can see both in the **User List** window as well as the **Awareness Map** window, active sources are indicated with green letters and squares while inactive ones are red. The data in the table below is the initialization data.

Table 3.6 Awareness Model Example: active_user_t Table Initial Data

user_id	source_id	Active
all	s1	Yes
sup	s2	No
sup	s3	No
sup	s4	No
emp	s5	No
emp	s6	No
emp	s7	No
emp	s8	No
emp	s9	No
emp	s10	No
emp	s11	No
emp	s12	No
emp	s13	No
twws	s14	Yes
grp1	s15	No
grp1	s16	No
grp1	s17	No
grp2	s18	No
grp2	s19	No
grp2	s20	No

user_focus_t

This table keeps track of the information about all the foci that each user creates.

This table will have no initial data, as users will create the required foci after logging onto the simulator. The users will be able to add only those sources that are in their Source Supersets in other words only those sources on which they have information privileges (IP).

Table 3.7 Awareness Model Example: user_focus_t Table Initial Data

user_id	focus_id	focus_name	focus_desc

For example: Let us assume that the users choose to create foci and add sources to them. Upon creating the foci the table could be populated as shown.

user_id	focus_id	focus_name	focus_desc
sup	supF1	Calendar Focus	Informs of all Office Calendar Changes
emp	empF1	Calendar Focus	Informs of all Office Calendar Changes
emp	empF2	Document review focus	Informs of all events related to the Document Review process
emp	empF3	Collaborative Editing Focus	Informs of all events related to the Collaborative Editing process.
grp1	grp1F1	Document review focus	Informs of all events related to the Document Review process
grp2	grp2F2	Collaborative Editing Focus	Informs of all events related to the Collaborative Editing process.

Based on the focus description we see that users may choose to add the corresponding sources to each foci, thus the assignment of sources to foci could be,

supF1 : s1

empF1: s1

empF2: s15

121

empF3: s18
grp1F1: s8
grp2F2: s11

The above table indicates for each user the current foci created. The same **focus_id** is entered in the **user_src_t** table when the user adds that particular source to the focus just created.

event_log _t

This table (shown in **Figure 3.12**) is part of the simulator and used to record each and every event that is generated during the interaction. The events generated are propagated as messages according to a message protocol that is explained in Chapter 4. The fields in the table are used to store those messages. User interaction events are propagated as messages, which are nothing but strings with the appropriate protocol headers to identify the type of message (event) that it is. Every messages could have been stored as one long string in a single field however to exploit the power of the RDBMS whose querying capabilities can be used on the **event_log_t** table to gain better insight of the role the awareness model has played in the collaboration, the message parts are broken and recorded in separate fields. The **timestamp** field is of course self-explanatory and uniquely identifies each message (event).

However this table could be used as a record of historical awareness. Queries on this table can actually reveal the state of awareness of various users at any given time. This information can often be invaluable. In this simulator the ability for users to query is during the simulation is yet to be implemented.

3.7 Feasibility of Implementation, Engineering Issues and Implications

This section discusses the feasibility of implementing an awareness framework of the type proposed and use it in the group collaboration situations such as the Office Collaboration scenario described earlier. We also examine the engineering issues involved in the successful implementation of such an awareness framework and the implications of those issues on security, performance and some other factors.

3.7.1 Feasibility of Implementation

Related work examined in Chapter 2 gave us examples of many research efforts that indicate very successful creation of awareness propagation systems. There is sufficient evidence from the related work as well as existing technology that each of the components that make up the awareness framework can be implemented. The issues that must be dealt with, we will examine in the next section.

The awareness framework requires some sort of a medium over which the information can be propagated. An excellent ubiquitous medium or substrate is the Internet itself, which consists of not only wired but also wireless computer networks. Telecom networks have been an integral part of the Internet since its beginning and have been instrumental in its widespread penetration be it the from the early days when modems were the sole means of access for most consumers to the cellular phone networks of today. Internet and telecommunication standards have been successfully exploited to network not only various computing systems abut also other devices (PDA, cellular phones, wireless sensors, displays and more) and successfully propagate information among them.

With respect to the middleware that must run over this medium to allow various applications running on these end-systems to talk to each other, various options exist. CORBA [Schmidt 2005] [OMG 2005] and RMI- IIOP paved the way but the tremendous success of the World Wide Web has prompted the use of standards and technology such as HTTP, hyper-linking to resources, markup languages (XML and HTML), resource identification (URI and URL), to give rise to Web Services standards and technology [Singh and Huhns 2005]. Web Services have been rapidly gaining popularity due to various factors. Web Services technology comes with support for mechanisms such as large-scale event-notification that is essential for awareness framework. Other event-notification systems have been used in the past as exhibited by the iScent system [Anderson and Bouvin 2000a] [Anderson and Bouvin 2000b]. Standards such as SIP [SIP 2005] and SIMPLE [SIMPLE 2005] can also be utilized effectively to and some research efforts [Christein and Schulthess 2002] have embarked on doing so.

The aspect, which is probably most crucial, is the ability for the individual applications and groupware to interface with the awareness framework. Here again we see existing applications have been successfully integrated in the iScent project, in the AWARE system [Bardram and Hansen 2004] as well as by Mangan and others [Mangan, Borges and Werner 2004]. Commercial products such as Microsoft's LiveMeeting [Microsoft LiveMeeting 2005] integrates various application capabilities such as PowerPoint presentation, Whiteboard, and chatting. Of course here all the applications

are under the control of the same groupware system unlike the awareness framework proposed here that is meant to integrate different applications.

Thus it is possible to create awareness frameworks. However the choice of which technology and standards to use as well as how to create interfaces to various types of applications poses very crucial questions and issues that require great thought. Some of these issues that must be considered in engineering the awareness framework are discussed in the following section.

3.7.2 Engineering Issues

The effectiveness of the awareness framework depends on the following issues which impact the implementation.

3.7.2.1 Applications and Information Formats

Applications that intend to be integrated to the awareness framework must be able to generate information in formats that are accessible by users employing client applications with varying capabilities. For example, web-based video conferencing application may also allow users the ability to chat. Suppose some user is mobile and cannot access the video on his handheld device, the user can still communicate and be aware of the group's status and work via chat. This is the situation in the Office Collaboration Scenario seen earlier. Thus the onus is on the application to provide such functionality that provides users alternative means. The awareness framework and model itself cannot create content. It can only facilitate content propagation, and provide the users the ability to select and to a certain extent filter by choosing from various alternatives. For example MegaMeeting [MegaMeeting 2005] is an application that allows users the capability to chat along with video conferencing.

3.7.2.2 Awareness Map Capability

Applications must also provide the capability to display the awareness map information for users employing them. The view of the awareness model viz. the awareness map is essential for users in their selection process and ability to keep aware of various aspects beyond the information displayed by the client itself. Again depending on the capabilities the client application may display the information in different formats, for example a handheld device may display a text-only description of the awareness map. However this is still more beneficial than not having any view.

3.7.2.3 Interfaces

Both the above issues lead to the most crucial issue, which is the design of interfaces for various types of applications that need to be integrated with the awareness framework. The interface between an application and the rest of the awareness framework, determines not only what information gets propagated but also various aspects about the information i.e. the awareness information characteristics. The interface will determine the description of meta-information of the source (since meta-information essentially describes the characteristics of the information that the applications generates). The

Interface design and capabilities also has significant impact on the security and performance of the application and the awareness framework as discussed below. Related work has shown that interfaces to existing applications can be created [Mangan, Borges and Werner 2004], iScent system. Furthermore many open-source applications, both groupware such as Lucane [Lucane 2005] and tools such as Gnomemmeting [Gnomemeeting 2005] come with source code that can be extended to create interfaces to the awareness framework. Research by Du Li as part 'Collaboration Modeling and Infrastructure' projects at Texas A&M university [Li 2005] indicate that efforts to integrate applications into a collaboration framework are underway. Other examples such as CoWord [CoWord 2005] and CoPowerPoint [CoPowerPoint 2005] systems show that augmenting applications with collaborative abilities is an active area of widespread interest.

The interfaces also determine the awareness map capabilities of the application. OF course one may not expect the application to be reverse engineered to such an extent that it can display information meant for other applications. For example it may not be feasible to engineer such that a video-conferencing application be augmented to open a Word file and display it within its user interface. It would be unnecessary especially if the Word application itself can also be integrated to the awareness framework. The user may become aware of the existence of the Word file through the video-conferencing application (it will be indicated through the awareness map and even provide a link to it). The user can then access it by clicking the link or if the link is not available he or she can open Word and logon to the awareness framework to get the document. Upon doing so the fact that the document has been accessed can be propagated to the framework, and may become visible on the awareness map displayed by the video conferencing client as a notification. Such is the capability that the proposed awareness model and framework seek to provide.

3.7.2.4 Connectivity to Awareness Model

Applications can be connected to the awareness framework in different ways depending on the application type and its capabilities. For instance a client-server application may have only it server, only the client, or both the client and server connected to the awareness framework. The configuration of the connectivity will affect the generated information characteristics that are propagated to the framework.

This will also determine how other users will access this information. The awareness need not necessarily propagate the actual information content and it may only propagate the meta-information. So in order to access the information content users of the awareness framework who are not necessarily affiliated to the application generating the information will have to access it directly. If the information is o a sensitive nature then all the information security issues become significant. The configuration will determine how external users will be given access to such information.

3.7.2.5 Interaction with the Awareness Model

In the Office Collaboration Scenario, the awareness model is a central logical entity through which all the awareness information (at least the meat-information) is propagated. As the model contain all the rules by which the awareness is propagated applications generating the information i.e. the sources will have to access the model to

find our where the information should be sent to. This can imply that on each action the source application queries the awareness model. On the other hand each application could store the rules, which instruct it how to propagate the information, and it could thus do a local check rather than a remote query, which can be so much more efficient. However this implies that upon change in the rules, the awareness model must update the source application. Thus the interaction between the sources and the awareness model can take different forms. This factor also affects how security restrictions to awareness model access are implemented.

3.7.2.6 Source Specific Medium and Standards

Sources of information already exist. So will the information networks, standards and protocols that connect them. Though it may be possible to crate interfaces to the existing sources, it may not be feasible to change the networks, standards and protocols that the source applications use. In such cases it may be more practical for the awareness model and framework to provide capabilities to work with those existing systems through gateways that are created. However based on the variety of networks and systems that are being connected to create information systems, there is great promise that existing applications can communicate with the awareness framework that uses a ubiquitous medium such as the Internet.

3.7.2.7 Awareness Model Design Issues

In the example we discussed earlier an RDBMS was used to implement the awareness model. Is an RDBMS the most suitable system to design and implement an awareness model? What are the other alternatives and how does the RDBMS approach compare. Furthermore each awareness model may have its own specific design issues depending on the collaboration requirements and any one approach to implementation may not be suitable for all situations. The choice made in implementing an awareness model must ensure that the model is scalable and efficient when dealing with large of information sources. Also the meta-information from each source will be of various types, so the model must have means to record the information in a form that can represent the source characteristics accurately. Also access control policies and privileges can get complex and the model must be able to implement complex privileges. Our research here proposes the notion of an awareness model and does not enforce its implementation in any one way. However the manner in which it is implemented has various effects on the way sources of information interact with it, as well as performance and security.

3.7.2.8 Integration Process & Change Management

Change in inherent in collaboration and the awareness framework itself will undergo change when various sources are added and others removed. Furthermore sources themselves may change when newer versions are incorporated and functionality is changed (added, modified, removed). All this affects the information generated by the source. Thus there must be a process that can make the integration of new sources easily into the framework without causing any interruption or inconsistencies in the ongoing collaboration. The meta-information of the new sources must be made available to all that

are allowed to view it. Similarly when sources are removed one must ensure that there are no side effects and that links to this source are updated users interested in this source are aware of the change. Information sources shave to be uniquely identified, meta-information must be appropriately described and all security restrictions must be identified. All this calls for a sound easy to follow process for integration and change management.

3.7.3 Implications: Addressing Security Requirements

Very broadly security requirements are essential in two areas of the awareness framework.

1. **Secure Access to the Awareness Model.** It is reasonable to assume that access to the awareness model itself have to follow certain security policies. The severity of the policy would depend on the nature of work undertaken in the group collaboration. Every access to the awareness model be it from a source application that is generating information or a source applications that is consuming information must be secure.

2. **Secure Access to the Sources themselves.** As mentioned earlier users may directly access information content from the sources. In such a case each source may have its own security policy and authentication requirements. Sources may determine how users can be authenticated and have specific restrictions on what information is accessible and by whom. The privilege mechanism in the awareness model is dependent on these restriction placed by the sources and reflects them.

An important point to be noted is that the awareness model privilege mechanism is with is with respect to the entire collaboration, and is based on the overall policy of the group. This access control is dependent on the collaborating group's policies and needs of the work. This can be time-specific, i.e. one may have access to certain information for a certain period of time and later on may be denied access. However individual sources will determine who gets to access the information and this is reflected in the privilege mechanism of the awareness model. Actual authentication and enforcement of security restrictions has to be done at the source level, as it may not be feasible for the awareness model to ensure security for each source.

In accomplishing the above requirements, every element of the awareness framework viz. the interface, the medium, the middleware and the awareness model has a very key role. The interface controls access to the information source and will propagate awareness only to those users who are allowed to access the source's information. It will authenticate the users in that process. The medium connects sources to the awareness model as well as other sources directly. The middleware is responsible for meta-information transfer. As mentioned before, information content may flow directly between the sources and the user's requesting them or through the middleware that connect the sources together. This is dependent on the capability of the middleware. Irrespective of the mode of transfer, the medium and the middleware must satisfy the

requirements for secure transfer of information. Thus the implementation may have to support the necessary security protocols (such as HTTPS). This is essential since the information sources may allow access only through certain secure protocols. The awareness model itself must also implement appropriate security mechanisms to meet the requirements of secure authenticated access by users who have the appropriate credentials. The above aspects address the issue of securing the information transfer.

3.7.4 Implications: Addressing Performance Requirements

From the description of the awareness framework, we see that the framework acts as "an umbrella" over all sources and users to maintain the state of awareness in the group. This is achieved by integrating sources and propagating the information through the framework's components. This has to happen in addition to the working of the application. Consequently there is the issue of overhead caused due to the additional activity of propagating the information about various aspects of the interaction to the awareness framework. This additional overhead may cause the performance of the applications to degrade and affect the timeliness requirements of the collaboration effort as a whole. However the nature of awareness information is such that it really is *information about information* or meta-information. The volume of this information is much smaller than the information content that is generated as a consequence of the interaction of the users. Furthermore state of the art middleware technology is well equipped to propagate this small volume of information efficiently. There are special event-notification systems that are designed to cater to such propagation.

Even so the manner in which overhead can be avoided is in choosing the information that must be propagated. Not all information is necessarily useful ad the collaborating group has to decide what are the elements of awareness (attributes) that must be propagated and those that are optional. The interfaces to the applications must be design and developed carefully to enable this propagation with as little overhead as possible. Finally choice of the appropriate technology for implementing the awareness framework especially the middleware can help in ensuring acceptable performance. There is a definite tradeoff between the benefits gained by the awareness and the overhead involved. But the above steps can help in great measure.

3.8 Comparison with Related Work

We examine how this research effort compares with the related work described in Chapter 2, with respect to both the work on awareness propagation systems (frameworks) and the work on awareness models.

3.8.1 Comparison to Awareness Propagation Systems (Frameworks)

Much of the related work on awareness systems deals with the issues involved with awareness propagation among users engaged in collaboration. Noteworthy are systems such as AWARE employing the Java Context-Awareness Framework [Bardram and Hansen 2004] [Bardram 2005] that were designed to work with mobile users employing different devices and iScent framework for the WWW [Anderson and Bouvin 2000a] [Anderson and Bouvin 2000b] which is designed to integrate various applications and propagate awareness amongst them. Sources and information were integrated in some fashion to facilitate context-awareness and user filtration. These and other related research have inspired this research effort.

The emphasis of this dissertation however is on identifying the notion of the "awareness framework" as an overarching system that binds all information sources. The systems investigated in the related work do have an awareness framework, but do not state it explicitly as done here. We emphasize the importance of having such a framework especially when heterogeneous systems are used. The emphasis is on applications and groupware that are heterogeneous and not meant to work in the same group context. Another aspect is that the application being integrated into the awareness framework may continue to work independently as and is not just another type of client that is dedicated to the awareness framework. Thus an application integrated in one awareness framework could be part of different groups and even different awareness frameworks. To the best of our knowledge related work does not reveal such notions. The clear demarcation of the framework into its constituent components can help in designing and developing all the components. As sources to be used cannot be anticipated, the emphasis is on a generic framework.

Another notion introduced is that of awareness quality and the ability to tailor the awareness quality to suit one's needs. The importance of relevant awareness along with the various quality factors that define it have been well researched in related work. Various systems have been created that enable users to select voluntarily and/or involuntarily receive awareness that is effective. However the contribution here is in bring all those characteristics under one term viz. "awareness quality". More significant is the notion of trying to empower users with the ability to tailor quality in heterogeneous environments i.e. environments in which the user has little or no control on the various information characteristics (type, form, volume and frequency) of information sources that are being integrated. In the process of defining the notion of an awareness framework, the requirements for such a framework have been provided. The hope is that they can serve as an enabler in designing effective awareness systems. Awareness quality, simplified integration process and adaptability to changes in group work are of primary importance and these three factors motivate our design of an awareness model as a central logical entity that accomplishes the information integration. Once again related

work demonstrates that every system has some means to integrate the information and define the manner in which awareness is propagated. But the objective here is to characterize that aspect formally as an awareness model, and emphasize its importance. This work seeks to contribute by proposing a clear separation of the functionality of awareness systems into the four separate components (tool/groupware interface, middleware, awareness model, medium) and is intended to aid system designers and developers. The very notion that information must be integrated in using such an entity is another key difference from the previous work. The generality of this awareness framework and especially the awareness framework is the key contribution of this research effort.

3.8.2 Comparison to Research on Awareness Models

With respect to the awareness model proposed here, the work by Benford and Fahlén [Benford and Fahlén 1993] as well as Rodden [Rodden 1993] forms the foundation of this research. The fundamental concepts that were defined by them both in terms of a spatial metaphor and then extended to applications without a physical space are used here as well. However only two terms are explicitly used, viz. *focus* and *medium*. From the description of the our awareness model it is clear that the term *focus* has been used in the same sense, i.e. as the focus of attention, a concept that denotes a user's attention to one or more sources of awareness information. The term *medium* has been used to denote the physical medium of information propagation (network and its characteristics). Great importance has been placed on the concept of source, and more specifically heterogeneous sources. The emphasis in our awareness model is on the ability of users to tailor the quality of information by using the focus mechanism. Furthermore the sources of awareness information are heterogeneous. No specific use of *aura* and *nimbus* have been made. However it is obvious that the notion of aura manifests to some extent, in the form of the meta-information characteristics that each source and corresponding medium exhibit. The privilege mechanisms that enable users to propagate or make their information available to others can be considered as the nimbus of each user and its corresponding sources.

The issue of how aura collision can be detected is left to the awareness model structure itself. The awareness model structure defines through its construct of the source superset which users can be aware of information sources and their corresponding users. The issue of how users can be made aware of being in each other's foci and nimbi can be seen in terms of the notification mechanisms that have been proposed. By providing users the ability to specify the privilege mechanism and issue notifications to others the nimbus concept can be further realized. However allowing the awareness model to determine the aura collision i.e. determine who can be potentially aware of what and when, is limiting. To make this really dynamic the intelligent agents described above can be used. Intelligent agents can detect events in the collaboration group that may be of interest and inform and suggest to their human masters. This can be considered as an act of aura collision. These agents would make use of the source and medium meta-information as well as the current context of a user's activities, his /her profile of interests (that denote his spheres of activity) as well as other users' activities, all of which can change with

time. Developing such intelligent agents in itself is a extremely challenging area of research.

In determining the quality of awareness we make use of the awareness information characteristics. The notions of strength and levels of awareness are in some sense defined by these characteristics. The meta-information describing the source of the awareness information being generated and medium that propagates it convey its type, form volume and frequency. For a user with all the appropriate capabilities to receive this information, one could consider this as receiving it at full-strength. For a user who has only limited capabilities, he may be able to receive the information in some modified form (such as text-chat portion of a video-conferencing session). This can be perceived as limited in strength. Of course the source must be able to generate information in forms that can be consumed by recipients with differing capabilities. For someone with no ability to access the information is of zero strength. Of course because the strength of awareness is high does not mean that the information is relevant or is of the appropriate quality. The notion of levels of awareness has a different interpretation in our model. Here levels of awareness implies being aware of someone else's awareness. Thus concepts such as *second order look-up*, *reverse look-up* and *WYSIWIS* provide different levels of awareness. The enhanced awareness concepts are not present in the awareness models described in the related work.

The MASSIVE system by Benford and others [Benford, Bowers, Fahlén and Greenhalgh 1994] is a realization of their model for the interaction of entities ion the virtual world based on a spatial metaphor. Here human users interact with each other in a virtual world environment via user interfaces of different capabilities) (from high-end graphics and audio to limited text-only interfaces). MASSIVE uses the paradigm of distributed object interaction known as spatial trading to realize the model they propose. Distributed objects known as traders, detect the collisions between client objects, which embody their human users. Traders are brokers with which clients register their interfaces. Interfaces are the means by which client objects expose their aura nimbus and foci. The spatial metaphor model is depending on awareness being created as a result of aura collision detection. This is implemented using the spatial trading methodology where trader objects detect the collisions among clients. Here though the interfaces are heterogeneous, users are all tied to one system. This is unlike the framework proposed in this research where users can be part of the framework without being tightly coupled or tied to the same framework. Here sources of information are envisaged as being completely independent but they can interact as part of an awareness framework if they propagate information through interfaces.

When we compare our work with the Model of Modulated Awareness [Simone and Bandini 1997] as well as the 3-ontology framework [Leiva-Lobos and Covarrubias 2002] we can see that there is similarity in the need for the entities to be able to have some form of control on the awareness that is available. Just as Leiva-Lobos speaks about the need for visibility and transparence to avoid cognitive overload, we speak of tailoring the quality of awareness information. The MoMA model is more sophisticated as it defines through the reaction-diffusion metaphor of how awareness is modulated both at the emitter and at the receiver ends. The rules of reaction and diffusion of awareness can be used by intelligent agents to modulate the awareness based on change in state of entities. As opposed to this, our model itself does not seek to specify how the awareness

should be modulated. All the modulation is left to the entities that consume the awareness. The notion of field is another key feature of the MoMA model. Entities are more receptive to fields of awareness emitted by other entities and can react inflexible way as opposed to deliberately orienting their foci. This is different from spatial metaphor's focus feature that implies an orientation on part of the consumer of awareness towards a specific source of awareness. Our model gives users the ability to have multiple foci. Multiple foci can be thought of as having interest in various aspects of collaboration all around the user. This is some sense can be perceived as being surrounded by fields of awareness being emitted by various sources while at the same time the recipient defines the sensitivity of reception. When one has multiple foci then one is trying to obtain awareness of various aspects related and unrelated and in one sense creating the ability receive from various direction. One course of future work would be to incorporate the rule defined in the MoMA research into the intelligent agents support that can be built for our model. Another aspect of MoMA is that the separation of application space and the awareness pace and the notion that changes in the awareness space can be propagated to entities. The awareness propagation does not necessarily follow the publish-subscribe model where entities subscribe to events of interest whereas in MoMA entities are given more flexibility in their ability to react.

Currently the awareness model proposed in this research is based on the lines of a publish-subscribe model and this is seen in the ability to create foci, and choose sources of information. Just as the objective of this research effort has been to specify the generic elements of an awareness framework of which the awareness model is part of, it would important to extend the awareness model proposed here to be generic too. In other words we should not enforce any one methodology of awareness propagation such as publish-subscribe but allow an awareness system designer to specify their own methodology. For example, the Office Collaboration awareness model had some rules and these were defined in the awareness model created for it. Other applications may have more specific requirements for awareness propagation. For example the methodology followed by the MASSIVE system or the semantic net notion followed by GroupDesk can be used on our system. Collision detection can be carried out by agents that match the meta-information of sources and inform interested consumers. Providing flexibility to customize the awareness model not just at the level of saying which information goes to whom, but at a more basic level such as publish-subscribe, or fields of awareness as in MoMA would be a very useful and interesting extension to this research.

In conclusion the model as part of this effort emphasizes providing users an integrated view of the various information sources based on which they can voluntarily select information, or employ intelligent agents that can that can aid in the selection process. Mechanisms such as publish-subscribe (event notification) are essential fro information propagation. Metaphors such as the fields and levels of awareness are essential for selection of the information. But the more fundamental aspect is to provide one cohesive view of information sources on which the selection and propagation methodologies can be employed. This research deals with issues involve in this cohesive whole with emphasis on heterogeneous environments. In such environments the integration aspect gains greater significance. It is I in this manner that this research that it differs from the other work so far.

Chapter 4
Validation of the Awareness Model

This chapter deals with the validation of the awareness model and awareness map concept. We begin by examining what it means to evaluate awareness and follow this by specifying evaluation criteria. Next an evaluation methodology is proposed with an experimental framework. We conclude with the architectural details of the experimental framework.

4.1 Evaluation of CSCW systems

Evaluation of CSCW systems is considered a challenge due to various factors. The evaluation methodology, frameworks and processes along with the supporting tools necessary to monitor, collect and analyze various aspects has motivated research. Evaluation of the system is not complete by verifying and validating just the functional and well-known non-functional requirements such as usability, reliability, performance, and security to name a few. As these systems are used in a variety of application domains evaluating the impact of the system on the outcome and "overall quality of group work" becomes essential. This evaluation is closely tied **to the way the system is used by its users in their work**. For example, a Wiki used by a research group might be evaluated in how useful it was in providing its users the ability to be aware of work done by the group, jointly review and modify documents in a structured manner where different users have specific roles and there is some agreed upon process to the group's work. The support that the Wiki provides in notifying users of other users actions, or changes to artifacts would be used to guide the process. On the other hand an online community sharing a specific hobby, might not impose the same structure to users actions. The variety of ways a CSCW system could be used imply that evaluation must go beyond conventional criteria and metrics and tools available. The evaluation of awareness too is no easier than any other aspect of CSCW evaluation.

4.1.1 Evaluation of Awareness in CSCW

Awareness evaluation has traditionally proceeded along the dimensions such as the *type of awareness support*, the *quality of awareness support.*

4.1.1.1 Evaluating Type of Awareness Support

One could evaluate how well a groupware application or tool supports a specific type or types of awareness (A discussion of types of awareness is provided in the chapter on "Background and Related Work"). This would involve measuring the support the groupware had for keeping its users informed of all the aspects related to that type of awareness. For example, a user logged in to a virtual room would be kept aware of other users and artifacts in the same room, as well as the actions of the users on the artifacts,

their "conversations" and so on. Comparisons can be made on the basis of **how much information they propagate to their users**.

4.1.1.2 Evaluating Quality of Awareness Support

Given a groupware or tool one could evaluate **the quality of the type or types of awareness it supports**. One could evaluate the system's mechanisms of informing the user, how distracting is the mechanism, is it audio or only visual or a combination of both. Similarly other quality factors such as *relevancy*, *frequency* and *volume* could be evaluated.

We believe there is need to evaluate awareness support **with an added dimension** viz. the *impact of awareness support on work.*

4.1.1.3 Evaluating Impact of Awareness Support on Work

Considering that there is close relation between awareness and collaborative effort it is essential to evaluate in **what manner awareness support provided by a groupware or tool impacts work**. However this evaluation cannot be done in isolation **but as an extension of the above two dimensions**. Both the type of awareness supported as well as its quality factors affect the outcome of work. To study this relationship one must include the evaluation of this third dimension while evaluating the first two. Thus the group's work becomes central to determining how well the groupware supported awareness. For certain work or types of work activities that a group may undertake the awareness support provided by the system (groupware and tools) may be inadequate. In some cases the system may provide the correct type but the users may not be empowered with the ability to tailor the quality of awareness. In other cases the impact on work may not be as positive as one expects. We elaborate this notion with respect to evaluation of the awareness model and map in the following section.

4.2 Evaluating the Awareness Model and Map

The primary purpose of the awareness model is to enable awareness to be propagated among groups working using a variety of applications. An important point to be noted is that various applications in use may each be designed to support a specific type or more than one type of awareness. For instance an email system support awareness of all the messages sent and received by a user with information such as priority of the message, when it was accessed by the recipient and so on. Secondly such a system may have limitations on the quality of awareness it provides its users. Users may have limited scope in choosing the mode of delivery, frequency, volume and type of information they need to know. Assuming various such systems are integrated in an awareness framework, the awareness model empowers its users with the following ability:

1. Select the type of awareness by searching and or browsing the available information sources (Done manually or by an agent)

2. Tailor the quality of awareness information received by choosing information that is relevant and of the appropriate volume and frequency, obtrusiveness, as well as establishing privacy. (Done manually or by an agent)

3. Providing enhanced awareness in terms of knowing who else is aware of the same information i.e. be aware of other user's level of awareness and so on.

In other words the awareness model and map act as an enhanced active directory of awareness information sources. Description of awareness sources and support for user actions such as those mentioned above are its primary features. **This support is provided to improve the work of the group**. Thus evaluating the awareness model and map must evaluate **the impact of its awareness support viz. the above features on group work**. In other words the validation of the awareness map and model deals with **finding out how the functionality it provides, impacts the outcome and quality of group work**.

To evaluate the impact of awareness on work would require some means to identify how awareness influences work and then design a framework **to identify such points in group work**. The efficacy of the awareness model would then be evaluated **at these points in measuring how well it supported group work viz. impact on group work.**

4.3 Influence of Awareness on Group Work

Awareness improves the ability of the group to perform tasks towards their work goals more effectively. Broadly speaking awareness of information related to a user's sphere of activities could be used in the following ways:

a. Information is used in performing tasks towards completing group's work.
b. Information is used in planning and coordination (Planning and coordination can be considered as tasks towards the goal).
c. Information may also be used to enhance the quality of work. For example knowing that a newer version of software has enhanced features may result in creation of better artifacts.
d. Information may be used to and improve the work environment and process. For instance some offices post the birthdays of all employees on the group calendar. This encourages employees to greet each other on their birthdays and build a rapport, which can positively influence the group's work.
e. Information may also be used to enhance ones own work related skills but also develop qualities that implicitly influence work. For instance reading a book on managing one's time better can lead to a developing better habits overall, reduce stress, which influence work positively.

A detailed discussion of the impact of awareness on human beings can be found in works devoted to the subject. We are more concerned with the visible impacts of awareness on the individual and thereby the group's work.

4.4 Evaluating the Impact of Awareness Quality on Group Work

Information can be sought by the user i.e. the user can seek information when necessary. On the other hand user can also be informed involuntarily. This adds the key notion of time in awareness. At a stage in one's task an individual may seek some information and may receive it promptly or may receive it later, and in some cases never. The timeliness affects the outcome of work. Similarly information may be received sooner than needed, which can be stored away (in human and /or artificial memory – electronic and non-electronic forms) and used later. We see often that information received sooner is used in being proactive and valuable in planning and coordination.

At every point in work, when information is received by a user (voluntarily sought or otherwise), the type of information received, the amount, and the mode of delivery, which determines how distracting it was, all influence the outcome of the task. If the information received meets all the requirements of the task it can be completed and work can proceed to the next stage. If not the task may not be completed as per requirements, or alternative paths of action may be chosen towards the goal. This could result in inefficient execution due to increased number of steps, increased utilization of resources (depending on their availability) and others. On the other hand acquiring the appropriate information may prevent such inefficiencies.

By acting as a directory for awareness information, the awareness model aims to empower users to choose the quality of awareness to enhance their work. The directory not only shows what is available but the meta-information it provides about each source indicates the quality of awareness available; in other words *how well we are connected and how good the source of information is*.

This ability that the users have, can help improve work by making users choose alternative paths of execution that are more efficient, being more proactive in their work by planning and coordination. Consequently the overall work could involve fewer tasks, less wastage, redundancy in actions, roles, as well as efficient resource allocation, utilization.

Based on our discussion above we propose that the evaluation of the awareness model **must include** the following: *How does the "directory" and description of information sources and ability to choose the quality of awareness affect the choices users make in accomplishing their tasks towards their goal.* Specifically it can be broken down into answering:

At each point in the user's sequence of tasks where information is received by the user:
1. Does knowledge of these awareness sources affect the normal execution of the task at that point?

2. If awareness affects the normal path then how do users modify or change their execution paths towards their goals when they are aware of these sources and are able to access the information? What sorts of actions are taken in the alternate paths?

3. When empowered with the ability to choose the quality of awareness how does the user's actions differ from the normal path? In other words we examine how the ability to ascertain and select the form, volume, frequency and access control privileges affects the sequence of tasks and the ultimate goal.

4. When information arrives sooner than anticipated or later than required, how does the sequence of tasks get affected? How does awareness model impact such situations?

We are evaluating the **impact of user actions with respect to the awareness model on the overall quality of work.**

Specifically these actions are:
1. Searching and Browsing for awareness sources.
2. Creation of foci and addition of new sources.
3. Modifying foci by adding and deleting information sources and deleting foci that are no longer relevant.
4. Tailoring the quality of awareness information received by choosing information sources and corresponding medium that provide the appropriate type, form, volume and frequency of information.
5. Ascertain status of other users and their current activities and initiate communication.
6. Use privileges to enforce privacy and access control to awareness propagation
7. Based on privileges obtain enhanced form of awareness such as
   a. **Second-Order (N-Order) Lookup**: If you are in my focus then I see all your foci. This can be extended to multiple levels (*N-Orders*)

   b. **Reverse-Lookup:** Who are all the users that can see my information sources viz. in whose foci am I in?

   c. **What I See is What You See (WISWYS):** Who else is seeing what I see?

The above aspects affecting work with respect to awareness can be considered the variables in each work scenario. To monitor and measure the impact of the awareness model and these variables we propose and experimental framework with the corresponding terminology. Before examining the details of the experimental framework we discuss our choice of validation methodology.

4.4.1 Defining the Quality of Group Work

The awareness models utility in propagating awareness is evaluated with respect to the *quality of group work*. Awareness plays a key role in the outcome and quality of work. Effective awareness about various aspects of the group effort is essential to improve quality of group work. We mention the criteria for quality of group work here:
1. The required outcomes are achieved. All requirements are met. To ensure this a common understanding of the expected outcome is necessary in a group.

2. Work is accomplished in a timely fashion. Delays within the acceptable limits or none at all depending on the type of work. Certain deadlines cannot be relaxed.

3. Efficient resource allocation and utilization.

4. Seamless adaptability to change in requirements.

5. Unanticipated benefits such as:
    a. Proactive planning and coordination for future.
    b. Just in time optimization of work, dynamics scheduling and coordination
    c. Opportunistic Collaboration
    d. Enhanced quality of work product

The role that awareness plays in ensuring the above is what is to be evaluated i.e. more specifically **the awareness model and map's role in ensuring the above**.

The aim is to study the impact of the awareness model's functionality and features in enabling the above. Thus the evaluation involves comparing how the above factors are affected with an awareness model as opposed to without one. This requires establishing a method to identify if such a feature of the awareness model aided or enhanced the above quality of work criteria.

4.5 Validation Methodology

A choice of validation methodologies for the awareness model and map are as follows

4.5.1 Implementation and Testing of Prototype

This involves creating a proof of concept implementation of the awareness model and map. The implementation would have to be deployed in a real-world group work environment and tested using the criteria described in the section above. Such testing would require that suitable mechanisms be employed to collect metrics of how users accessed and used the awareness model and map during the course of their work. A complete picture of the overall impact the awareness model and map had on group work would be obtained by collating the results of all user's interactions with the awareness model and map would give the complete picture of the impact of the awareness model and map on the quality of work.

4.5.2 Simulation

Verifying and validating a system's dynamics through simulations is a popular approach and an alternative to testing an actual implementation. A model of the system is created with the awareness model and map as essential components. The model includes abstractions of human users, applications, groupware, and infrastructure that represent group work. User's behavior towards accomplishing the group's work and consequently their interaction with the awareness model and map is simulated to ascertain the system's dynamics. Similar to a full-fledged implementation simulation data is collected along the above criteria. The data collected must indicate the impact of using the awareness model and map on the user's behavior.

4.5.3 Comparison of Full-Implementation and Simulation

A comparison of the two approaches is useful in determining the strategy for evaluation. A key requirement for evaluation is that adequate amount of data needs to be collected. The criteria for comparison are as follows:

4.5.3.1 Distraction, User Effort and Amount of Data Collected

Testing a full implementation is desirable however great care must be taken to ensure adequate user participation. Obtrusive means of data collection that distract the users from their tasks as well as extra effort on the part of users to collect data can be some of the pitfalls to be avoided. For instance if users had to spend time and effort recording data based on their interactions with the awareness model then they might be less inclined to participate in the evaluation. Distractions may prevent users from their normal course of actions. In other words the system's normal execution should not be perturbed by the requirement to record data.

Simulations can be run a number of times and sufficient amount of data can be collected. With the human element absent system perturbation is not an issue.

4.5.3.2 Reproducibility of Results

It is difficult to reproduce results of human usage and behavior in actual testing involving humans. Even if the same situations are reproduced, human actions may be different.

Simulations can be run a number of times with the same set of parameters to verify the results. Parameters can be carefully changed to create small changes in situations.

4.5.3.3 Variability in Human Behavior

However an aspect that closely follows the previous argument is the fact that variability in human behavior may be desirable in evaluation. The unpredictability of human actions with respect to some situations can actually provide valuable data on how the system under evaluation has fared. A positive aspect of the real testing is that user behavior need not be predicted

With regard to simulations, this behavior has to be carefully encoded. The difficulty is in anticipating the possible behaviors that occur most often. The second aspect that is inherently true for simulations especially involving social experiments is that incorrectly designed and developed simulation will provide incorrect results. The validity of the results themselves are difficult to detect.

4.5.3.4 Time and Resources

Implementing and testing a full system successfully requires time and resources such as software, hardware infrastructure and more importantly the human users necessary to test the system.

Simulations can be programmed much quickly with the use of various agent-based simulation environments.

4.5.4 A Hybrid Approach to Evaluation

Our approach is a hybrid of both the above traditional approaches. The awareness model is implemented fully. However the elements of the awareness framework such as the medium, and the applications (sources of information) are simulated. Human users are required to interact with this system, which is **emulating** an awareness model and simulating the rest of the awareness framework. Their interaction will proceed according to pre-defined scenarios designed to evaluate the awareness model. However users are free to act in a natural manner with the applications. Thereby the advantages of actual human involvement and interaction with an actual awareness model implementation are realized. On the other hand the advantages of simulation such as reproducibility in reenacting the pre-defined scenarios, automating some aspect of the human behavior by using agent programs, as well as saving in time and resources is obtained. The entire

evaluation is thus a mixture of actual working elements and part simulation. **Figure 4.1** below illustrates our approach.

User Interaction Simulates User Tasks in Collaboration

These include

1. Interactions with various applications that run on different types environments (hosts,. devices, nmetworks)

2. User Actions with the Awareness Model and Awareness Map in tailoring the Quality of Awareness

User 1

User 2

User 3

Awareness Simulator

-Captures User Interactions

-Emulates Awareness Model Behavior
   -Updates Awareness Model (AM) based on user actions
   -Propagates Awareness based on AM
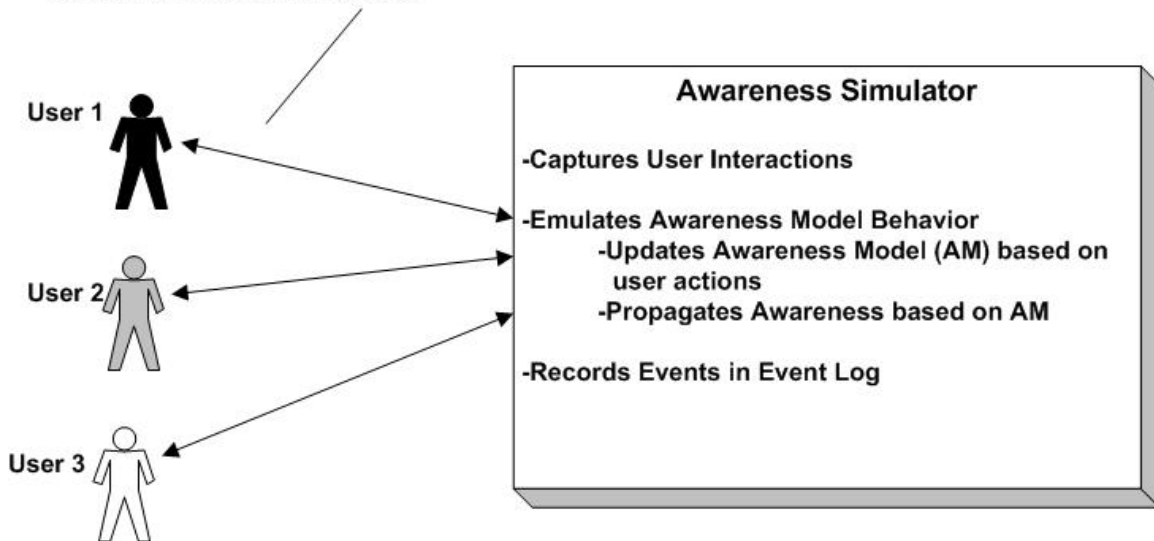
-Records Events in Event Log

Figure 4.1 Awareness Simulator

4.6 Experimental Framework

The aim of validation is to simulate scenarios, which demonstrate the utility of the awareness model in group work. This involves the ability to create all relevant elements of group work including the environment and its users. All the activities of the group such as tasks, interactions of the users with the applications, communications among the members, accesses to artifacts to name a few must be simulated. These activities must be analyzed both in the presence of the awareness model and when it is absent. This is essential to detect the improvements in quality of group work due to the awareness model. Apart from this comparison we are also interested in the manner the awareness model impacts work when various conditions change such as information arriving sooner than one uses or later than one requires, changes in users location, unanticipated requirements, changes in the environment (application and systems) and other variables. To create, simulate and analyze the above we propose the following experimental framework. The structure and terminology associated are as follows:

4.6.1 Scenario

A *scenario* is a description of all relevant elements of a group's collaborative effort. The description is in natural language. The overall objectives of the group as well as description of the actors, their environment in terms of applications and systems and their actions to achieve the goals are summarized in the scenario description.
Each scenario can be further described by the following elements.

4.6.1.1 Work Process Model

The group's collaborative effort is described in a ***work process model* (WPM)**. A work process model will consist of the series of tasks required to accomplish the goal from each user's perspective. The series of tasks will include the temporality and ordering. All activities of each user towards the goal are modeled in the work process model. This includes access to information, and communication among users.

The most important aspect of the model is that it does not specify the environmental details of the group. For instance if a user accesses a database through a web-based portal the details such as authentication, filling of the query form and submission are all modeled as "get information" or "seek information". If a user communicates with another through telephone, this aspect is not modeled and only the act that user sends message is modeled. This is deliberate and the rational is as follows:

Our aim is to compare various simulation runs (both with and without awareness model by varying simulation conditions as described above). This implies that there needs to be **a standard set of actions** that can be simulated irrespective of the work environment to act as reference set that can be reproduced. Especially as the simulation involves human intervention, **it is essential to ensure that users actions are towards accomplishing goals of the group effort**. In that sense the work process model **provides a map or script** of the of the generic work process that each user should follow. Each simulation run may have a different work environment and the details may differ but the actions essentially remain the same.

The actual work environments affect the details of the actions in terms of how it is carried out, how long it takes, the constraints and so on. Such variance is welcome and is in fact a crucial in comparing how one wok environment fared versus the others in its effect on group work. Given a particular work environment the information generated by various information sources is integrated in the corresponding awareness model. The basic set of actions scripted according to the work process is enacted or simulated on this work environment and awareness model by users. The results of the simulation can then be compared with each other thereby providing insight into the effectiveness of the awareness model.

A work process model will be represented as a ***work process model graph*** (WPMG) and described in detail in the next section. But before we examine its details it is necessary to be familiar with the rest of the terms we use.

4.6.1.2 Work Environment

The ***work environment*** **(WE)** is a description of all the elements in the scenario that are used in the group's collaboration. The actual applications an their functionality, operating systems, individual hosts, and network infrastructure are part of this description. The connectivity and capabilities of each element are also specified. These attributes of each of the above elements especially the capabilities and connectivity of all these elements **provide the constraints** for the group's effort. Conceptually this can be thought of an as **an instantiation** of a collaboration environment that is necessary to accomplish a work process.

A single group work process may be realized by many different work environments each with varying capabilities. However different work environments will determine the manner in which the same work process gets accomplished and therefore a significant impact on the corresponding quality of the group work.

The awareness information sources and the awareness attributes they generate come from this set of work environment elements. As the awareness model integrates the elements of the work environment, its structure in terms of the connectivity and privileges is determined by the work environment. The awareness model is now the directory of all these sources

4.6.1.3 Work Practice

All the activities and their consequences in an actual work environment constitute what we term as a ***work practice***. The term work practice is defined as *"what people actually do"* in an environment while accomplishing their work [Sierhuis and Clancey 2002]. As opposed to abstracting human behavior in terms of processes and tasks that hide the details, work practice actually describes all the steps people take in the course of their work. For instance the incidental interactions that two co-workers share when situated in adjoining cubicles can often play a part in the outcome of their work. We often react to information from our coworkers in the next cubicle. This information can be distracting and irrelevant or sometimes may actually speed up the work process by providing us a more efficient way of accomplishing the work.

Such details are ignored in traditional process descriptions. In our work process model graph we do not model these details. We enact the work process model on the work environment and the result is what we consider a work practice. Thus work practice

is the result of a user's interaction and this interaction is a consequence of the work environment that the user located in. However this practice is essential since it highlights the aspect of awareness in a co-located workspace and its impact on the overall goal. Such interactions are recorded and analyzed in out simulation.

The emphasis of work practice however is to understand the behavior of people and their interactions with other humans, applications, systems and artifacts which can impact work. In our evaluation methodology we define work practice as *"the actions that users take in a work environment towards accomplishing the work process"* i.e. it is the output of the simulation run. The variance in terms of incidental communications with co-workers, distractions, use of certain tools, adoption of certain work habits, as well as the constraints posed by the work environment all make up the work practice. As awareness is an integral part of the work practice, monitoring and analyzing work practice is the most crucial step in the evaluation of the role of awareness and more specifically the awareness model itself. Corresponding to the work process model graph, which is a representation of the work process model, we will represent a work practice with a **work practice model graph** (**WPrMG**). The details of these representations will be seen in the next section.

4.6.1.4 Simulation Run

A single **simulation run** is the sequence of events that occur when the work process is enacted on a given work environment. In our multi-player simulation each user will try to accomplish the goals as per the steps in the work process. The exact actions taken however will depend on the capabilities of the work environment he or she is working with. For instance if the step in the work process involves communicating some information to another user, the first user will do so using which ever application is available. These could be any among the ones such as email, telephone, instant messaging or fax. **The act of communication will now be constrained by the capabilities and type of application as well as the connectivity it provides and so will be the outcome.** All events occurring in the simulation run are recorded in a log known as the *event log*. Analysis of the event log will reveal the impact of how the work process progressed in the work environment. In other words it reveals the work practice of the group. Most importantly the events will reveals how the use of the awareness model the group made an impact on the outcome of the group's work. In the next section on simulation metrics we describe the types of events recorded and the analysis performed on them.

The entire process of simulation that we are using is illustrated in the **Figure 4.2** below. This shows that in the real world work processes are enacted on a work environment and this forms work practice. In a simulation work environment is modeled along with the awareness model. Human users simulate the work process on this model and the results are collected in an event log. Analysis of this log reveals the work practice and awareness model impact.
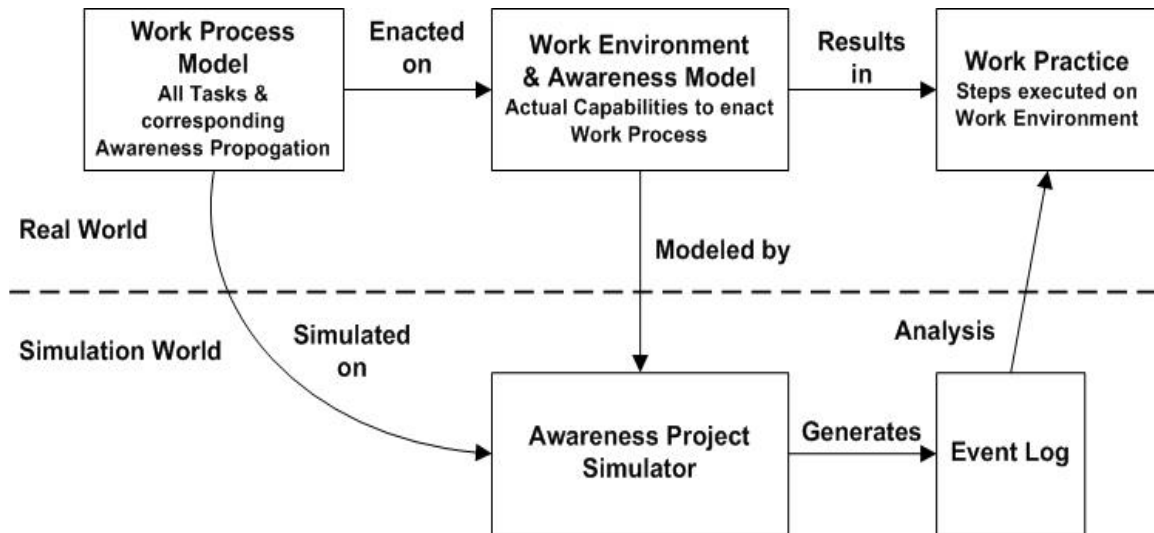
Figure 4.2 Awareness Model Evaluation Process

Before we describe the analysis of the simulation outputs and the metrics collected it is necessary to examine the work process model itself as it provides a basis to understand the simulation.

4.6.2 Work Process Model and Work Process Model Graph

The objective of the simulation is to compare the impact of the awareness model (use of its functionality) on the group's work. This implies there must be a way to represent an "ideal" work process so that the results of the simulation can be compared to that reference. This helps us realize how close the results of the simulation are to understand how close it is to the desirable work process. This ideal work process is the one described by the work process model.

The work process model is based on the following ideas. awareness information in a group's work process is the information about all aspects relevant to the group's work process. Awareness is continuous in the sense that we are always being made aware of various things happening around us through all our senses. Sometimes information that we are being made aware of is not relevant to our sphere of activity and often it is. Sometimes we search and retrieve information through various means (electronic and non-electronic). This process could be with external entities such as people, systems or could be the act of recollection and introspection. Sometimes the information is brought to our attention again in a variety of ways. We process this information and depending on the relevance (type and timing) of the information we may use it immediately, or store it away for later (use again on external media or in our memory).

Thus modeling *where*, *when* and *how* information affects our work is not a trivial task. Unlike systems that have explicit points of information access in their working, humans are constantly aware.

However for the purpose of this study there must be a means to model **how information affects the work process**. We use a set of symbols to model our work process model and the resultant representation is a work process model graph. The symbols used in our models are as follows:

4.6.2.1 Awareness Information Point

We use the notion that in order to perform some task a certain amount of information is needed. The manner this information becomes available may vary as described above. Furthermore the timing may vary. Also form, volume and frequency may vary. The aim of the simulation is to study the impact of those factors. As we stated above clearly identifying how all these aspect of awareness information affect a work process is difficult. We attempt to do so by examining the role of information. Information with certain characteristics is essential to accomplish some task. Without the information the task does not get done or gets done partially. Also if the information is not available alternative paths may be taken by the user to acquire that information. The task may later be completed when all information points are available. If the information is not available at all the task may never be accomplished. All the above are commonly observed in work practice. In our model we identify before each task all the information that is needed to accomplish it. The entire work process is modeled and such points for each task are identified. Each of these points is called an ***awareness information point* (AIP).** Each AIP has a unique identifier. The work process model consists of AIPs and corresponding tasks and provides the basic map of the work process in terms of the awareness information requirements. **Figure 4.3** shows the graphical symbols. An AIP represented as shown and annotated with the information needed at that point. A task is

represented as shown and annotated with the task description. Arrows connect tasks and AIPs forming a work process graph. Each work process graph has a start and end symbol. The arrow after the AIP connects it to the task that can be accomplished if all the information needed at the AIP is available. Thus it represents the path of successful completion of the task. The dotted arrows arriving at the AIP represent the different pieces of information arriving that make up the AIP. When the information obtained matches the AIP the task can be accomplished successfully. This is the basic premise of the model. By the same token if the information does not match as mentioned above either the task may be abandoned and alternative paths could be taken. These alternative paths could be towards acquiring the information or relate to completely different tasks. During the course of those alternative paths if all the information required for the original tasks is received, depending on the current state and context of the user the task may get accomplished.

A plain AIP symbol represents a logical point in the work process where information has been sought and needs to be made available. Thus it is a point in work where information is sought and the user is acquires it. As it is a logical point in time the delays due to the applications, systems and environment in acquiring the information are not modeled. However in some cases the user has to wait for perceptible amount of time to acquire the information. In such cases the act of waiting itself is modeled as a task and the acquiring of information occurs while waiting. This is modeled as follows: There are also situations where a user may be acquiring information while performing some task. This is due to the multi-sensory capabilities that humans possess. For instance over hearing a co-worker while typing a document. If such information is applicable to the work at hand then superimposing an AIP on a Task symbol represents its acquisition. Generally the first case with a plain AIP is applicable to situation where information is sought voluntarily. The second applies to situations where information is made available involuntarily to the user. However there is no such hard demarcation. It is perfectly natural that information being made available to the user while she was working on another task was due to a request made sometime before.

To simplify the model the tasks of actually absorbing the information, such as cognition, comprehension and storage in user memory are not modeled as explicit tasks. It is assumed that these occur at all AIP. Another key task is evaluation of the incoming information at the AIP to the required information. This is also assumed to occur at an AIP implicitly. The consequences of the evaluation however are represented as the subsequent tasks.
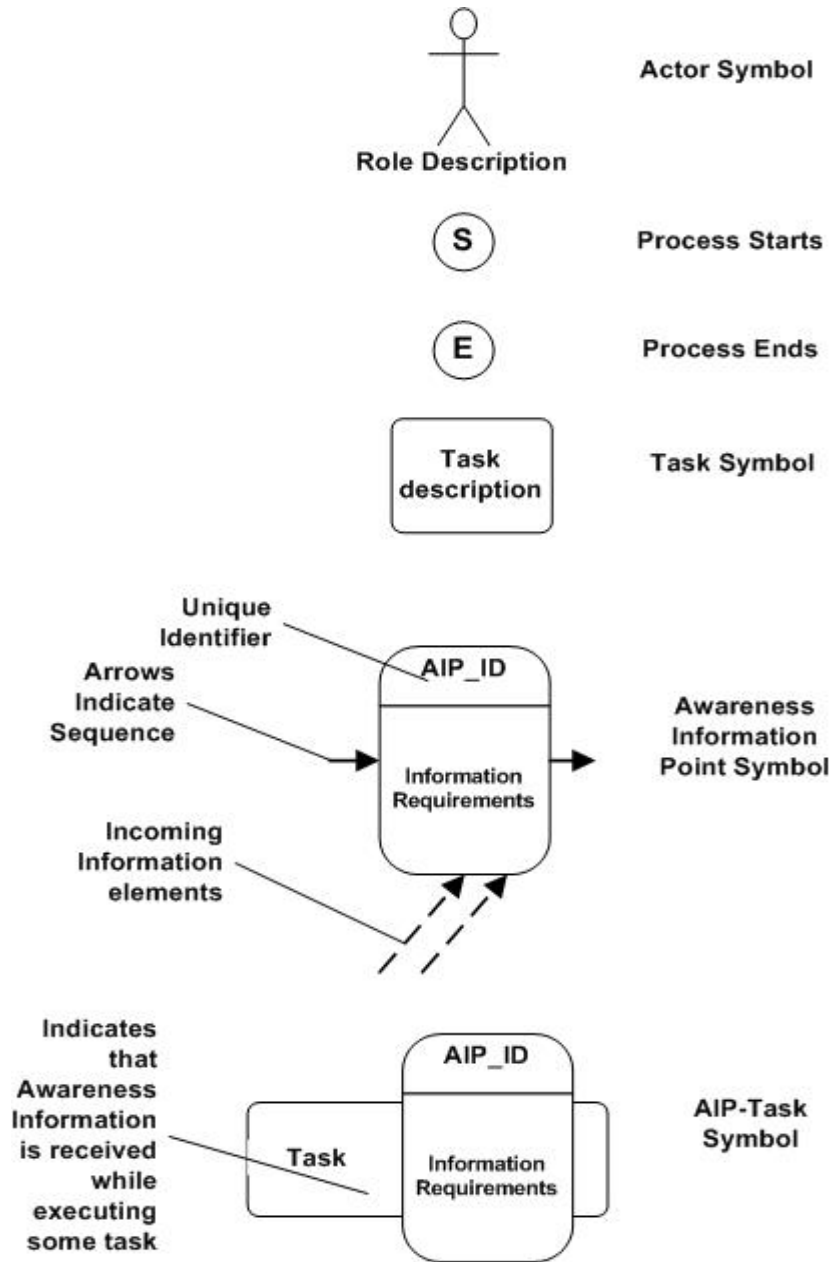
Figure 4.3 Work Process Model Symbols

4.6.2.2 Task

A *task* symbol depicts any task that is part of the work process. The symbol is annotated with the description of the task. In the generic work process model the details of the environment are abstracted. For instance a search and retrieval of information is modeled without the details of how the search is carried out. The work practice model has the details. Forking from a point and joining later represent concurrent tasks. If a task results in propagating information to an AIP then an arrow represents connect the task to the AIP with the appropriate annotation of the information. This is applicable to AIPs

superimposed on task symbols too that denote that in executing the task information is being aware.

A work process model that consists of all AIPs and tasks required to accomplish the goal without the details of how the tasks are actually carried out is generic and forms the baseline for comparison of simulation runs. The primary objective of the work process model in depicting the tasks **is to identify clearly the information flow in the process.** Awareness of information being the theme of the simulation it is essential to identify all aspects of information flow that occur. This is done irrespective of the work Environment. The work process model defines a basic set of tasks and corresponding information flows. Simulation runs vary based on the work Environments and other factors that constitute a work practice (discussed below). However they follow the same work process. The AIPs denote where information is needed. Comparisons can be made at these points between simulation runs.

The figures below illustrate the concepts of work process model and work process graph we have described so far. We use the Document Review work process, which was described in the "Office Collaboration Scenario". To accomplish the overall goals of the collaborative effort various tasks have to be accomplished by the members of the group. We model the tasks of for each member as a work process graph. The overall work process graph (model) is a composite of each user's work process graph. A work process graph is created for each user. **Figure 4.4** illustrates the work process graph for a single Reviewer and the Editor. The process of document review is set to consist of a reviewer and an editor. Both roles access a page of a document. The reviewer expresses his comments and the editor records them. This sequence is depicted in their individual work process graphs. The initial AIPs for each denote acquiring the first page. The next task denotes communicating the suggestions, and then he waits. Meanwhile the editor is waiting while for the suggestions. He receives the suggestions of the reviewer while waiting. This is represented by an AIP on the wait task symbol. Upon getting the suggestions the editor records them and this is depicted in the next task symbol. The next task is to communicate that he is done and ready for the next page. The reviewer who was waiting receives this information. Again the AIP on his wait task represents this aspect. The process continues till all the pages have been reviewed and then terminates at the end symbol. Thus the AIPs, tasks and communication lines represent the awareness information propagation among the users during the process. The next figure (**Figure 4.5**) extends the same process to *N* reviewers. A typical document review has a negotiation phase but that has been overlooked in our model to avoid obscuring the key notions of awareness propagation in the model. Thus **Figure 4.5** represents the work process model for the entire document process involving all users. This is a generic work process model. This work process could be enacted in various work environments such as a co-located face-to-face meeting, or be distributed and use a video conferencing or web-conferencing infrastructure of various capabilities. Based on the environment various actual actions may occur such being interrupted by other actors who are not part of the process, applications failing, delays, and so on. These form the work practice.
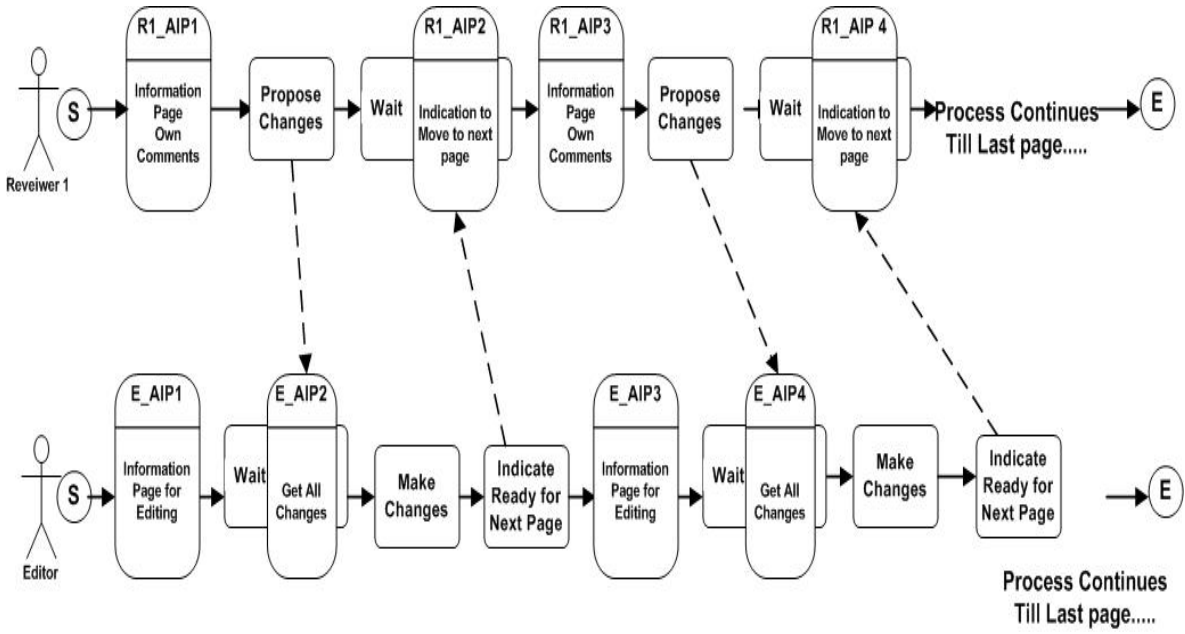
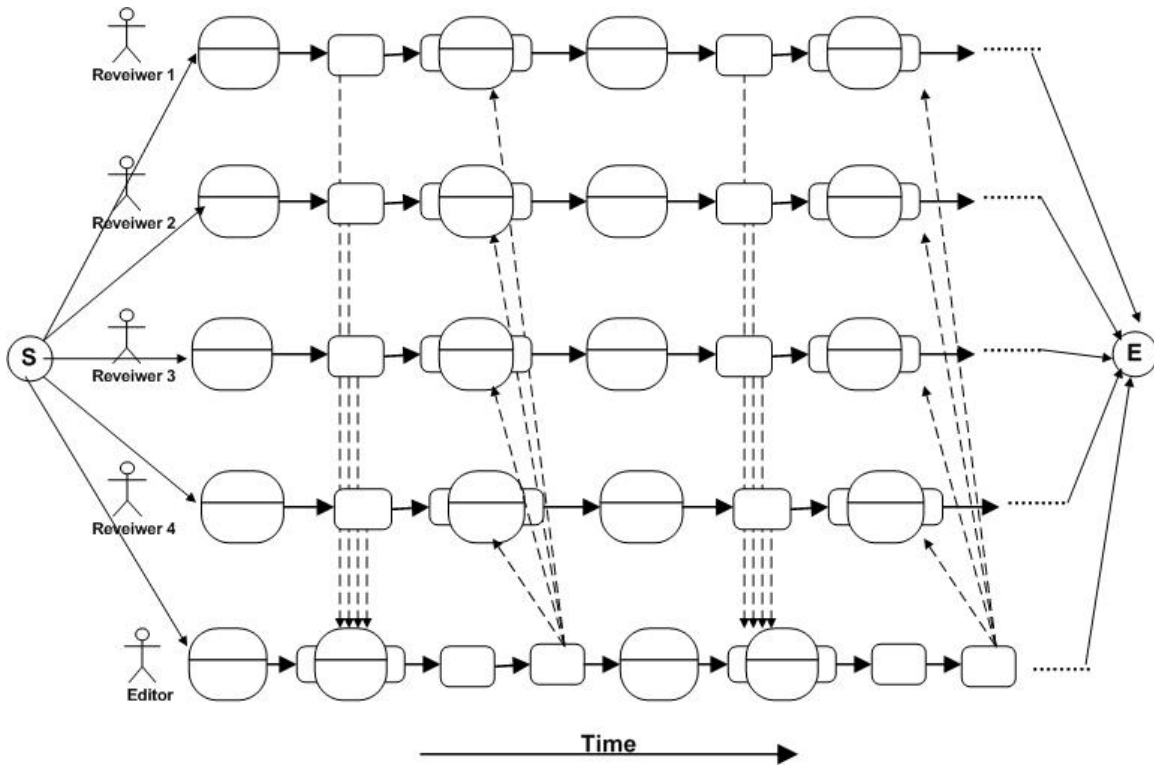**Figure 4.4 Document Review Work Process Model Graph**

Figure 4.5 Complete Document Review Work Process Model Graph

150

An example of work practice is as follows: One of the reviewers may have to leave the meeting due to some unexpected work at the office. This is depicted in **Figure 4.6**. This is a path of execution taken due to some constraint such as an interruption. The normal course of action is seen and as can be inferred the interrupted reviewer will miss the process. Consequently his input will not be recorded which can be detrimental to the process. Assuming he was able to somehow be aware of the current status of the review process, he may be able to convey his comments when everyone reaches the particular pages. Let us consider that the users were collaborating using a typical video conferencing application. Some video conferencing infrastructures are not capable of conveying information to mobile users on limited bandwidth and small screen. Thus the user may not be able to view the documents themselves or the faces of his co-reviewers. However there may be ways he could still chat or exchange messages with them using his hand-held device (PDA or Cell phone) not permit the users to roam. This implies that as soon as he leaves his video conferencing client and switches to a hand-held he should be aware of the video conferencing server and be able to connect to it. Not only should be able to connect but he should also be able to specify that he wants to receive only messages as opposed to complete images. The video conferencing infrastructure itself must be able to support such clients. Thus there is a requirement to be able to connect as well as to be able to find out about the ability to connect and acquire information. The awareness model provides this ability. Correspondingly his co-reviewers should themselves be aware that one of the users is now no longer at his desk and cannot see them, but there is still some way they can maintain communication. The awareness model also can facilitate this since it keeps track of the users current status. The concepts involved in a work practice are described next.
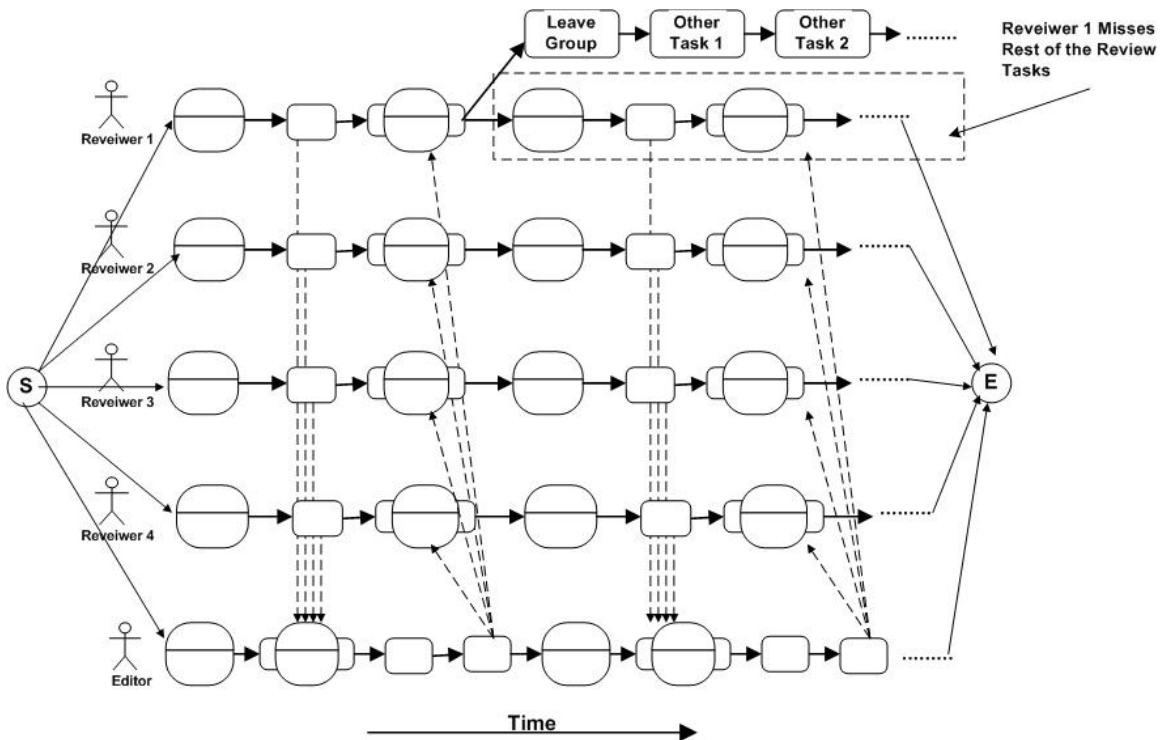

Figure 4.6 Document Review Work Process: Reviewer Leaves Meeting

151

4.6.2.3 *"Mismatches" and Recording Alternative Courses of Action*

Our simulation involves enacting the work process models such as the one described above on a work environment modeled in the simulator. An awareness model is created in the simulator that integrates the information source in the work environment. Users enact the work process model by performing the tasks using the simulated applications available to them. They make use of the awareness model and map to identify the sources they receive information from and can interact with. This interaction is based on the needs of the work process. The work environment constrains the flexibility and extent to which they can interact. The results of the simulation are recorded in an event log in the simulator. Analysis of the event log gives us the work practice and provides insight into the impact of the awareness model on that work process.

In order to gauge the impact of the awareness model we have to identify the points in the work practice where the awareness model made a difference in the ability to propagate effective awareness. Based on this we can compare various simulation runs for a particular work environment and awareness model. This comparison can be extended to different work environments and corresponding models. For this purpose we introduce the concept of a "***Mismatch***".

Mismatch: *When the AIP information requirements do not match the information available then there is a "Mismatch".* We denote AIPs with a mismatch with an M on the AIP. This is a key metric recorded for analysis. Users will record a " mismatch" whenever they perceive that the information they receive does not match the information they require. After each AIP there is a solid horizontal arrow connecting it to the next task along the path of the normal course of action taken if the information required is available. However if the information does not match the requirements the user may take alternative course of action and perform other tasks. Arrows that emanate from the AIP at an angle denote this. **Figure 4.7** illustrates this concept. This is a representation of the work practice with the normal execution path that should have been followed.
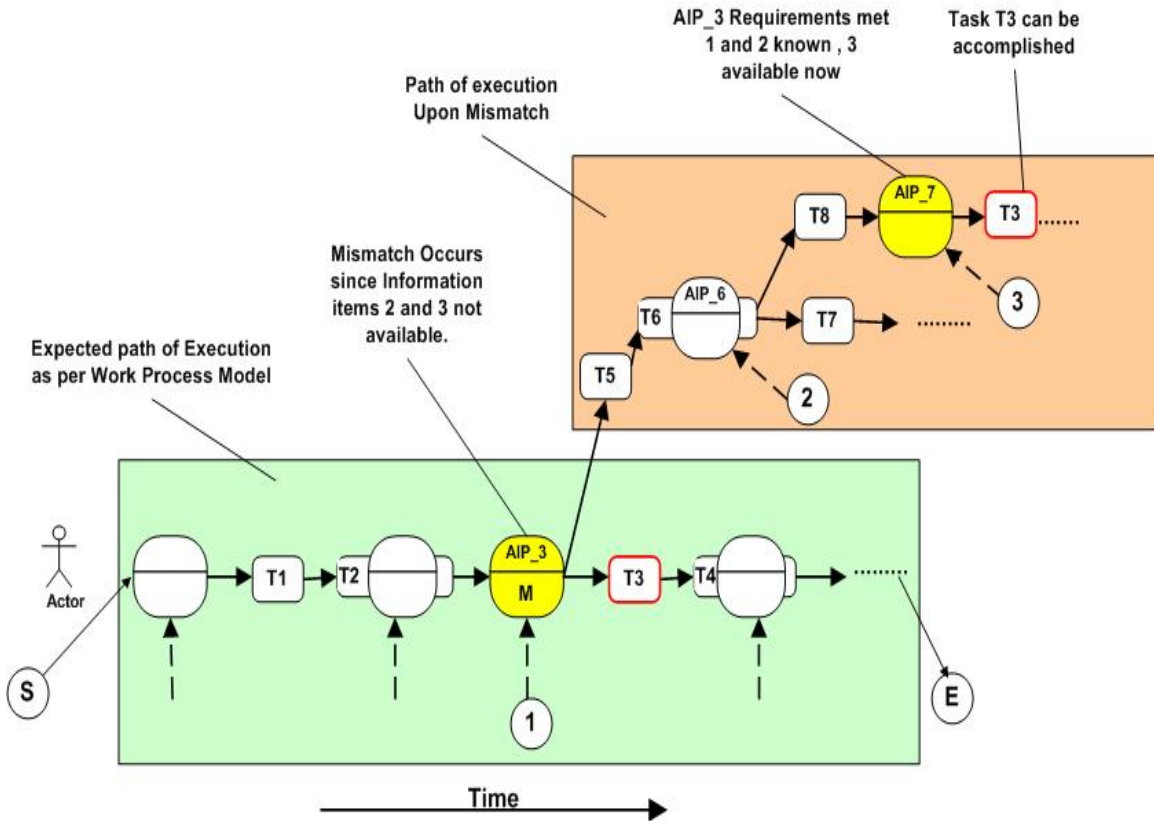
Figure 4.7 Notion of Awareness Information "Mismatch"

Why is a normal course of action modified in work practice? Various reasons can be attributed to such a change in execution plan. The key factors are:

1. The work Environment: The application and systems available as well as the connectivity automatically set constraints on what the user can do and how he does it.
2. The degree of mismatch between the information available and information required results in mismatches. The range of information availability can be from absolutely nothing to some meta-information, to the getting an exact match. Depending on the degree of mismatch the user may choose some course of action that is seen appropriate at that point in time.
3. There may be other conditions not part of the planned work process that the user needs to satisfy. Generally these are unanticipated. Even if they are planned for in advance they may be unavoidable. For instance in our document review scenario one or more reviewers may have to attend to a meeting that cannot could not be scheduled at another time. Such interruptions may involve changes in location in addition to being away from the application that he currently collaborating with.
4. The very nature of the user's work may require him to execute other tasks concurrently with the following work process. These tasks are not part of the work process but do influence the outcome of this process as they compete for a primary resource viz. the user.

5. The individual user may chose to take certain course of action due to some prior knowledge about the work process and past experience. For example the user may deliberately decide to postpone the task so some other conditions can be met. There could be external needs such as lack of time, which may force the user to postpone the task. Thus the users current context of which the work process is just one part of, determines his or her actions.

The alternative courses of actions taken are essential to the simulation and are recorded. Even though various alternative courses of action can be taken the rational with, which the users simulate their actions will be towards accomplishing their own and the group's goals. However all actions taken determine how soon the goals are accomplished, if all parts are complete as per the requirements and if other quality factors of the work are satisfied. In some case the goals may not be accomplished at all. However taking an alternate course of action does not imply that the task gets abandoned always. When an alternate path is chosen due to a mismatch, the information required for the task that could not be accomplished may arrive later. At that point the user processes the information and if all conditions are met the task in question may get done. However the total number of steps taken to accomplish the task increase since the steps taken to retrieve information for the task has to be included in this count. As the total time taken increases correspondingly, the latency to get the task done increases. The latency involves all the time the task could not get done till the information was available. In our simulation we measure the latency by the number of steps. However as timestamps of all events are recorded the actual simulation time can be computed. Figure 4.7 illustrates the notion of the task occurring upon an alternate course of action being taken. These concepts lead to the awareness model Evaluation Hypothesis

4.7 Awareness Model Evaluation Hypothesis

Figure 4.7 illustrates the situation when an awareness model is absent. Users have to wait on information to arrive or take more steps to retrieve the information. The main purpose of the awareness model is to make all the information relevant to a user's sphere of activity available. In case the information content is not completely available the awareness model provides meta-information that can help the user take steps to retrieve the information more efficiently. Thus, upon a mismatch and alternate course of action, by using the awareness model user's must acquire the needed information and accomplish the tasks in fewer steps compared to not having the awareness model. This is the hypothesis that is evaluated in our simulations. Interactions with the awareness model are also counted as tasks. However this interaction leads to more efficient work practice. The impact on work should be positive. **Figure 4.8** depicts such a situation.

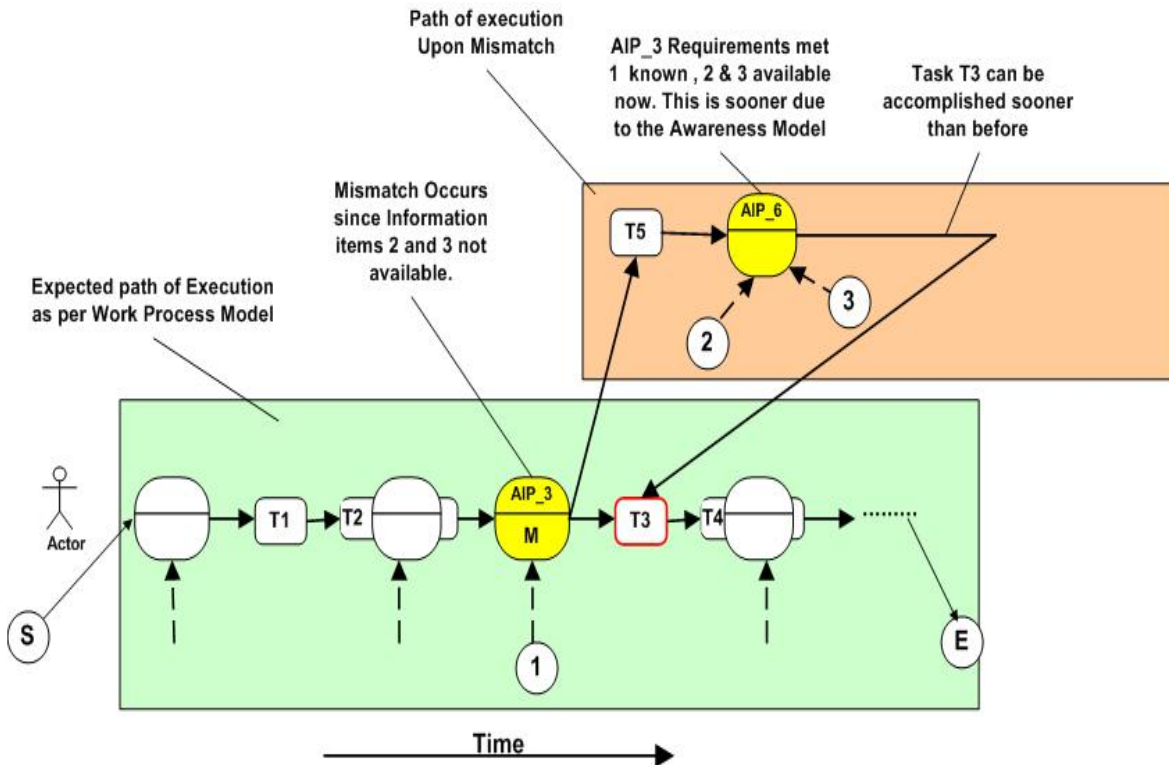Apart from the work getting accomplished the quality itself may be enhanced due to unanticipated benefits.

Figure 4.8 Minimizing "Mismatch(s)" with the Awareness Model

4.8 Simulation Output and Metrics

To validate the hypothesis of the awareness model the following metrics are recorded and analyzed. Every action taken by the user in the simulation is recorded with the timestamp. These include the actions taken with respect to the awareness model. For every AIP in the work process model, mismatches are recorded when they occur, along with the reason for the mismatch as well as the degree of mismatch. This is the key metric. The degree of mismatch is the number of information elements that did not match over the total number of elements required. All these activities are recorded as part of the event log, a view of which is shown in **Figure 4.9** below. If there is complete match then the regular course of action is taken. However depending on various factors and the degree of mismatch, the alternate course of action can be taken. The alternate course of action is recorded with corresponding rational. Analysis of the event log reveals the work practice for the scenario. This is compared to the work process model as well as other simulation runs. Analysis reveals all the AIPs where the awareness model plays some part. Comparing this to the situation when the awareness model is absent will reveal if the use of the awareness model resulted in fewer steps to the same goal. Detailed analysis can be conducted as all relevant aspects of the event are logged. If a user is waiting on information the amount of time this WAIT action took can be calculated from the timestamps. This is especially true for this action since measuring the WAIT action as just one step does not provide an accurate description of the latency that would not have normally occurred. Close examination of the log should also reveal the unanticipated benefits. For instance awareness of some information may result in better quality of the work product created, or more efficient allocation of resources, better utilization of resources, avoidance of redundancy and of course faster execution of the work (fewer steps and time taken). The user generating the action also must record this information. The logs can also demonstrate the ability to accomplish task not possible in the work environment with out the awareness model (since without the awareness model did not provide the sources). As actions using the awareness model are also recorded one can analyze the log to tell us which actions where used more often and how that impact the work. As the overall work process model is the composite f each user's work process model, the overall results are a composite of each user's work practice. **Figure 4.10** shows the work process model, the event log and the work practice derived from it.
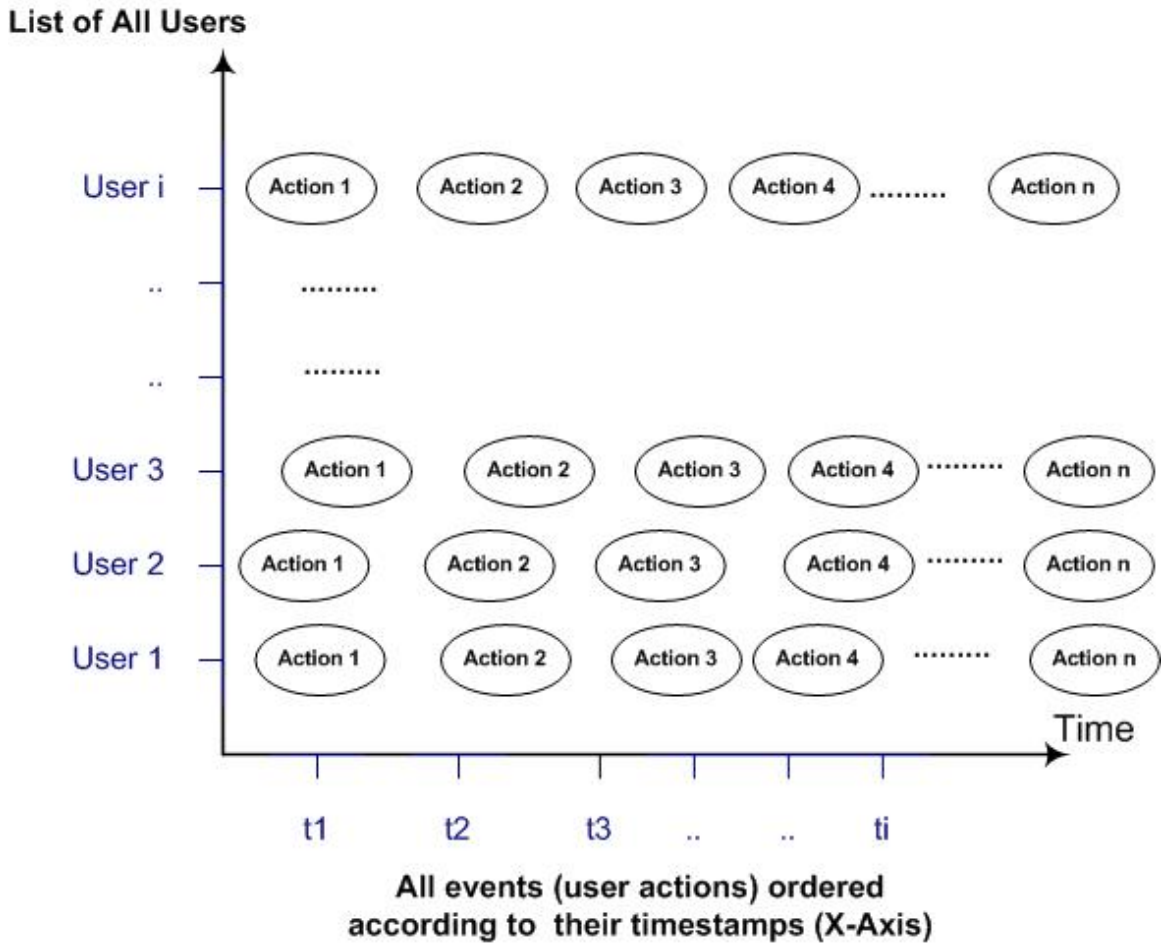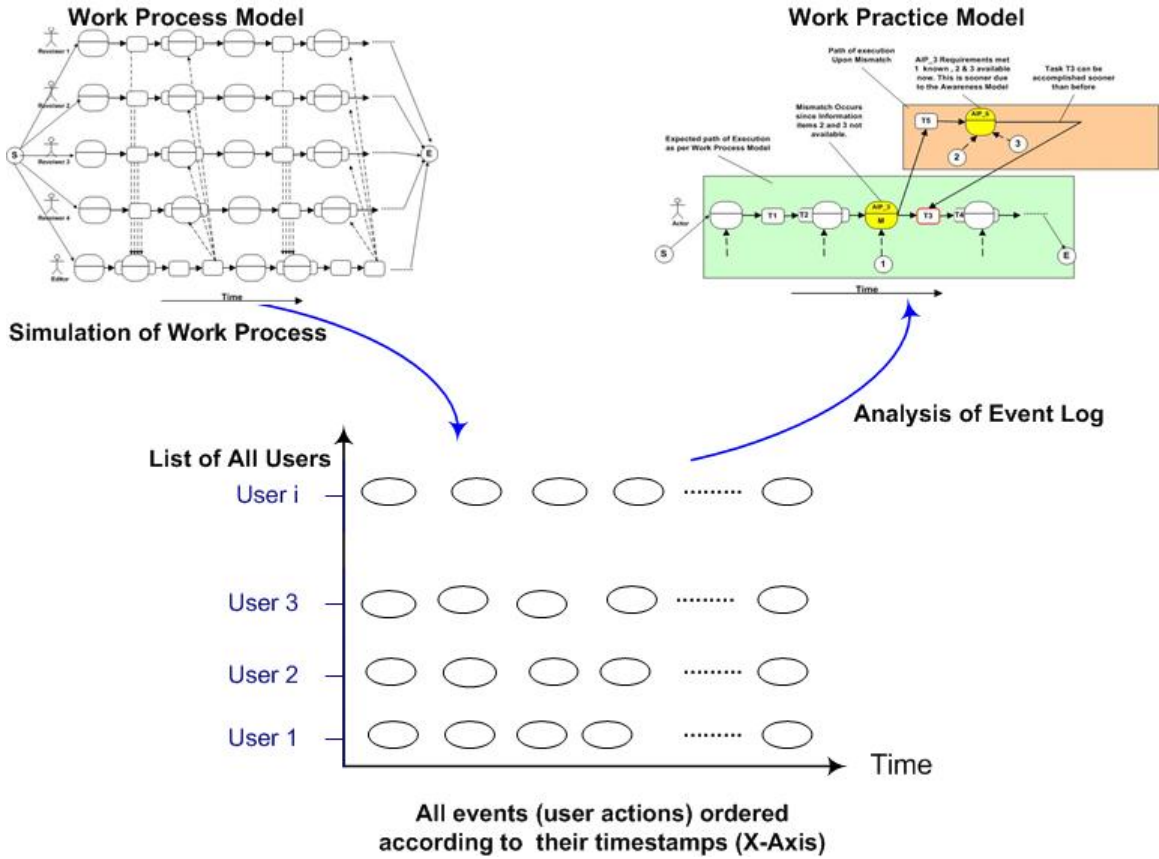
Figure 4.9 Event Log

Figure 4.10 Simulation Process

4.9 Experimental Methodology, Framework Development and Current Validation Process

The experimental methodology and framework that was described in the preceding sections has been followed in the validation of the awareness model, but in an informal manner. The underlying theory for the simulations is the same, viz. the notion of work process, work environment, and work practice, which are used to model collaborative processes and the role of awareness, the concept of mismatches, and the awareness evaluation hypothesis. However the simulation scenarios have not been explicitly modeled using the work process model graphs, in terms of the AIPs and other symbols. The work environment is nothing but the sources and their characteristics describe in Chapter 3. Work practice was just the enactment, which was recorded in the event logs. The logs are in their raw form, i.e. text-based and are yet to be distilled to crate the work practice model graphs.

The steps in the scenarios were defined informally in natural language, as was the information propagation that was supposed to occur. The objectives of the scenarios and the role that each human participant (role) was supposed to play was articulated before the simulations, however they were not instructed on an exact order in which to take the steps and how to take the steps. We found that giving the actors that freedom was helpful because actors were free to use awareness model as they pleased and use it in any way they wished to aid them in their work. Also the hope was that as they were unencumbered from following a strict script there activities would unfold more naturally. Each actor did take all the steps towards their individual tasks based on the roles. Thus more realistic situations were created based on the natural behavior of the actors which was actually beneficial for the evaluation of the awareness model since we wished to know how it behaved in a real-world setting. However users were asked to enter the mismatches and mismatch avoided whenever they were faced with such situations.

The reasons for not modeling each scenario in a formal manner were because the modeling formalisms (WPM graphs and WPrM graphs) of the experimental framework are still under investigation. There is work needed to be done to completely specify these formalisms as well as create tool support to both manually as well as automatically (from event logs) create these models. However as concepts in methodology and experimental framework were specified, we were able to begin the simulations, since validation of the awareness model and map was the primary objective.

The work done as part of the evaluation process in devising the methodology, experimental framework, and the Awareness Simulator will be extended (as described in detail in Chapter 5) and there is hope that it will lead to a better understanding of awareness in collaboration as well as its evaluation.

4.10 Awareness Simulator Architecture

Figure 4.11 illustrates the Awareness Simulator Architecture. The Awareness Simulator is based on the working of an awareness Framework. The Framework connects multiple information sources and the awareness model is the element that integrates these sources. Users interact among themselves and the awareness model via the awareness Framework. The simulator has the same logical architecture. We use a client-server paradigm where the awareness model is on a simulator server. Users interact with the model and amongst themselves via clients that connect to the server. The server handles all user interactions with the awareness model as well as among the users based on the awareness model's connectivity and privilege settings. The server also records all events in the Event Log. The architectural details are as follows:
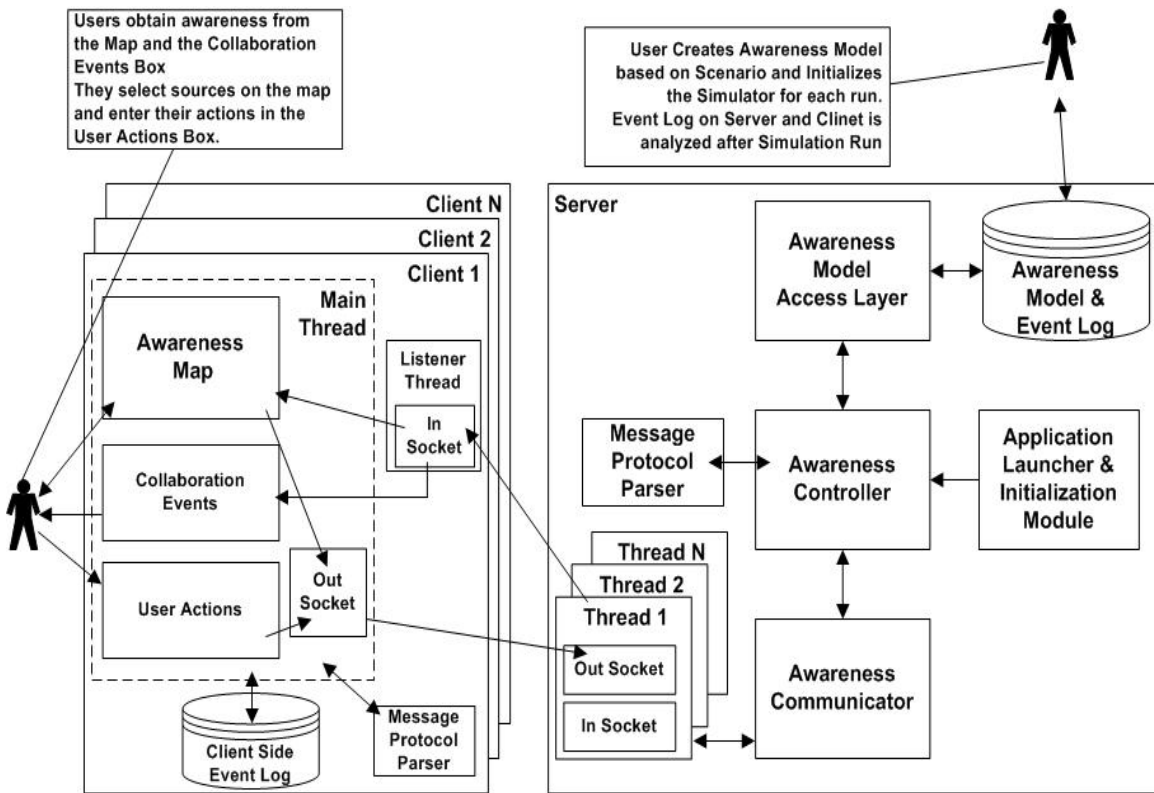

Figure 4.11 Awareness Project Simulator Architecture

4.10.1 Awareness Project Simulator Server

The server handles multiple client connections using sockets. The use of sockets allows users to access the server remotely. The server's IP address and port number are all that is needed for any user's client to connect and enact his or her work process model. This was multiple clients can be part of the simulation irrespective of their location or the host machine.

The Server uses a multiple threads, one per client to manage the communication. Thus each client has its own dedicated server thread with which it communicates with the Awareness Simulator Server and the awareness model. The threading mechanism allows the user interaction to process naturally as it would in any work environment with multiple users and applications. The manner in which users interact with the awareness model is not encumbered by other users' interaction. Clients can concurrently communicate with the server as they would in a natural setting.

The server consists of three main parts, viz. the **Awareness Communicator** – responsible for all communication between the clients and the server, the **Awareness Controller**- responsible for handling incoming messages, parsing them, determining the action to be taken based on the awareness model information, and sending messages to clients, and the **Awareness Model Access Layer** – responsible for interacting with the physical awareness model.

4.10.1.1 Awareness Communicator

The Awareness Communicator handles all the messages to and from the clients. It is responsible for creation of new threads whenever a new client contacts the server. Each thread has a dedicated socket pair, one for incoming messages (in) and the other for outgoing messages (out). Once the client is assigned a thread, it is started which means it begins executing its logic. The thread is added to the list of all threads that the Awareness Communicator maintains (this is done by the Awareness Controller explained next). The logic that each thread executes is essentially to listen to listen to messages from its client. When it receives a message it passes it on to the Awareness Controller. When the Awareness Controller needs to send a message back to the client it hands off the message to the Awareness Communicator. The Communicator in turn, sends the message to the client via the thread's socket interface (out socket).

4.10.1.2 Awareness Controller

This component performs the essential functions of the simulator. It receives messages from various clients via their server threads. It parses the message to determine what type of message it is. This is done using the Message Protocol Parser module. It contacts the awareness model and based on the message it may update the model and or retrieve information that determines what action needs to be taken. For instance if a message arrives from a particular source the Awareness Controller will retrieve form the model all the users who have the particular source in their foci. These users must receive the message. The Controller uses this list to send the message via the Communicator. As mentioned above the Communicator has a list of all clients (users) and sends the message to each one through their socket interface. This is a typical function of the Controller. All user actions map to functions that the controller performs. Most importantly the

161

Controller also records every event (messages) in the Event Log of the Simulator. The Awareness Controller is also responsible for any initialization work when the server is launched.

4.10.1.3 Awareness Model Access Layer

All interaction with the awareness model is handled through this component. The Awareness Controller invokes this module to perform any and all information retrieval as well as additions, modifications and deletion to the awareness model. In essence it is the data access layer for the Simulator.

4.10.1.4 Awareness Model

The Awareness Model is the entity storing all the information that is essential to the Awareness propagation the group's collaborative effort. As mentioned before the organization of data elements as well as actual data formats and restrictions depend on the individual work environments and sources that need to be integrated. We choose to implement the Awareness Model using an RDBMS. Tables describe the relationship between the users, the information sources and medium, the connectivity and the privileges the users have.

4.10.2 Awareness Project Simulator Client

Before describing the architecture of the client in **Figure 4.11** we examine the client interface. The screen of the Awareness Project Simulator Client's Interface for a user (in this case the employee from the Office Collaboration Scenario) is shown in **Figure 4.12** below. The graphical user interface (GUI) consists of three parts. The Awareness Map Window is displayed at the top. This window shows all information sources that the user has access to. The user can create foci, drag and drop sources into the foci, modify foci by removing sources from the foci and delete foci. The user can also view information about all other users and sources belonging to the group in the "User List" window that is to the right-hand side. The "User List" window displays, for each user, all the sources of information they use as well as the current status of the users, viz. the source they are currently communicating with. This information can be used to contact the users even if their sources are not available in the source superset. It demonstrates the notion of a *directory of information sources* in other words the **Source Superset**.

The user can view all the events occurring with respect to the sources in his superset in the window below the Awareness Map Window. This is called the "All Collaboration Events" window or just Collaboration Events window. This window displays all the messages that denote information generated by the sources in the user foci. Furthermore when sources are added and or deleted to the user's superset he can view messages about those events in the window.

The user's own actions with respect to all his sources are entered in the input window below the Collaboration Events window. The user enters the source id and the corresponding action. These messages are sent to the Awareness Simulator Server. The server communicates this message to other clients and it is displayed in their Collaboration Events Windows if this user's source is in their foci.
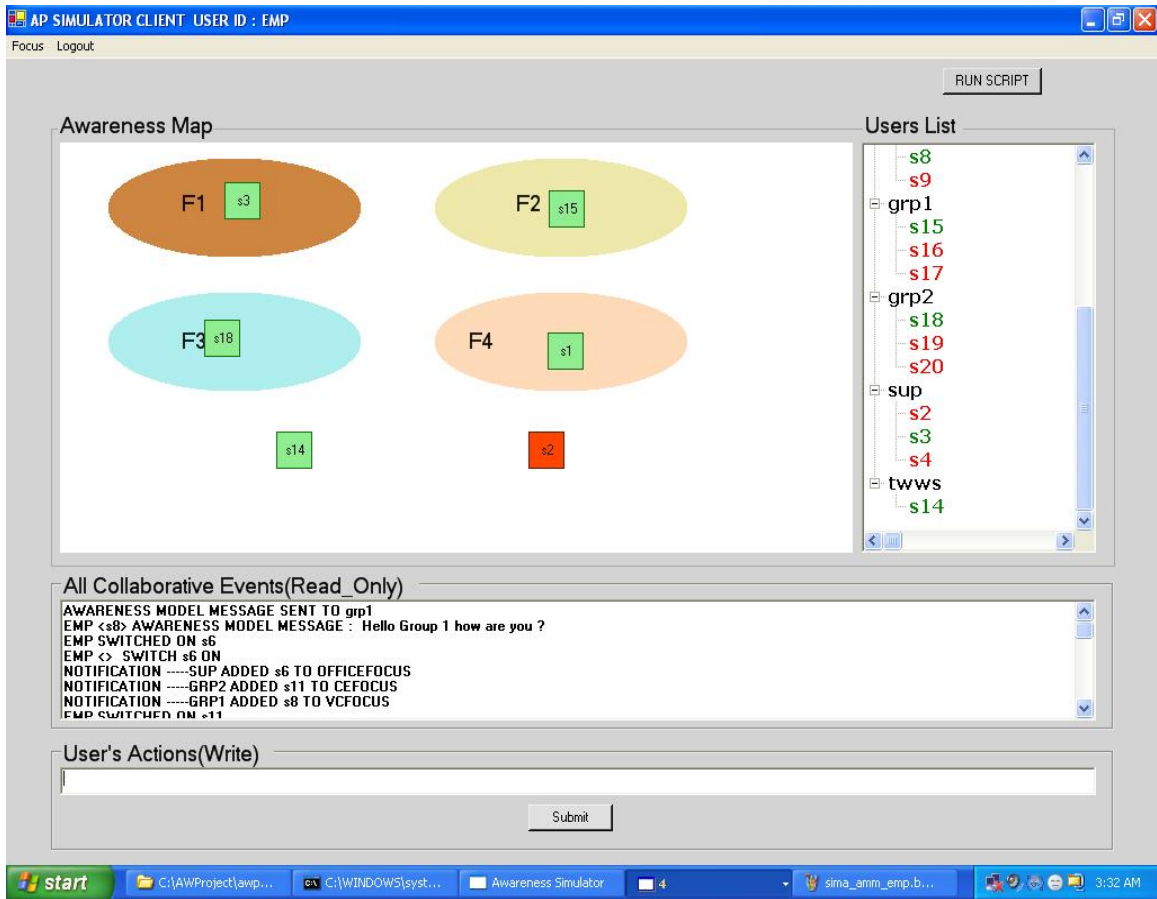
Figure 4.12 Awareness Simulator Client Graphical User Interface

The client's architecture consists of the main module that displays the GUI and handles the user's interaction via the Awareness Map Window and the Input Window. The client has a socket pair (in and our sockets) that communicates with its dedicated server thread. All the outgoing messages generated as a result of the user's interaction are sent via the out socket. There is a dedicated listener thread that listens to all incoming messages from the server via the in socket. There is a message-parsing module that helps in determining the type and contents of each message.

4.11 User Interaction with Awareness Simulator

This section provides a brief overview of the capabilities of the Awareness Simulator application in terms of the user interaction. For further details about all the functionality please refer to Appendix B. All the interaction is described from the perspective of users simulating a scenario. The Office Collaboration Scenario described in Chapter 3 was simulated as part of the validation process. Human participants were used in the simulation. Each human actor assumed a role from the Office Collaboration Scenario, viz. the employee, supervisor, Group1 representative (point of contact) and Group2 representative (point of contact). Each actor launched an instance of the Awareness Simulator Client and connected to the Awareness Simulator Server. A view of the interaction between the roles and the server is shown in **Figure 4.13.**
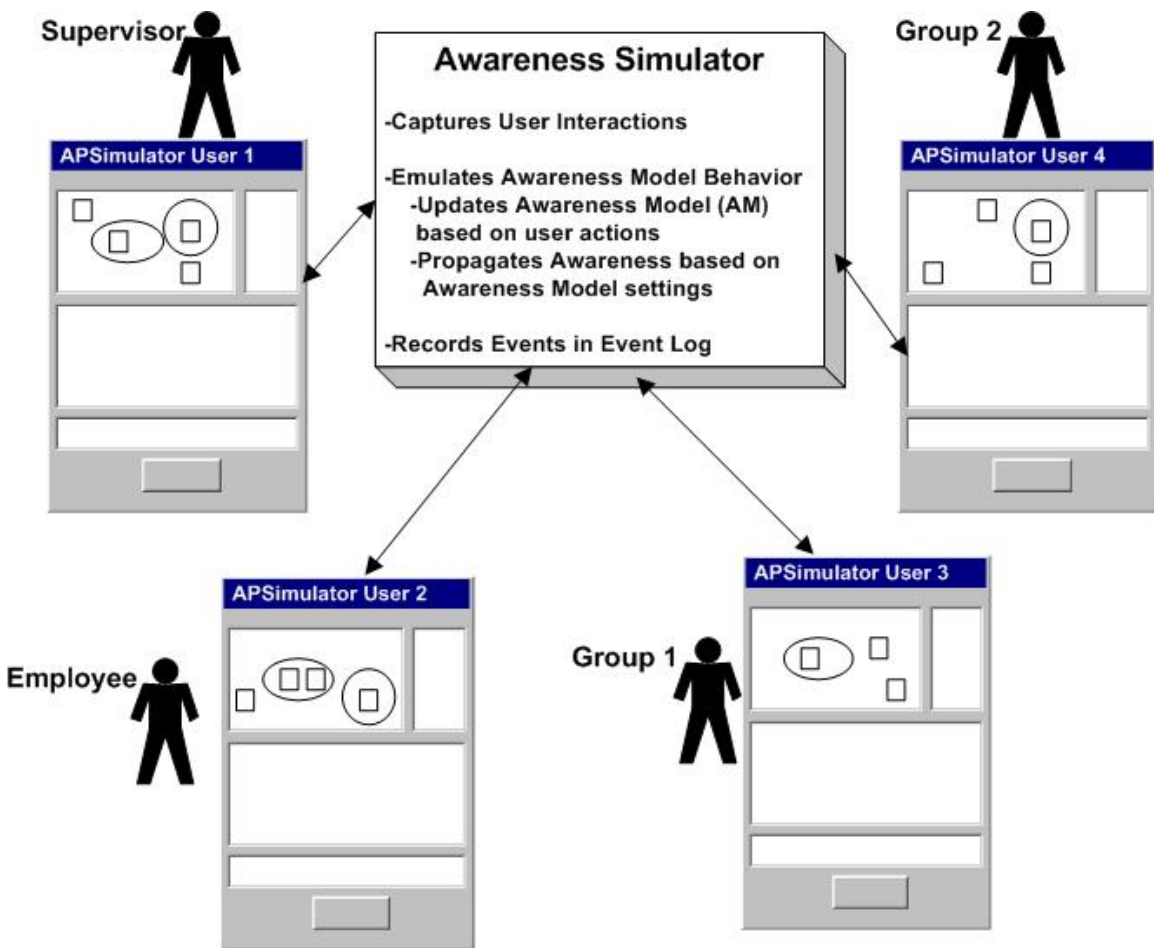

Figure 4.13 Office Collaboration Scenario Simulation

4.11.1 Initialization

When the server is launched, the Awareness Controller, Awareness Communicator and Awareness Data Access Layer modules start up and the awareness model described in Chapter 3 (awareness model database schema) is created in the MySQL RDBMS (The RDBMS service must also be started separately). All the tables are initialized with the data as described earlier. The server waits for clients to connect to it. Each actor logs on to the server using his login credentials. The server authenticates the role and if successful, sends back the appropriate awareness map to be displayed on each role (user's) client. This initial view is based on the data in the awareness model (**user_src_t** table). **Figures 4.14, 4.15, 4.16** and **4.17** illustrate the initial client view for each role. The user ids of the four roles are '**emp'** (employee), '**sup**' (supervisor), '**grp1**' (Group1) and '**grp2**' (Group2) and we will henceforth refer to the roles using their ids.
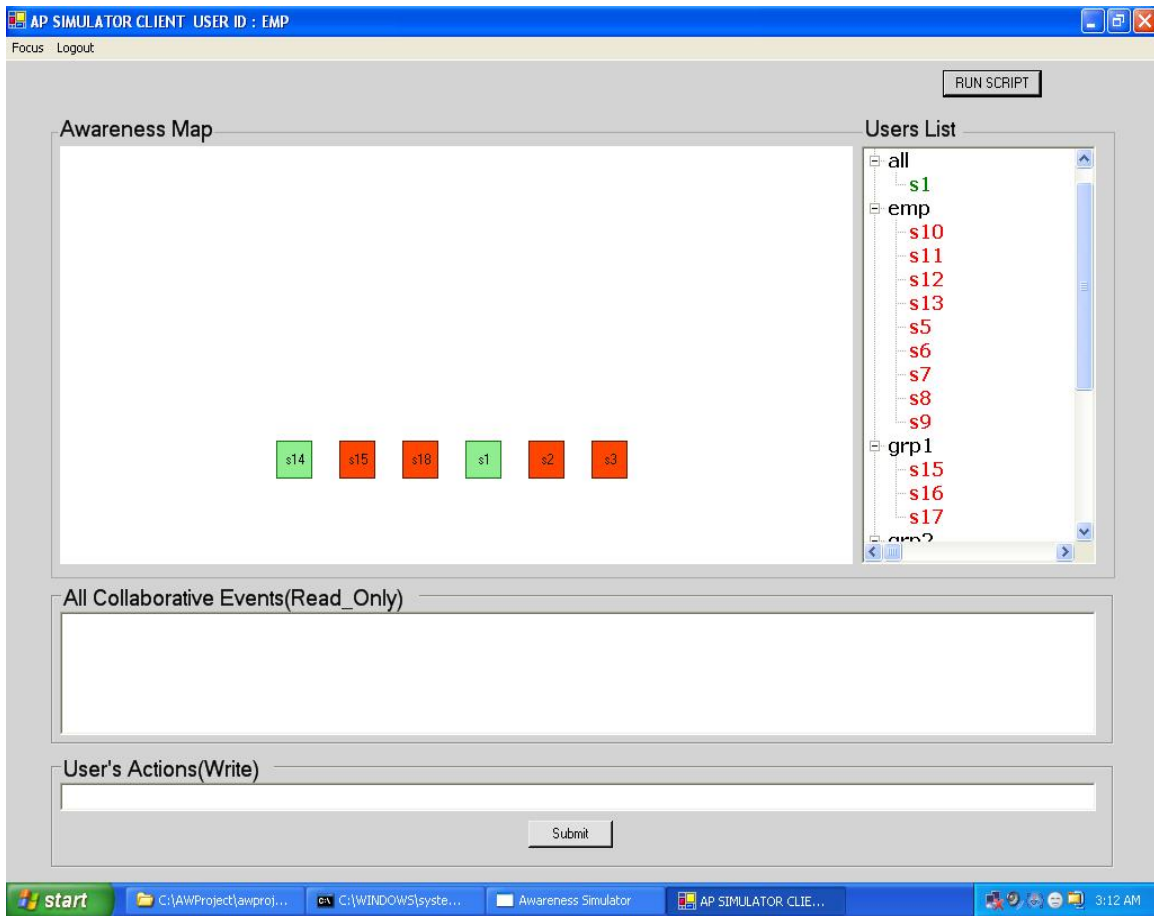

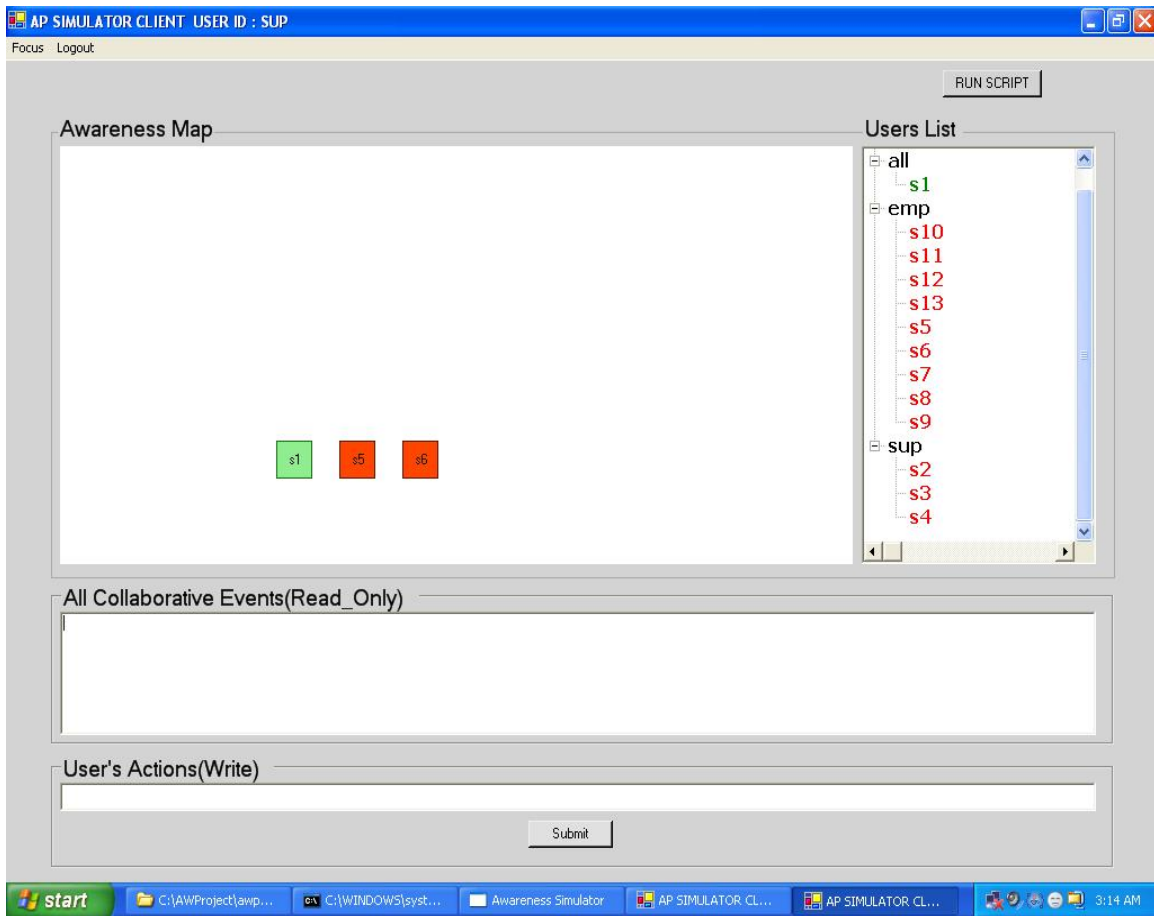**Figure 4.14 Awareness Simulator Client: Employee Initial View**

Figure 4.15 Awareness Simulator Client: Supervisor Initial View
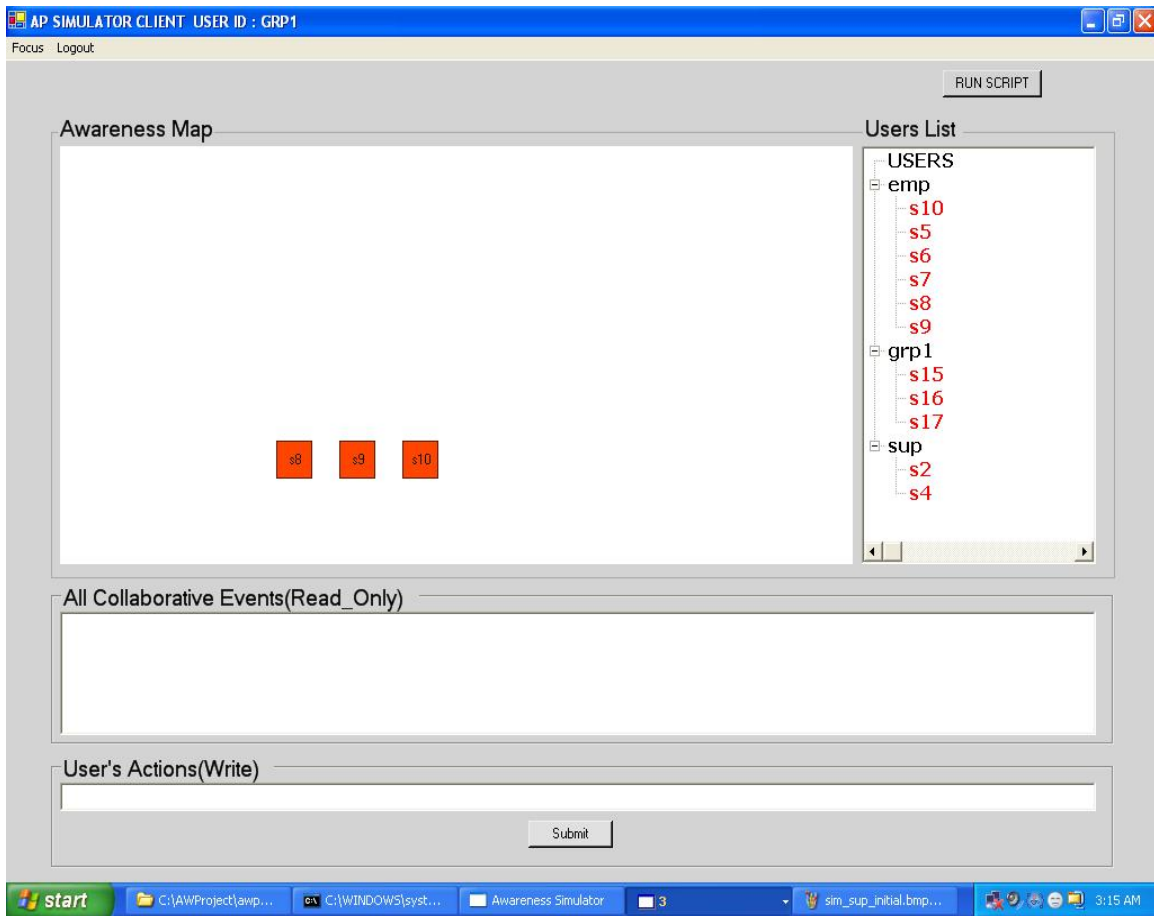
Figure 4.16 Awareness Simulator Client: Group1 Initial View
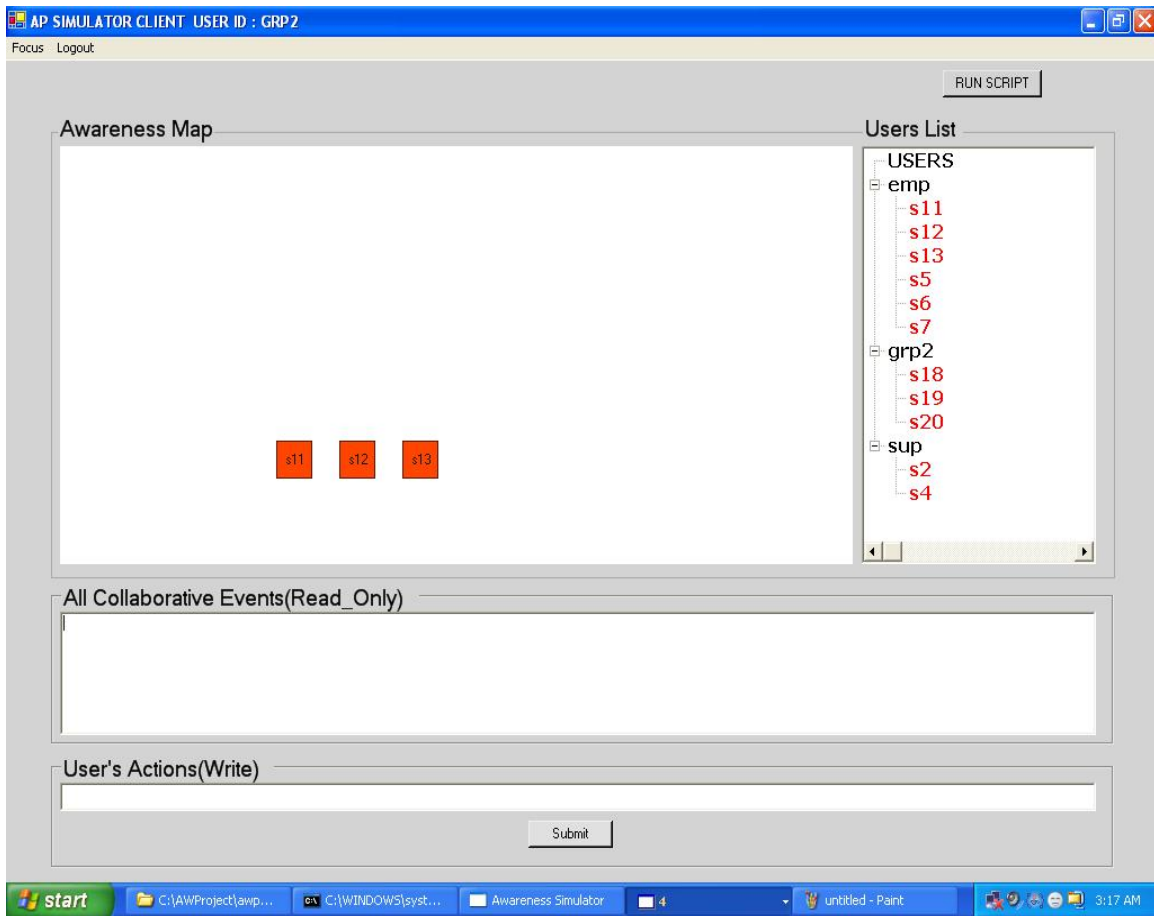
Figure 4.17 Awareness Simulator Client: Group2 Initial View

4.11.2 Foci and Notifications

Figure 4.18 shows the employee's client. The employee has created foci using the 'AddFocus' function available in the Focus menu item. The employee has added sources that sup owns (s3) in one focus and one each from grp1 (s15) and grp2 (s18) to two other foci. He has also added the s1 (calendar) and to the fourth foci. When users create new foci they have to assign a focus identifier, name and description. This information gets recorded in the awareness model tables and is used later to propagate information to the appropriate client when sources that are in their foci generate information.

When a user adds (drags and drops) a source into his focus, the source's owner gets notified. Furthermore sources that are active are indicated in green font and green icons, while those inactive are in red. Users are notified when other users change their status on any source from online to offline and vice-versa. User can activate their sources by right-clicking on the sources they own and selecting "switch on" or "switch off".

All user actions are echoed in the Collaboration Events Window. These include messages that the user sends (discussed below). All notifications are also seen in text form in the same window including those that update the Awareness Map and Users List Windows.
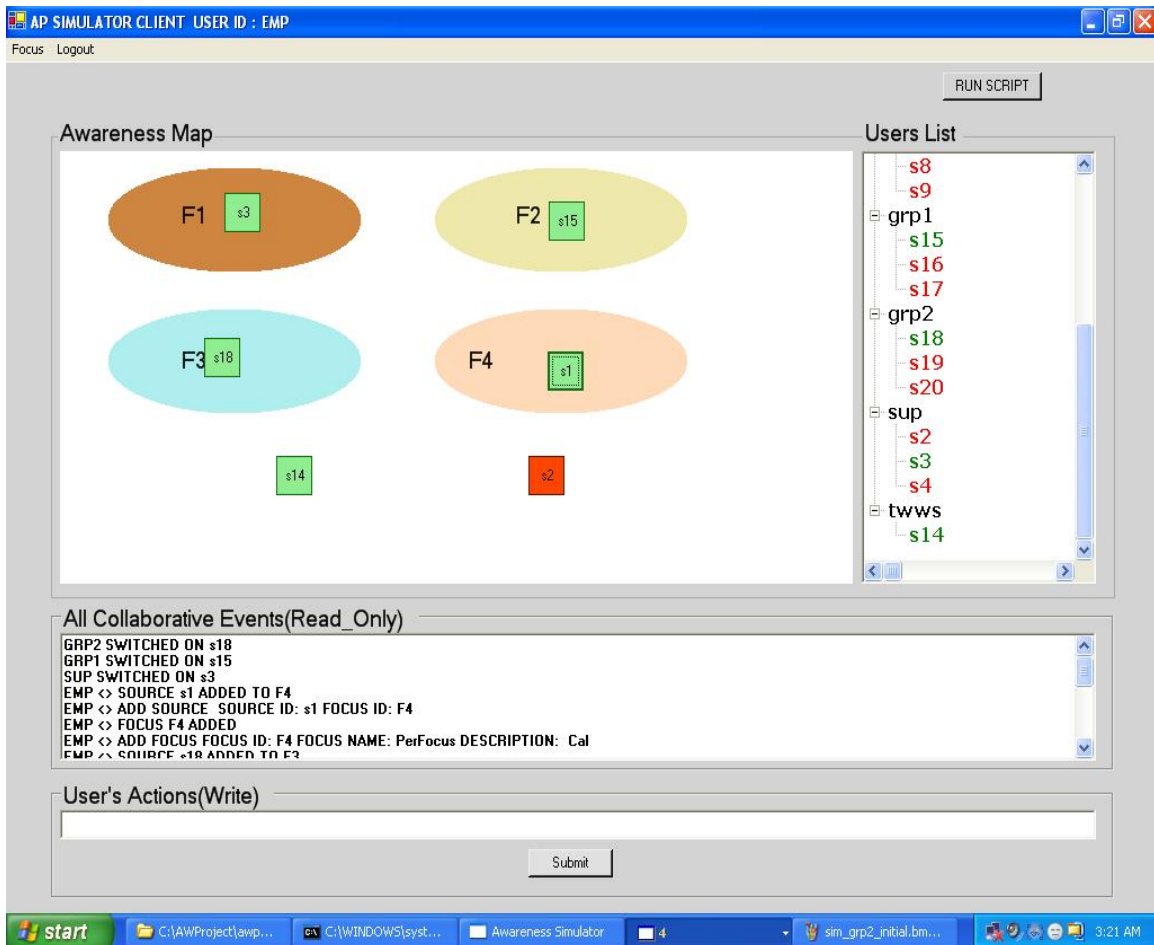
Figure 4.18 Employee Adds Foci and Sources

4.11.3 Interaction through Messages

Users interactions with their sources are simulated by having the users enter messages into the "User Actions" window below. Based on the interaction the following three types of messages are allowed.

1. If a user interacts with a source and wants the interaction to be propagated to other users based on the settings of the awareness model then he enters what is known as an Awareness Model Message or AMM. Thus any other user who has this source in his or her foci gets the message. This is because by including the source in their foci they have essentially subscribed to receive all information generated by the source. For example we see that the emp and grp1 are interacting in the meeting through the Video Conferencing Infrastructure. The video conferencing client application that **emp** uses to do so is **s8**. Grp1 has s8 in his focus so when emp simulates the act of "talking to Group1 using his Video Conferecing client" by typing an AMM message with the following protocol

➢ **AMM** *sourceId* text_string which in our example is

➢ **AMM s8 Hello Group 1 how are you?**

An email message from emp may be simulated as (s5 is his email account)
➢ **AMM s5 "My email is as follows…".**

This message is seen by grp1 in his Collaboration Events Window. **Figure 4.19** and **Figure 4.20** illustrate this feature. **Figure 4.19** shows the emp's client as soon as he has pressed the Submit button, the message can be seen echoed in the All Collaborative Events Window above. The awareness model's **user_src_t** table has the information about all the foci (and users) that the sources for a particular user are in and so it determines the recipients from that table and sends them the messages.
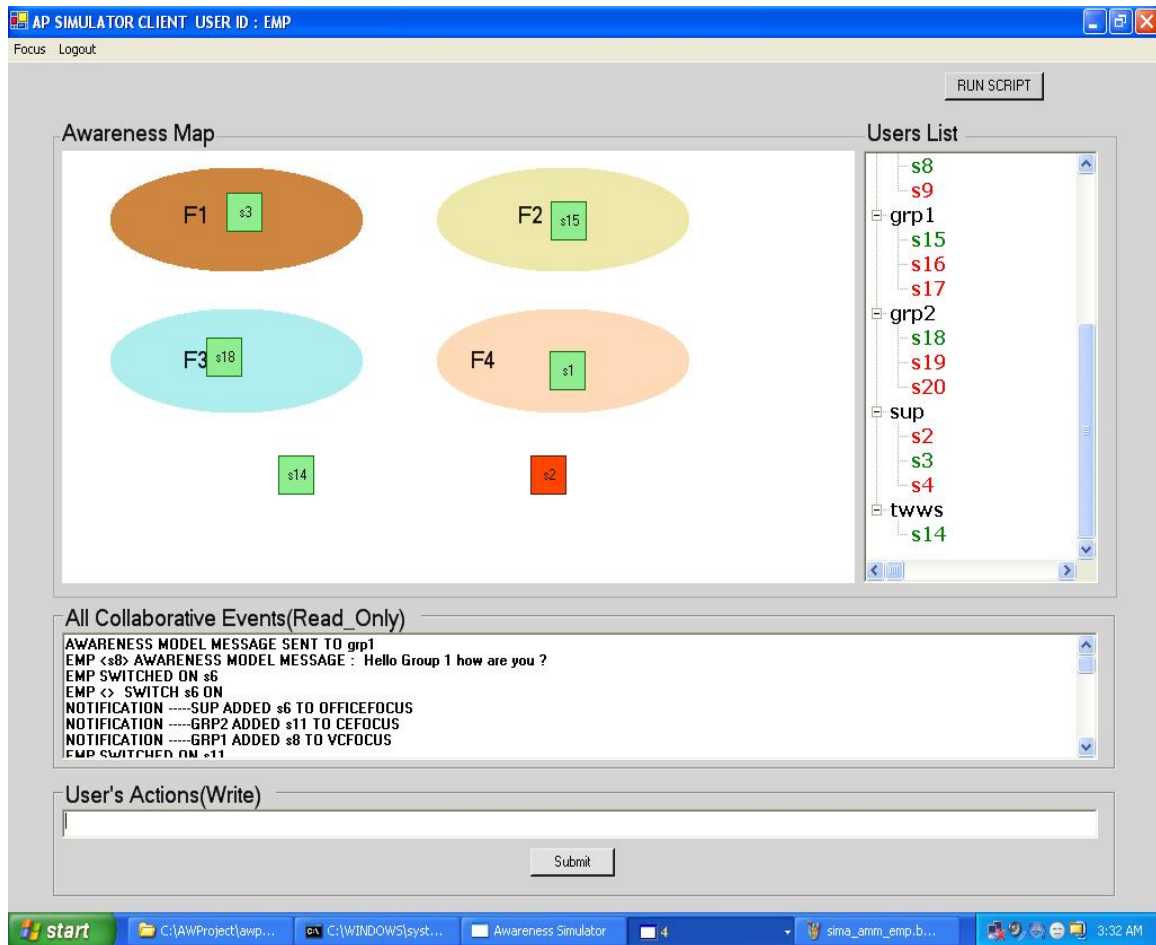
Figure 4.19 Employee Sends an Awareness Model Message (AMM)
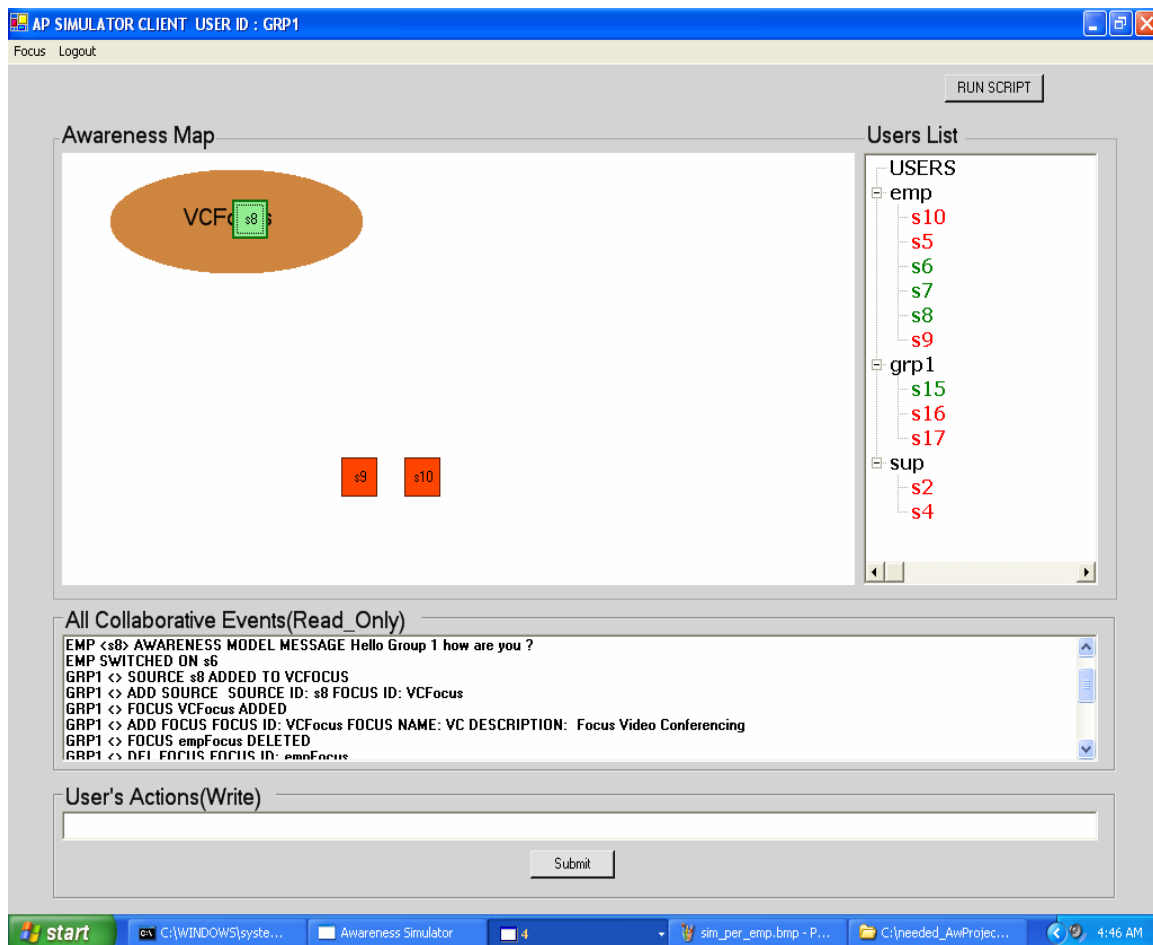
Figure 4.20 Group1 Receives the Employee's Awareness Model Message (AMM)

2. If a user wants simulate ***default application behavior*** when he uses some source application and does **NOT** want that interaction to be propagated according to the awareness model, then he simulates that action by typing a personal message **(PER)** according to the following protocol.

➢ **PER** *source_id recvUserId recvSourceId* text_string

Here the initiator has to type in his source id as well as the receiver's user id (recvUserID) and the source on which he would be receiving.
For example, if emp makes a phone call to grp2 using his phone (**s7**), then he would call grp2's phone which is **s20**. This message would be encoded as,

➢ **PER s7 grp2 s20 Hello Group2, how are you doing today?**

Grp2 would see this even if the source s7 is **NOT** in his/her focus. This is because the message is not propagating according to the awareness model settings. The

172

ability to simulate this behavior is one of the highlights of the Awareness Simulator since it is very natural that in real-world collaboration users do not choose to have the awareness model involved in their activities. However we often find out that these interactions can influence the collaboration also and they must be taken into account in collaboration. This is similar to the case where one may talk about official work when making a personal call to a colleague.

Figures 4.21 shows the employee entering a personal message and **Figure 4.22** shows grp2 receiving it even with s7 not being in any focus. Also you can see that the sources s7 and s20 are not active when used. This is because the current implementation of the simulator does not check to make sure users are working on active sources only. This should be remedied for the future. On the other hand one may consider this appropriate for a behavior of applications not intending to propagate information via the awareness model. Just as the information they send need not be propagated as per the awareness model, their status could also be hidden from the awareness model.
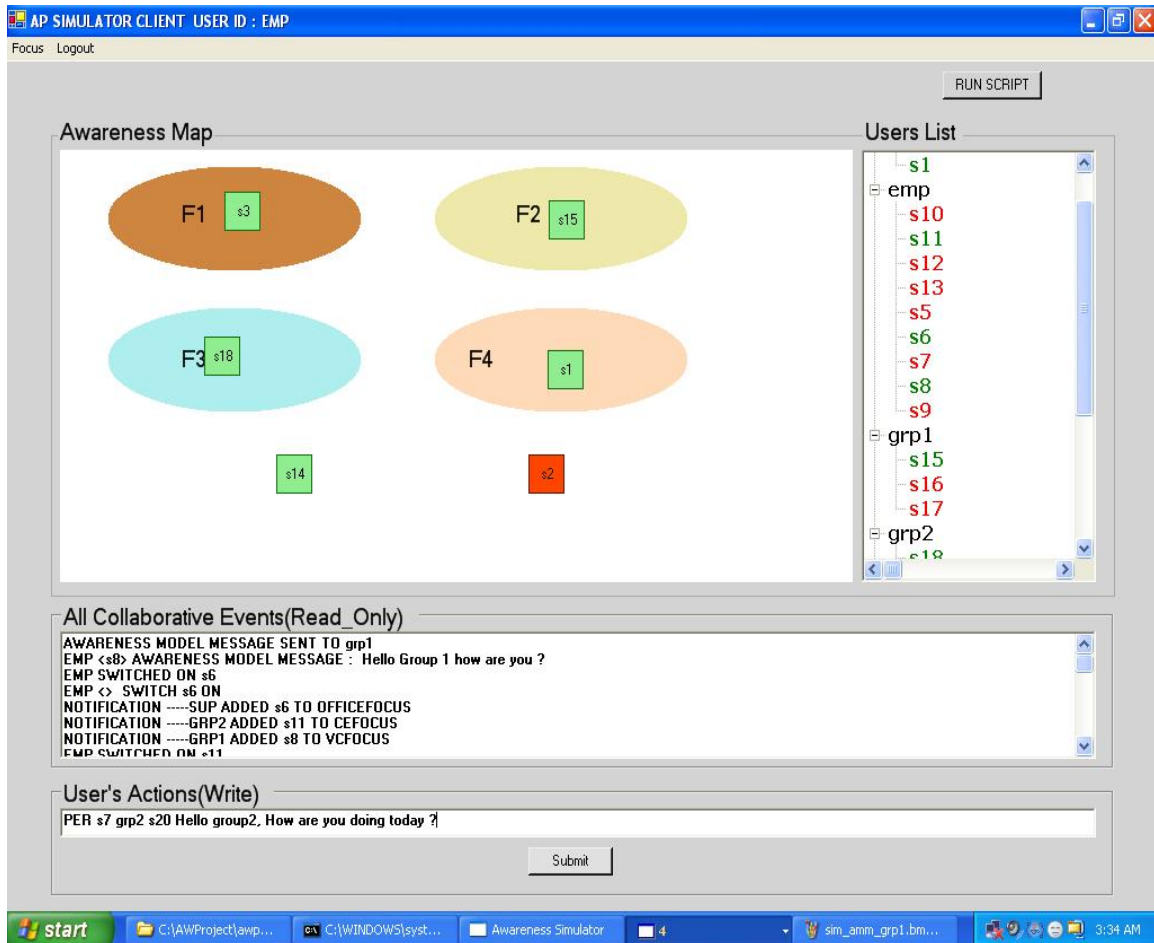
Figure 4.21 Default Application Behavior: Employee sends PER Message to Group2
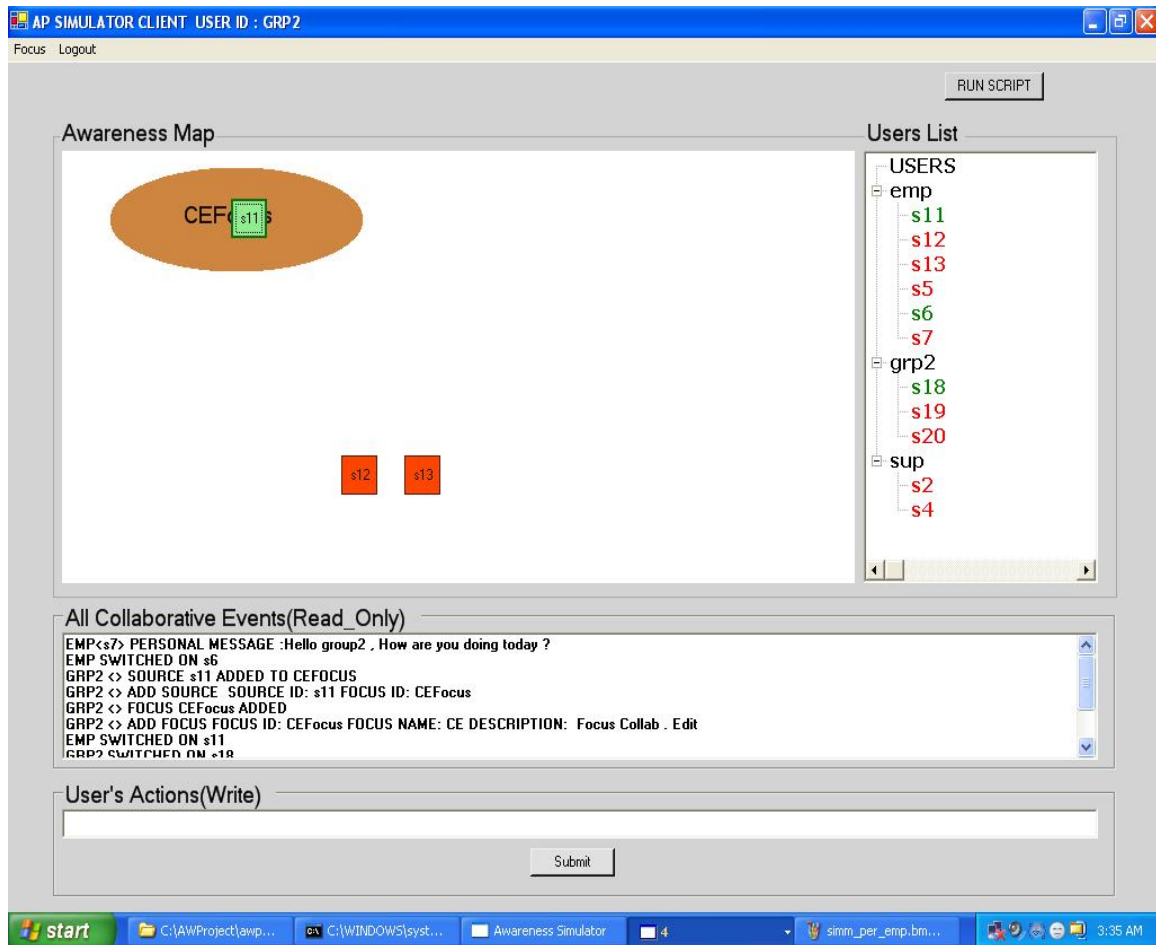
Figure 4.22 Default Application Behavior: Group2 receives Employee's PER Message

3. The third type of message is for the simulation purpose. These are the messages for recording the mismatches (MM) and mismatches avoided (MMA). They follow a very simple protocol of

➢ **MM** text_string

➢ **MMA** text_string

The text string records all the details that the user wishes enter such as what the mismatch (or mismatch avoided) was about and any explanation about why it occurred.

4.11.4 Enhanced Forms of Awareness

Features to obtain information about information sources as well as enhanced forms of awareness can be invoked as shown in **Figure 4.23** by right clicking on the user or the source as the case may be. The features that can be invoked are as follows:
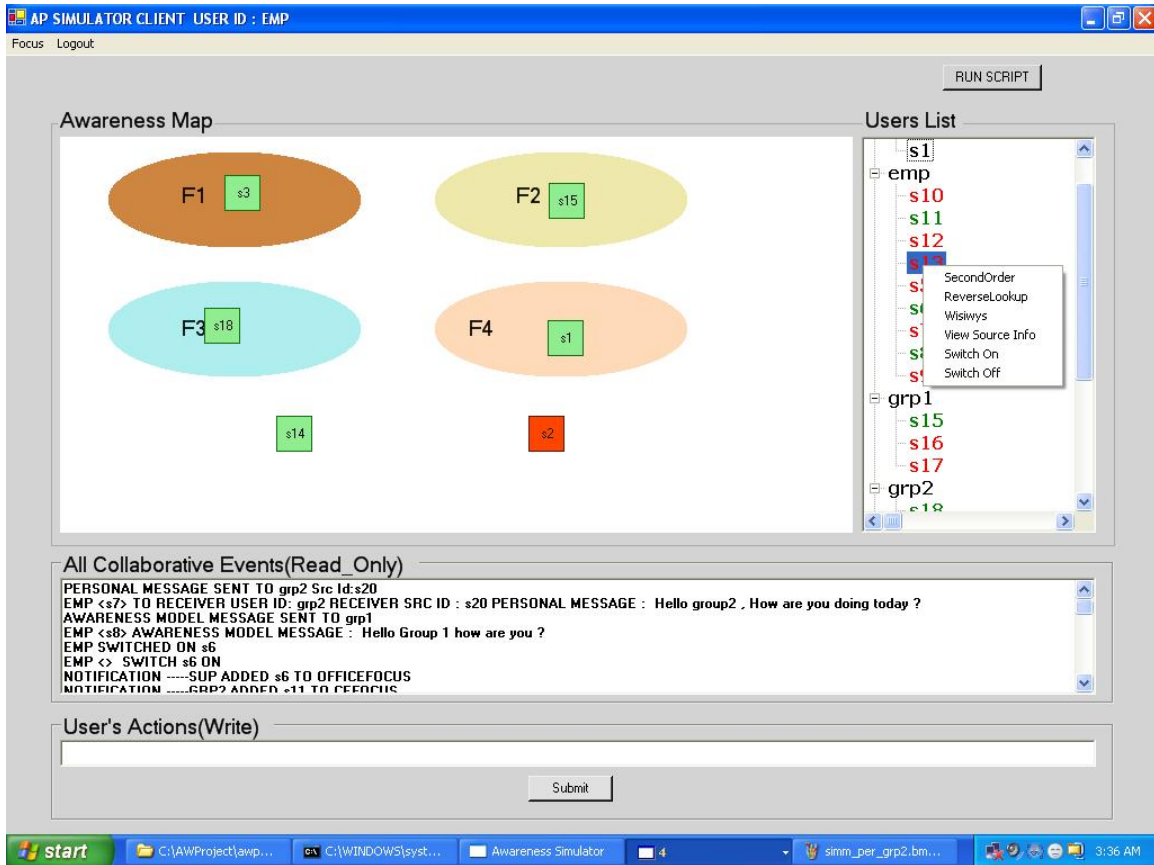
Figure 4.23 Invoking Enhanced Forms of Awareness Features

4.11.4.1 View Source Info

Figure 4.24 shows the result of invoking the "View Source Info" on a source in the User List Window. The information about the source is displayed. This information is retrieved from the table **source_t** in the awareness model. In the figure grp1 views the information about source **s9** that the employee owns. A user can obtain information about any source that is visible in the Users List Window.
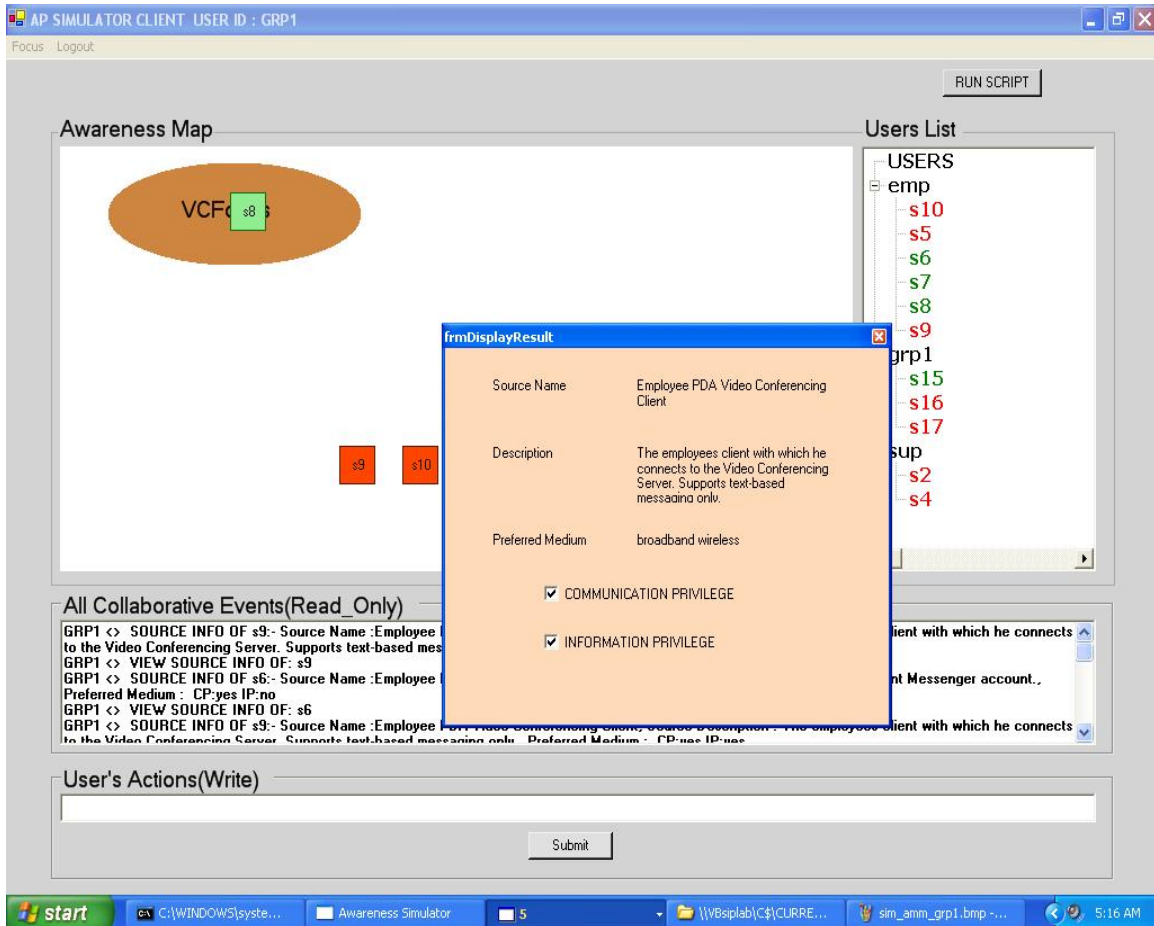
Figure 4.24 Group1 Invokes View Source Info on Employees Source

176

4.11.4.2 Second-Order Lookup

Figure 4.25 shows the result of invoking the SecondOrder (Lookup) feature on a user. This feature can be invoked on a user and renders to the initiator the entire awareness map of the user he is interested in. For example, **Figure 4.25** illustrates that the sup's client displays the awareness map of the emp on whom he has performed the SecondOrder (Lookup). If one compares this awareness map with **Figure 4.18** that shows the emp's client we see that it is the same view.
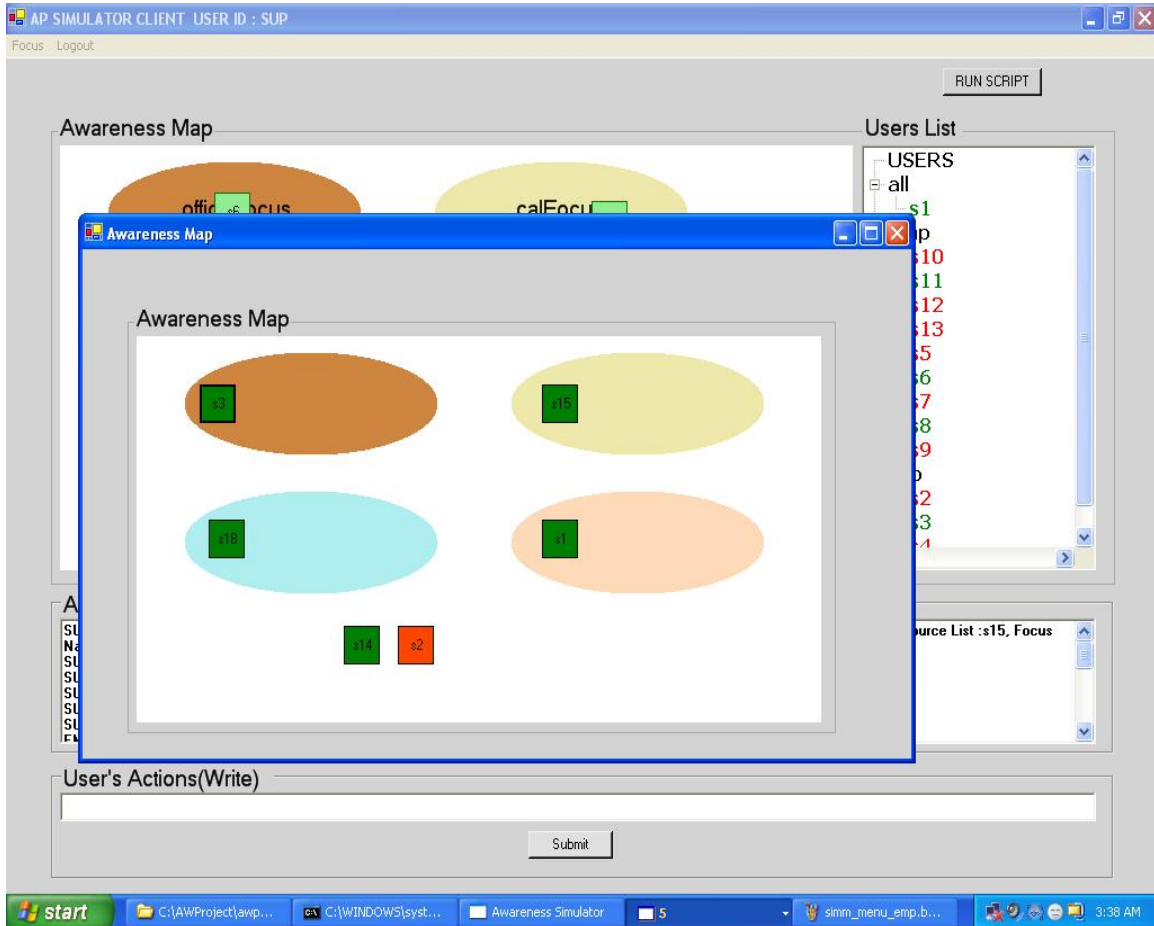
Figure 4.25 Supervisor Invokes Second-Order Lookup on Employee

4.11.4.3 Reverse-Lookup

When a user invokes this function on a source he owns, it tells the user the list of all users in whose user list this source is present as well as the foci of that user this source is currently present. In other words it provides the user an awareness of who may access information from his source (or have contact privilege) and who is currently accessing information he is generating. **Figure 4.26** illustrates the information seen by emp when emp invokes ReverseLookup feature on the source **s6** which happens to be in the sup's focus.
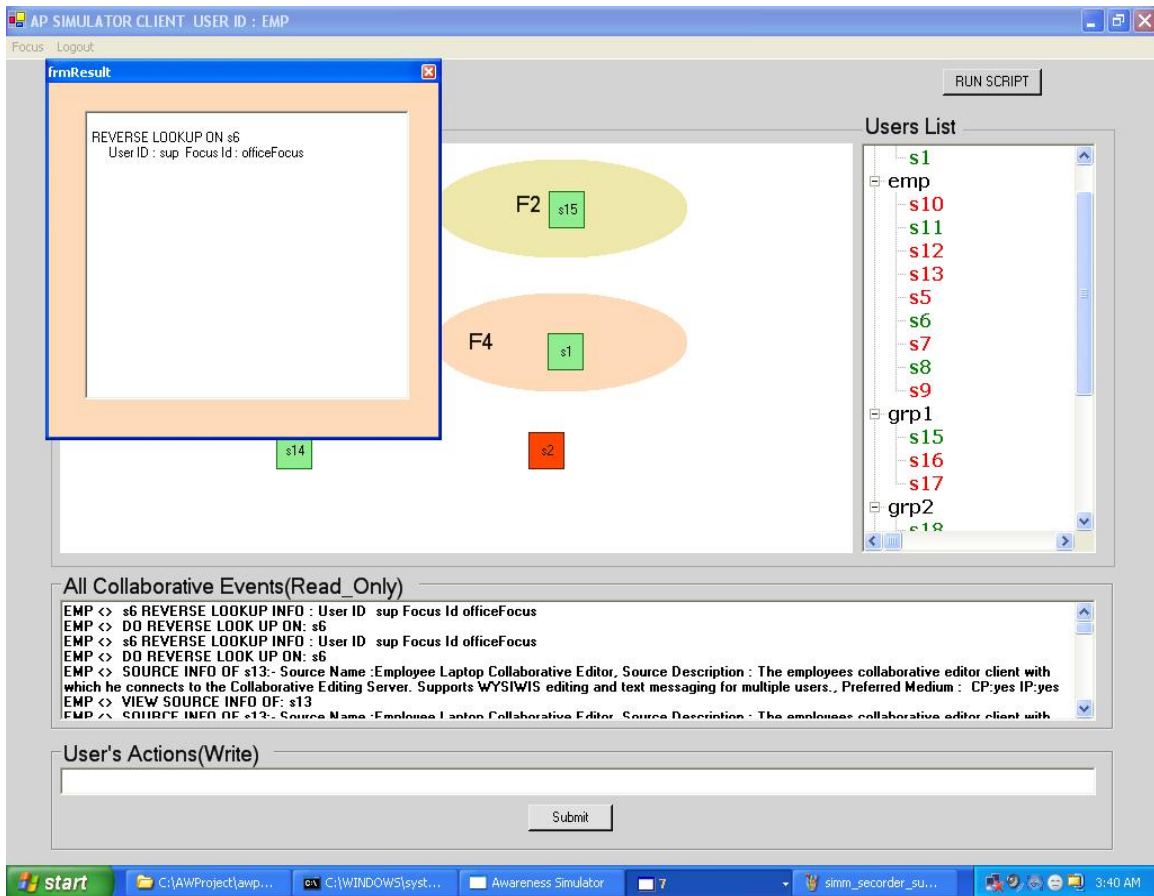
Figure 4.26 Employee Invokes Reverse-Lookup on Source

4.11.4.4 What I See Is What You See (WISIWYS)

Invoking this feature on a source tells the user who else has access to the same source, i.e. who else has it in their foci. One can perform a WISIWYS on any source even those not owned. This is unlike reverse lookup, which one can perform on only those sources they own. **Figure 4.27** illustrates what emp sees when he performs WISIWYS on the **s1** source. He can that the sup has s1 in his focus too. The emp can see his own access details in the list along with the sup. This is obvious since he also has the source s1 in his source superset and awareness map (The repetition of the information needs to be remedied in the future).
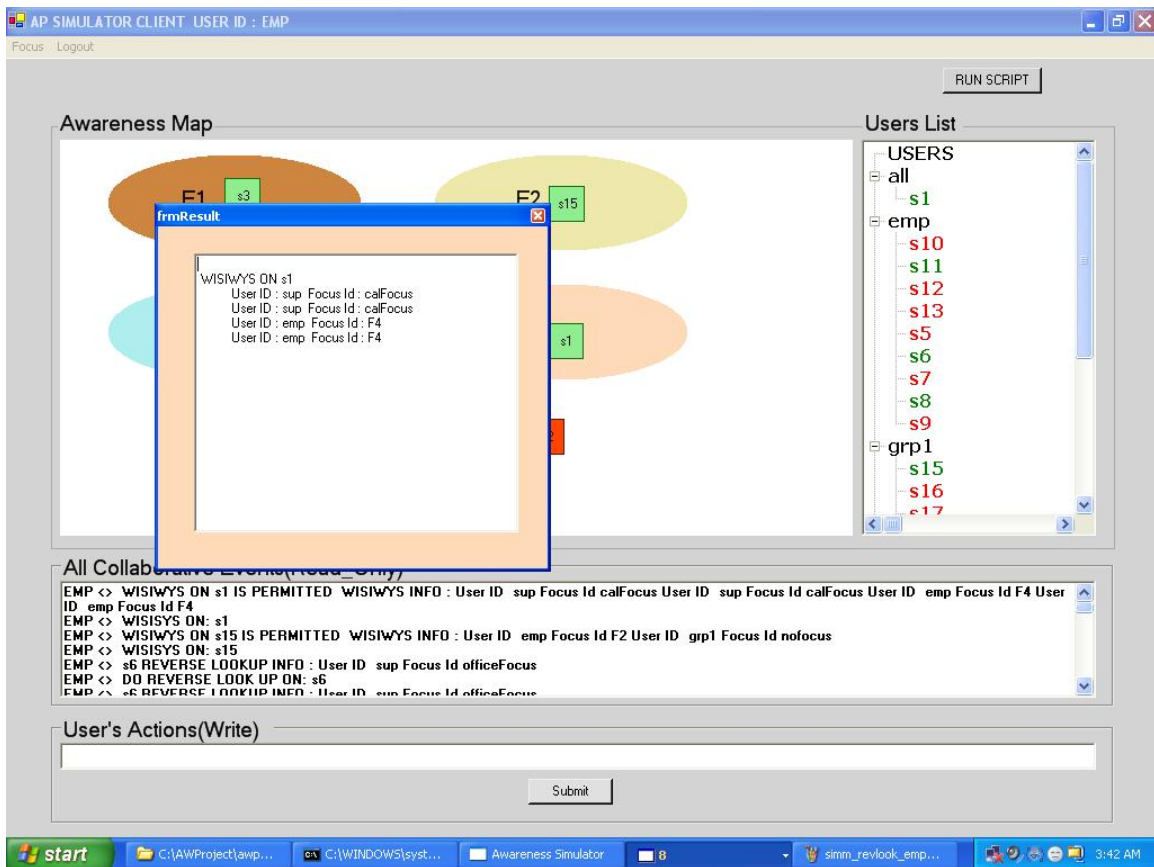
Figure 4.27 Employee Invokes WISIWYS on Source

4.11.5. Run Script

The Run script feature asks for a text file that consists of messages that a source can generate. These messages are read off the file and sent to the awareness model just as any other user's messages except that they are done automatically. If this source is on someone's focus they will receive these messages otherwise no one will receive them.

The objective of this feature was to incorporate an automated agent in place of another human actor. This was used to generate alerts for a traffic watch web site (**twws**) that alerts its subscribers by periodically sending them information about the road conditions (conditions due to inclement weather, traffic, accidents etc.). The employee who is a subscriber can get this information if he has the source twws in his focus and then act accordingly. When he is not interested he can remove it from his focus. This source just like the office calendar (s1) is always active, however the recipient has the choice to obtain information when he chooses to. This is another example of the user's ability to choose relevant information as per one's requirements. To launch the twws website one launches a client as usual and logs on to the server with the twws credentials. Then the Run Script feature should be invoked which asks for a script file. Upon submitting the name of text file, the client reads off the messages and sends them to the awareness simulator.

Chapter 5
Results, Conclusions and Future Work

This chapter begins with an analysis of the results obtained from simulations conducted using the Awareness Simulator. The utility of the awareness model and map and their limitations are part of this analysis. The insights gained from conducting this research and the outcomes achieved are summarized. We conclude this chapter and this dissertation with various avenues that this research effort opens for future research and applications.

5.1 Simulation Results, Observations and Analysis

The Office Collaboration Scenario was modeled along with its work environment and the corresponding awareness model was created as described in Chapter 3. A number of simulation runs were conducted and recorded. The runs were conducted with human users where each human user took on the role of the employee, supervisor, Group1 representative and Group2 representative. Each human had an Awareness Simulator client with the corresponding awareness map displayed. The awareness map seen was according to the actor's role and privileges as per the Office Collaboration Scenario. All parts of the Office Collaboration Scenario were simulated. Simulations were conducted according to the sub-scenarios identified in **Table 3.1**. Along with the description of the scenario, **Table 3.1** has described where there is need for awareness, and the impact on the scenario both **with and without the awareness model**. In each simulation run the users tried to enact according to the sub-scenarios objectives. They were also encouraged to enter the Mismatch (MM) and Mismatch Avoided (MMA) messages as was described in the experimental framework in Chapter 4. A Mismatch message (MM) with appropriate rational, was to be entered whenever they felt that they were unable to acquire some information during the course of their interaction evening the presence of awareness model. A Mismatch Avoided (MMA) message along with the reason, was to be recorded whenever they felt some aspect of the awareness model has aided them by providing some information they could not have gotten without it and therefore avoided a potential mismatch. The aim was to ascertain the role of the awareness model and gauge its impact on the overall outcome of the scenario.

5.1.1 Recording Simulation results

The results from the simulation runs were collected in the event log in the Awareness Simulator Server component. Apart from the event log in the Awareness Simulator Server, each client also recorded all the actions being carried out by the user as well as every piece of information being sent from the awareness model to the user, in other words everything that is displayed in the Collaboration Events Window. The logs indicate the manner in which the users acted in these situations. Their use of the awareness model is recorded in rigorous detail. **Figure 5.1** shows the event log in the

MySQL Query Browser tool, for a particular simulation run. **Figure 5.2** shows the same table and alongside it the corresponding client-side log. The table has been queried to display all the events generated by the same client, which is the same as the client-side log. The rational for the client-side log was that it gives an immediate view of the client's activities without having to query the server. Furthermore it was invaluable during the development of the Awareness Simulator in ensuring (and recording) that messages sent from the server were getting to the client and vice-versa.
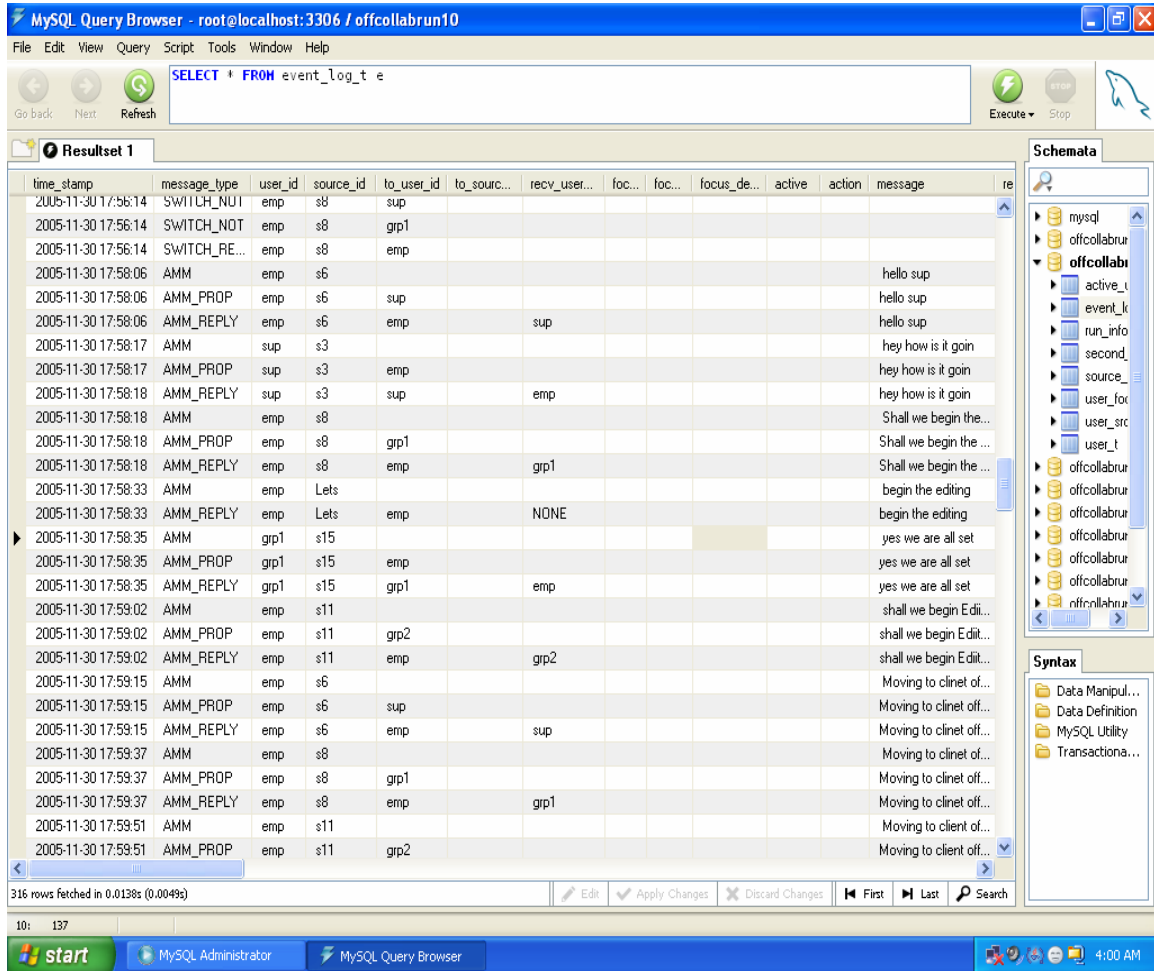
Figure 5.1 Awareness Simulator Server Event Log
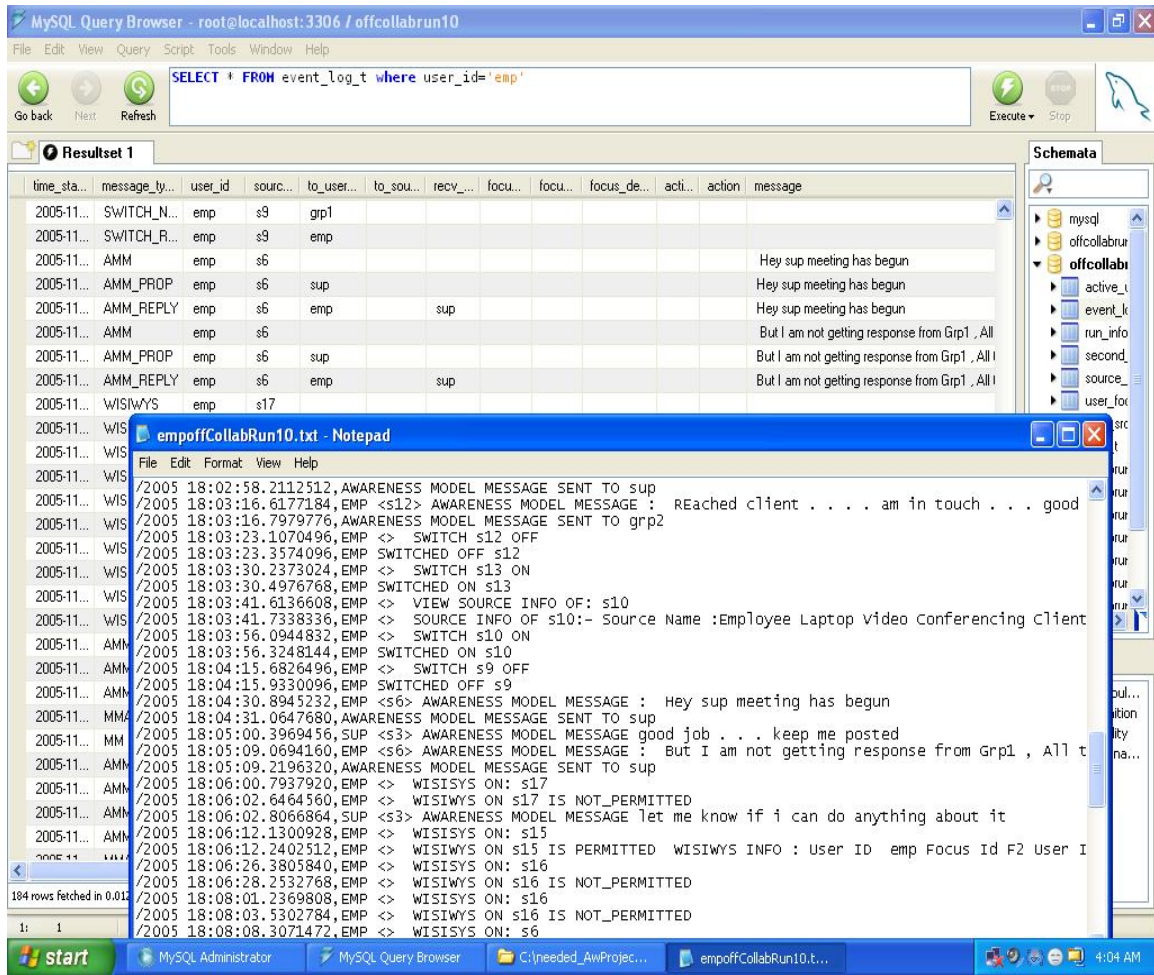
Figure 5.2 Awareness Simulator Client Collaboration Events Log

An examination of the logs show that every user action be it an interaction with some applications (simulated by entering a text string) or interaction with the awareness model (invoking some feature) was propagated to awareness model and then based on the presence of the originator's sources in other users' foci it was propagated to their clients. This is revealed in their logs.

On close inspection of the logs one can see that certain actions are taken based on the information received by the user, which is recorded in the lines above the line indicating the user's action. The messages recorded in these lines provide information to the user. These messages are actual information content such as an email message or the results of an action he performed such as invoking the functions for features such as WISIWYS, Reverse-Lookup and others. As a consequence of the information obtained the user performs some other actions. This can be clearly seen in the lines recorded after. Even information such as change in status of the users (offline to online and vice-versa on a certain source) which can be seen by the user as change in color on his awareness map, was also propagated in text form and recorded. This was essential from a simulation perspective because just by inspecting the logs one was able ascertain the effect that this

awareness information had on recipient. This effect can be seen because if there was an action that was a consequent of this information, it was recorded below.   For example, as soon as a user comes online we see that another user who is made aware of this, contacts that user. If this information were not propagated as text, then it would have been lost. The user cannot be expected to manually record everything he sees and at the same time continue to enact his role. Apart from the requirement to manually enter the MM and MMA messages, one of the highlights of the simulator is the ability for the user to work as per his role with the same interaction capabilities without having to worry about the simulator in the background interfering with the flow of his activities. At the same time the design of the simulator enables collection of all the necessary information in a manner so as not to perturb the users in their activity.

More Information about data collected from the simulation and its availability can be found in Appendix A. The general observations from all the simulations carried out are discussed.

5.1.2 Observations and Analysis about User Interactions

Observations were made about the manner in which the human participants interacted with the simulator itself. These were the following:

1. Some orientation is needed to become familiar with the notion of awareness model. It was found that not every participant immediately made use of all the functionality provided by the awareness model. This was especially true of features such as What I See is What You See, Reverse Lookup, Second-Order Lookup, and even View Source Info. These are not commonly available in other applications in everyday use. However participants immediately identified with the notifications of other users logging onto sources and logging off. This is a commonly observed feature. The fact that these new features were not used initially is most likely due to the feeling of unfamiliarity that one experiences with any new application. After the first three to four runs, when the actors discussed their actions and found out that they could use these features, they made more frequent use of them.

2. Manual recording of Mismatches (MM) and Mismatch Avoided (MMA) messages:
   To evaluate the awareness model utility, actors were expected to record a "Mismatch message" (MM) whenever they needed some information and the environment they were in did not provide it. Similarly if they were able to use the awareness model in any manner to obtain some information, (which was otherwise not possible given their environment) they were supposed to record a "Mismatch Avoided" (MMA) message. The messages were recorded with a description and rational of why the corresponding mismatch occurred or was avoided.

   During the flow of events and actions the actors would not interrupt to record the above messages. So some participants chose to record them at the end of the simulation. The actors went over their actions and recorded the messages. This is one aspect of the simulator that could be improved, as users cannot be expected to stop what they do and record the MM and MMA messages. However this limitation can be

overcome by a review of the event logs on both the client and server side. If any awareness model feature has been used, that action and the corresponding information obtained have been logged. Furthermore the subsequent actions that are recorded in the log reveal the role the information played in influencing other actions. Thus the intention of the user in using the awareness model can be inferred. This reveals how the awareness model aided in terms of providing information thereby avoiding a mismatch in information requirements. This was the method used throughout in analyzing the event logs and they proved to be very useful. However one of the future improvements to the simulator would be to try and minimize the overhead of recording by some other means.

3. In the initial set of runs, when users were still not familiar with the awareness model dynamics, users would send messages to each other simulating actions and if they did not receive a response for a while, they would orally ask if the recipient received the message. There was also concern if the awareness model implementation was working correctly to send the messages to the appropriate recipients. A confirmation notification from the awareness model was added in response every user's message to help test the awareness model and also give a better sense of acknowledgement to the users initiating the messages. It made the simulation sessions smoother without interruptions and thereby the simulator user-friendly. This is similar to an application's acknowledgement to a user's action. For example when you send an email, the application informs you the message was sent. This is no way informs the sent if the recipient has received the message or let alone seen it. This message only informs the user that the awareness model has sent the message as per its rules and the list of all recipients who will receive it. This knowledge was useful in the simulation since it helped to reinforce the initiator's state of awareness about who was receiving the messages. Furthermore if the message did not go to someone because the user had mistyped the source or incorrect syntax it would act as an error message and help the user redo the action.

5.1.3 Observations and Analysis about Awareness Model Effectiveness

The following observations were made about the effectiveness of the awareness model and map in improving the work of the participants of the scenario. Careful inspection of the event logs reveal that in each sub-scenario there were situations where having the awareness model helped to accomplish the task which would not have been possible otherwise. This was possible because of the various features of the awareness model and it ability to propagate the information via the notion of foci and notifications. In each of the cases from Table 3.1 that were simulated, information form awareness model was used in the following ways to improve the group's work.

1. Selection of sources of information based on their characteristics and the current requirements of the selector is one such example- this is reflected when the employee has to be mobile and uses a PDA (sources s9, s12) and then laptop (sources s10, s13). The Video Conferencing (source s15) and Collaborative Editing applications (source 18) provide information in different forms, that is accessible by the employee's client applications and the employee can maintain awareness. Because of his ability ascertain the capabilities of his collaborators' sources' (sources could generate information in different formats) the capabilities of his own sources, as well as the ability select and switch easily, the employee can continue to maintain his awareness. This is major boost to the productivity of the groups to have him participate in some form in spite of various changes.

2. When the employee switches his sources (please note that every application is a source even if it is acting only as a client), the other users (Group1 and Group2) get notified of his switching to those devices (via notifications) and they in turn can take this into account in their communication with him. Thus there is a ***heightened level of mutual awareness*** that comes about due to the awareness of each other in that situation. This heightened sense of awareness is essential in collaboration as it can ensure success in many processes.

3. Notifications that were received by all users helped in ascertaining the status and activities of other users. This helped avoid unnecessary overhead in trying t find out each time the status and current activities of others.

4. Notifications sent were very useful in augmenting the state of awareness for every actor. Especially for role such as the employee who was multi-tasking and concurrently involved different processes, notifications were invaluable.

5. The ability to simultaneously maintain awareness of various contexts of activity y such as working with both Group1 and Group2 was extremely useful The employee was able to make some decisions about the work with respect to one group based on the information he was made aware of by while interacting with the other group. This is an example of opportunistically using the situation to improve the quality of work because of awareness of both groups. The groups themselves may not be aware of the relationship, and neither was the employee however the awareness maintained brought about such an opportunity.

6. The ability to maintain awareness of status and activities was also fully exploited by the supervisor. He was able to contact the employee who was away at a meeting, regarding work related to a client in the main office. The supervisor was not only able to reach him but more importantly reach him in manner that caused as little disruption as possible to the employee's participation in the meeting. The supervisor's awareness of the fact that the employee was online and responding to this instant messenger application, enabled him to first of all make a decision as to contact him at all and even then doing so via text-messaging which he knew would not disturb the employee. The employee in turn had continued to maintain awareness of the groups he was working with and was able to contact them, and accomplish the required work.

7. Ascertaining the connectivity and capability of the other users proved very helpful in another situation as well. If communication with a particular user was not possible because the user was not active (not logged on to a particular source), or the user wishing to initiate communication was not able to do so (for instance the user might be in a meeting and cannot use his phone but can text message the other party), the user wishing to communicate could request some other user to communicate on his/her behalf. This was possible because the awareness model displayed the online status of the third user. Thus the first actor knew the status of both the other actors.

8. There were situations in the simulation where the presence of the awareness model did not directly aid in getting information needed to accomplish a task. These were the situations that were recorded as mismatches (MM) in the event log. For example if user A intends to communicate with user B, but is unable to do so because user B is not active at one of the sources that A can contact B at. In the scenario the employee is unable to place a phone call to the Group1 representative. However employee can contact the representative only through an instant message and or email. But the representative is not active at either application so the employee knows it is not possible to contact him. The representative could be called on his phone, however the employee cannot do so being in a meeting. The task of actually contacting the representative was not accomplished. However the awareness model provides an alternative way he can be contacted. The employee is aware that the supervisor at the office is online. He is able to pass Group1's contact to the supervisor and have him contact the representative on his behalf. Here is an example of the awareness model aiding in completing the task though it took more steps than normally required. Of course if the supervisor is also offline then there is nothing the employee can do except send emails regarding s the fact that the had tried to contact each of them. Again if the supervisor and was also available only via his phone the employee could have done noting. The usefulness of the awareness model can only be exploited if users are connected to it in some manner through some applications that is integrated to the model. If some connectivity is possible then the awareness model can be used to somehow delegate actions to others or use more steps to accomplish tasks.

9. Another limiting aspect that was encountered during simulations was that users could only make use of those awareness attributes that are propagated by an application

(information source). Thus to be more useful the interfaces to the awareness models must be designed to propagate more information.

Thus the lack of connectivity to the awareness framework and the capabilities of the application interfaces can create situations where even with the awareness model, the collaboration may not necessarily occur as desired. On the other had the above results show that even with some amount of awareness about a certain situation, one can find alternative means to accomplish the goals, even if it may take more time and resources to do so. Finally if there are failures historical awareness support can always be invoked to determine the cause of such failures.

10. One aspect that was not covered was the "presence of memory" in the information sources themselves. The current version of the Awareness Simulator does not simulate the storage of information that sources such as email servers, shared spaces and others exhibit. Due to this the search capability could not be simulated. The memory capability would have brought in the dynamics of applications that employ asynchronous communication and made the interaction closer to the real-world applications. Currently asynchronous communication (email) was simulated as synchronous messages.

5.2 Analysis and Conclusions

The results from the simulations show the impact of the awareness model and map on group work. The events recorded in the logs demonstrate the utility of almost every aspect of the awareness model (all the features which were discussed in Chapter 3). It was observed that a ***heightened level of mutual awareness*** could form among group that is due to awareness of each other and their individual situations. This notion of mutual awareness is difficult to identify and characterize due to its implicit nature. However the lack of such awareness can result in failure in collaboration. It is this mutual awareness of various aspects of the situations that make the processes go smoothly and ensure success. The logs also reveal that notifications about various aspects and the use of enhanced awareness have all been utilized to their benefits. This is evidence to the claim that the awareness model and map can be tremendously useful in improving the quality of collaboration in heterogeneous environments.

This research effort was motivated by the fact that awareness is a key enabler to successful collaboration. '**Quality of awareness**' is a new notion introduced along with the factors that define effective awareness. The impact of heterogeneity in impeding effective awareness propagation was discussed. It was established that need of the hour was to enable awareness propagation in non-uniform environments, while empowering users with the ability to tailor the quality of awareness they receive. In the quest of solution the impact of heterogeneity on awareness quality was determined. based on this requirements for effective awareness propagation were identified. There are two categories of requirements – for physical integration and those related to information integration. Inspired by related work a change in perspective was called for in dealing with heterogeneity and the notion of an **awareness framework** has been proposed along with a description of all the features and required components in its generic architecture. However in dealing with the complex task of information integration something more than a physical framework was needed and this was a central logical entity that could provide a complete picture of all the information sources available in a collaboration setting. An **awareness model** has been proposed as the solution. The desired features of such an awareness model were identified and discussed in great detail and the awareness model forms the primary contribution of this dissertation. The practical implementation issues were identified and it was argued that engineering such a framework is very feasible especially with the current innovations in technology. A proof of concept model was created and validated through simulations of realistic collaboration scenarios that revealed that the awareness model and map could be extremely effective. One of the outcomes in this effort was the creation of an Awareness Simulator application as well as investigation into an experimental framework that could be further developed to aid awareness evaluation in CSCW. Apart from its obvious use the Awareness Simulator application itself could be used as a teaching aid in CSCW instruction of awareness concepts and awareness systems.

The power of the awareness model lies in how versatile the enhanced awareness and access control model can be made as well as in the amount of control users are given to customize their privileges and profile. The power can be limited by the connectivity i.e. if some application and user is disconnected from the awareness framework, or if the communication capabilities are limited then the ability to obtain effective awareness is diminished. The power of awareness model is limited by the amount and type of

information is propagated by the interfaces and so the correct engineering of the awareness framework is most crucial. However the beauty of awareness is such that knowledge of aspects that someone is disconnected or that someone has access to only limited capabilities can itself prove to be very useful information in finding alternate means to accomplish the same tasks that would have required better capabilities. Thus the true power of the he awareness framework (and awareness model) lies in their ability to propagate such meta-information. The utility of awareness framework can be seen in any domain where that heterogeneous tools and applications are employed, be they critical situations that require a very high level of awareness in real-time, such as Emergency First Responders responding to an event (Emergency Medical Technicians, Emergency Medical Services, firefighters) and military, to industrial applications such as distributed software development, services industry (law firms and healthcare), distance education and even consumer services for entertainment (location–based services).

5.3 Future Work

There are two broad areas with respect to which this research can be further explored. These are (1) with respect to engineering and awareness framework and awareness model and (2) in the realm of awareness evaluation. There are a number of significant items of future work possible in each area.

5.3.1 Engineering an Awareness Framework and Model

5.3.1.1 Engineering an Awareness Framework, Awareness Model & Ancillary Components

The immediate endeavor is to create an awareness framework as described in Chapter 3. This would include the awareness model with the features proposed. The awareness framework would be implemented keeping in mind the engineering issues that were described in Chapter 3. This would involve creation of interfaces for some candidate applications (email, instant messaging, video conferencing and others) with different clients (desktops, handhelds, cell phones). These would be bound together with an awareness model using some sort of middleware. Some preliminary work in this direction has revealed that different configurations are possible based on the applications themselves and their supporting infrastructures. One of the key aspects is design and developments of interfaces for the representative set of applications chosen.

Interfaces designed must not affect the applications own functionality as well as their non-functional attributes such as security and performance. If being part of the awareness framework is detrimental application vendors might be less inclined to provide users with the ability to connect to the awareness framework. Similarly choice has to be made for the appropriate middleware technology from the various standards available. The middleware must be able to connect the applications and propagate different types of information along with the necessary timing characteristics. For example, in the case of user interaction with an application, all the awareness attributes that contain information about the user's interaction must be conveyed faithfully. Thus for each type of application, the key challenge lies in identifying the types of information to be propagated and the specification of an interface. If interfaces are specified, then vendors can implement them (standardization would help in widespread adoption). As an example, in the case of an email application, all the functionality that the user can undertake (send, view, attachment and so on) may be part of the awareness attribute set that is to be propagated. In doing all this, the impact of all the above design and implementation choices on security and performance of the applications and the overall framework itself must be considered. Apart from the application interfaces and the middleware the awareness model needs some components to make it work. The architecture of the Awareness Simulator Server (in Chapter 4) provides an idea of the required components. These are required components such as a communicator to handle incoming events and propagate information, a controller to handle the logic as to when the awareness model should be accessed and for what reasons (retrieve, add, modify, delete its contents), and the data access layer that actually performs those actions on the awareness model store. In earlier work by the author [Bharadwaj and Reddy 2003], the

awareness model architecture with essential components required to integrate various sources is suggested. This includes components for recording the meta-information for each source in the form of a manifest for that source, creation and access of physical storage for the values of those meta-information attributes and others functions. This architecture was suggested keeping in the mind the process of integration and change management that is discussed below. However much work is to be done to extend those concepts and create a awareness model architecture that is comprehensive.

With the implementation of the Awareness Simulator a first step of sorts has already been taken in this direction. The experience obtained in developing the Awareness Simulator is very useful in the effort to create an awareness framework. This is so because with respect to the awareness model and the awareness map, some of the main design principles that pertain to the awareness framework were dealt with in developing the simulator. It would be interesting to see how much of the Awareness Simulator core can be used in creating the awareness framework.

5.3.1.2 Towards a Generic Robust Efficient Awareness Model Implementation

A very important item of work is in creating an efficient robust implementation of the awareness model. The awareness model is a concept and with certain desired features. It can be implemented in different ways. Depending on the requirements of the collaboration various features can be chosen in its implementation. However there are certain generic capabilities that the awareness model must have, such as the ability to integrate and represent information from a variety of sources and about users, and define the manner in which the sources are connected, privileges and rules of awareness propagation. It must have mechanisms to handle changes to sources and users as well as incorporate new sources with new meta-information attributes that cannot be anticipated. The key abilities of the awareness model must be generic.

The example in Chapter 3 and 4 involved an RDBMS implementation. There are many benefits to such an approach. However further investigation is needed to determine the scalability and efficiency of the data model discussed. Also in simulation described, the awareness model was accessed upon each user-generated event. This may not be the most efficient solution for large groups with many applications, and high frequency of events being generated. As an alternative, sources may keep a local copy of the awareness model rules that affect them. Even at the server level, an RDBMS could be used only for persistent storage and programs could use main memory itself to improve efficiency. If found useful, markup languages (XML) could be utilized to represent the awareness model information. Thus it is essential to consider alternative approaches to implementation. The end result would be different implementation choices but with the same generic capabilities. The most important criterion in determining if a technology is suitable would be the support it provides for the generic features that the awareness model must possess.

5.3.1.3 Investigating Intelligent Agent Support

The intention of the awareness model is to provide a cohesive picture of all information sources involved in the collaboration process. Based on this cohesive view, humans can make decisions to tailor the quality of awareness they receive. The role of

intelligent agent programs has been discussed in this aspect to enhance this empowerment of the user's voluntary abilities with behind the scenes help.

Intelligent agents should be able to act based on the same awareness model as humans do. But to enable that, the information may have to be represented in a manner or transformed so that agents can interpret and execute sophisticated techniques, such as inferring the relevancy of new available and then suggesting it to the human in the manner most suitable (unobtrusive yet informative). Inference is invaluable since the awareness model information does not itself explicitly contain relevancy and other subjective quality factors. It is up to the eye of the agent to recognize them. The agents have to be cognizant of not only the awareness model and its changes, but also the profiles of the human users they work for, and their current state and activities. Creating mechanisms and agents that can do the above, is an interesting and extremely challenging area of work that needs to be pursued.

5.3.1.4 Process for Integration and Change

In order to manage the integration of new sources and users and the changes that occur to them and other aspects of collaboration, a streamlined process is needed. The process has to be easy to incorporate with adequate tool support. This is essential if the awareness framework concept is to be widely adopted in any collaboration. If users perceive that setup and integration require significant amount of effort, it would dissuade them from using an awareness framework and model.

5.3.1.5 Different Visualizations for the Awareness Map Concept

The awareness map concept needs to be investigated further. As different client applications (they are sources too) access the awareness map for their users, the same information may have to be represented according to the limitations of the device and application. Thus making the awareness map as comprehensive and effective as possible in spite of the limitations of the screen and display capabilities of the application is a challenge.

Apart from that we need to investigate how the awareness map itself could display information in a manner that is less distracting but at the same time does enough to catch the user's attention. Icons representing sources of information that are of less relevance could diminish and go into the background. Those that are more relevant could expand, change color and so on. This is particularly effective if intelligent agents are working in the background and based on their inference suggesting to the user various sources and new information that is available. Changes in color, size, shape of the icons, callouts (balloons), scrolling tickers, sound, even changes in dimension (objects appearing in3D) and other effects could be used as means to display awareness information more effectively.

5.3.1.6 Towards an Awareness Framework Standard(s)

All the above work in creating an awareness framework could lead towards establishing a standard for the awareness framework. It is more likely that a set of standards will be needed since there are various components involved. Every component in the awareness framework, including the interfaces could all adhere to standards so they

can be easily incorporated into awareness frameworks. These standards could be at the application layer, but could utilize already established standards such as those used in the Internet. This is a major step in addressing the lack of interoperability in awareness propagation. With widespread adoption the potential positive impact on collaborative work within and across organizations as well as within and across domains is tremendous.

5.3.2 Awareness Evaluation

5.3.2.1 Awareness Evaluation theory

The experimental methodology proposed in Chapter4 for awareness evaluation discussed concepts such as Work Process Model (WPM), Work Process Model Graphs and Work Practice Model (WPRM). Currently still in their infancy these bear much investigation. If these formalisms can be developed they could be used in not only in the awareness evaluation process as described in Chapter 4 but also in modeling awareness in collaboration on the whole. Awareness is itself difficult to characterize, and especially in collaboration. With the involvement of multiple people, places, tasks, artifacts, constraints and various other elements, gauging the impact of awareness is even more challenging. Thus any progress towards modeling at least some of it aspects in collaboration would be very useful in aiding the design and development of awareness systems. Tool support to create such models would be essential part of the research. For example, automatic creation of Work Practice Models from the simulation data in the Awareness Simulator or from any log that records collaborative activities would be invaluable in the analysis of awareness propagation and the role it played in the collaborative process. Just as the Unified Modeling Language tools have made working with UML models so easy and hence popular these tools could be used to create the above mentioned WPM and WPRM collaboration models.

5.3.2.2 Awareness Simulator

Various improvements can be made to the Awareness Simulator application itself. These are follows:

1. Provide a report viewer for the event log in the application itself. The reports could be customized to provide various views of the simulation data that has been collected. We are currently using the MySQL Query Browser, which is quite sufficient for most queries, but graphical representations are always beneficial as they provide a different perspective.

2. Provide more meta-information fields to describe source and medium characteristics, so that users have more information when they need to select sources.

3. Provide a "memory" for information sources so that users can search for information generated by sources in the past. For instance one can search email messages stored on an email server to obtain historical awareness of the some aspect of collaboration. One can also search the awareness model itself to find out who else was aware of the

information generated by that source in the past (this would involve finding out every user's name in whose foci that source was in, at that time). In fact this is possible even now, as the event log table (event_log_t) can be queried. However no user interface has been provided for such queries. This is one area of immediate improvement.

4. Create an administrator role so that new sources and users can be added along with the appropriate meta-information descriptions. Currently sources and users have to be added to the awareness model via the MySQL interface and ten the simulation has to be started. Adding the source sand users dynamically would be more helpful.

5. Also users themselves should be given certain privileges such as modifying their own profile, and privileges o information sources they own. This feature and the one above (administrator role) should be present in any awareness model implementation.

6. As mentioned in the section above on improving the awareness model (section 5.3.1.2), one must be able to create new scenarios easily by creating the awareness model through the simulator interface. Currently new scenarios can be run, by creating a new database thorough MySQL interface. Also the data access layer has to be modified if the database schema changes. This is not too efficient when one wishes to conduct a variety of simulations when making changes among the scenarios. A very useful feature would be the ability to crate new awareness models and simulate them dynamically.

7. Create a more intuitive user-interface for interaction. Investigate how users interaction with multiple applications could b simulated better, by providing them the appropriate type of interfaces and generating data that is resembles actual information in a real application. Currently this simulation is through text messages that the user enters.

8. The awareness map itself could be improved in the client. As mentioned above in section 5.3.1.5, through various means the information could be displayed more effectively. Also one could simulate different views of the awareness maps that would appear had the user been on a specific device or application. This is possible through the use of emulators for handheld devices and "skins" to show the corresponding user interface. The awareness map information itself would have to be customized for display according to the device emulator.

9. Support for agent programs could be far more than having an automated client generate events. One could incorporate the agent-based simulation (ABS) paradigms to augment human users with agent users who collaborate and work with them off the same awareness model. This role of agent in place of the human, in the simulator, is different from the use of agents as helpers to humans (section 5.3.1.3). In fact both types could be incorporated in the simulator.

10. The overhead on users to enter the MM and MMA messages should be minimized and if possible avoided by the investigation of other methods that can be used to record information in a manner not to interrupt the user. However users should still have the option to enter such messages if they choose to. Some users may wish to record certain aspects not automatically collected by the simulator.

As mentioned briefly in Chapter 1 there is promise that the work done as part of this dissertation can lead to future work in these directions too.

5.3.3 Awareness Characterization and Awareness Patterns

In the attempt to devise solutions to address the research question, a characterization of the impact of heterogeneous information sources and mediums on awareness quality has been gained. Furthermore the notion of awareness attributes has been proposed. This is very useful in characterizing awareness information that is to be propagated. These findings could aid the overall research challenges in the field of awareness. They could promote better design and development of awareness systems in CSCW. The validation methodology and framework have proposed a simple formalism to model awareness propagation (details in Chapter 4). This formalism could be extended and developed to model the flow of awareness information in collaboration. Just as software design can be factored into software design patterns, there is the idea that collaboration itself can be characterized as patterns. As mentioned above much work needs to be done to completely specify the formalism itself. However if the concept of 'awareness patterns' is realizable, then the formalism proposed in this research could be used as a means to specify such awareness patterns in collaboration. As mentioned in the section on awareness evaluation above, the tools created to model the work process and work practice model graphs could be extended to distill patterns from the graphs. The graphs themselves could be created manually or automatically be generated from existing data. This is an area of investigation that is to be pursued as part of future work.

5.3.4 EkSarva: The Awareness Model Component

Though the primary intention in designing the awareness framework and model was with respect to heterogeneous environments, an extremely useful application of the same would be to use it in a homogeneous collaboration system. One such effort currently underway is the EkSarva [EkSarva 2005] at SIPlab [SIPlab 2005] at West Virginia University. It would be interesting to incorporate the awareness model in the EkSarva environment. The objective of this project is to create a collaboration framework for adaptive workflow-centric applications that improve users collaboration by providing appropriate context awareness and support through intelligent agents.

5.4 Revisiting the "Philosophies": A Final Thought

Very broadly speaking, the objective of this dissertation was to propose the notion that heterogeneity can not only be dealt with but in fact actually exploited, by maintaining **oneness** through the propagation of effective awareness among various systems. The dissertation has aimed to establish this through the discussion of heterogeneity, the importance of awareness, and proposed solutions for awareness propagation and identified related issues that must be dealt with. In doing so a departure from the conventional methodology and practice of **all-in-one** systems and adoption of a **one-from-all** methodology is called for. With the wealth of applications and systems that can be exploited to enhance collaborative work, and the various standards and technologies that can be harnessed to create awareness frameworks, there is great promise that barriers in heterogeneous collaboration environments can be overcome.

Bibliography

[Anderson and Bouvin 2000a]
Anderson, K. M., and Bouvin, N. O. "*Enabling Project Awareness and Intersubjectivity via Hypermedia-Enabled Event Trails*". *Technical Report*, CU-CS-911-00, Department of Computer Science, University of Colorado, Boulder, December, 2000.
Available at:
http://www.cs.colorado.edu/current/department/publications/reports/docs/CU-CS-911-00.pdf
(Retrieved December 2005)

[Anderson and Bouvin 2000b]
Anderson, K.M., and Bouvin, N.O. "Supporting Project Awareness on the WWW with the iScent Framework". In the *ACM SIGGROUP Bulletin*, Volume 21, Issue 3, December 2000. ACM Press, New York, NY, USA, 2000, pp. 16-20.

[Bardram 2005]
Bardram, J.E. "The Java Context Awareness Framework (JCAF) – A Service Infrastructure and Programming Framework for Context-Aware Applications". In *Pervasive Computing: Third International Conference, PERVASIVE 2005, Munich, Germany, May 8-13, 2005. Proceedings,* Editors: Hans W. Gellersen, Roy Want, Albrecht Schmidt, Lecture Notes in Computer Science, Volume 3468, 2005. Springer-Verlag Berlin Heidelberg, 2005, pp. 98-115. (ISBN: 3-540-26008-0).

[Bardram and Bossen 2005]
Bardram, J. E. and Bossen, C. "A Web of Coordinative Artifacts: Collaborative Work at a Hospital Ward". In the *Proceedings of the 2005 International ACM SIGGROUP Conference on Supporting Group Work (GROUP' 05),* (November 06 - 09, 2005, Sanibel Island, Florida, USA). ACM Press, New York, NY, USA, 2005, pp.168 – 176. (ISBN:1-59593-223-2).

[Bardram and Hansen 2004]
Bardram, J.E., and Hansen, T.R. "The AWARE Architecture: Supporting Context-Mediated Social Awareness in Mobile Cooperation". In the *Proceedings of the 2004 ACM conference on Computer supported cooperative (CSCW'04)*, (November 6-10, 2004, Chicago, Illinois, USA). ACM Press New York, NY USA, 2004, pp. 192-201.

[Benford and Fahlén 1993]
Benford, S.D. and Fahlén, L.E. "A Spatial Model of Interaction in Large Virtual Environments". In the *Proceeding of the 3rd European Conference on Computer Supported Cooperative Work (ECSCW'93)*(September 13-17, 1993, Milano, Italy), Editors: De Michelis, G., Simone, C. and Schmidt, K. Dordrecht. Kluwer Academic Publishers, pp.109-124.

[Benford, Bowers, Fahlén and Greenhalgh 1994]
Benford, S. D., Fahlén, L. E., Bowers, J. M., and Geenhalgh, C.M. "Managing Mutual Awareness in Collaborative Virtual Environments". In the *Proceedings of the ACM SIGCHI Symposium on Virtual Reality Software and Technology (VRST'94)*, (August 23-26th 1994, Singapore), ACM Press, New York, NY, 1994.

[Bharadwaj and Reddy 2003]
Bharadwaj, V. and Reddy, Y.V.R. "A Framework to Support Collaboration in Heterogeneous Environments", In the *ACM SIGGORUP Bulletin, Special issue on community-based learning: explorations into theoretical groundings, empirical findings and computer support, Session: Student Research Projects,* Volume 24, Issue 3, December 2003. ACM Press New York, NY, USA, 2003, pp. 103-116.

[Bharadwaj et al. 2004]
Bharadwaj, V., Reddy, Y.V.R., Kankanahalli, S., Reddy, S., Selliah, S., Yu, J. "Evaluating Adaptability in Frameworks that Support Morphing Collaboration Patterns". In *Proceedings of the 13th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE'04)*, (June 14 - 16, 2004, Modena, Italy). IEEE Press pp.186-191.

[Brush 2005]
Brush, A.J.B. "Terminology and Evaluation: Two Challenges for Awareness Systems Research". Position paper for *Proceedings of the SIGCHI conference on Human factors in computing systems* (*CHI 2005) Workshop on Awareness Systems: Known Results, Theory, Concepts and Future Challenges,* (April 02 – 07, 2005, Portland, Oregon, USA). (http://www.awareness-research.org/ Retrieved December 2005)
Available at: http://research.microsoft.com/~ajbrush/papers/BrushAwareness.pdf
(Retrieved December 2005)

[CERC 2005]
Concurrent Engineering Research Center. http://www.cerc.wvu.edu
(Retrieved December 2005)

[Christein and Schulthess 2002]
Christein, H., Schulthess, P. "A General Purpose Model for Presence Awareness".
In *Distributed Communities on the Web: 4th International Workshop, DCW 2002, Sydney, Australia, April 3-5, 2002. Revised Papers,* Editors: Plaice J. et al. Lecture Notes in Computer Science Volume 2468, Springer-Verlag, Berlin, Heidelberg, 2002, pp.22-34.

[CoPowerPoint 2005]
CoPowerPoint Demo Center. http://reduce.qpsf.edu.au/copowerpoint/
(Retrieved December 2005)

[CoWord 2005]
CoWord Demo Center. http://reduce.qpsf.edu.au/coword/
(Retrieved December 2005)

[Dey 2000]
Dey., A.K. "*Providing Architectural Support for Building Context-Aware Applications"*.
Ph.D. thesis, College of Computing, Georgia Institute of Technology, Atlanta, Georgia,
USA, December 2000.
Available at: http://www.cc.gatech.edu/fce/ctk/pubs/dey-thesis.pdf
(Retrieved December 2005)

[Dey 2001]
Dey, A K. "Understanding and Using Context". *Personal and Ubiquitous Computing
Journal*, Vol. 5, No.1, February 2001. Springer-Verlag London Ltd., pp. 4-7. (ISSN:
1617-4909)

[Dourish and Belloti 1992]
Dourish, P., and Bellotti, V. "Awareness and Coordination in Shared Workspaces".
In the *Proceedings of the 1992 ACM Conference on Computer-Supported Cooperative
Work (CSCW'92)*, (November 01-04,1992, Toronto, Ontario, Canada). ACM Press, New
York, NY, USA, 1992, pp107-114.

[eGroupware 2005]
eGroupware website. http://www.egroupware.org/
(Retrieved December 2005)

[EkSarva 2005]
EkSarva Project homepage. http://siplab.csee.wvu.edu/research/EkSarva/
(Retrieved December 2005)

[Ellis, Gibbs and Rein 1991]
Ellis, C. A., Gibbs, S. J., and Rein, G. L. "Groupware: Some Issues and Experiences".
Communications of the ACM, January 1991,Vol. 34, No. 1. ACM Press, New York, NY,
USA, pp. 39-58.

[Farshchian 2000]
Farshchian , B.A. "Gossip: An Awareness Engine for Increasing Product Awareness in
Distributed Development Projects". In *Advanced Information Systems Engineering: 12th
International Conference, CAiSE 2000, Stockholm, Sweden, June 2000. Proceedings,*
Editors: B. Wangler, L. Bergman, Lecture Notes in Computer Science, Volume 1789,
Springer-Verlag Berlin Heidelberg, 2000, pp. 264-278. (ISBN 3-540-67630-9).

[Fuchs, Pankoke-Babatz and Prinz 1995]
Fuchs, L., Pannkoke-Babatz U., and Prinz, W. "Supporting Cooperative Awareness with
Local Event Mechanisms: The GroupDesk System". In the *Proceedings of the Fourth
Fourth European Conference on Computer Supported Cooperative Work (ECSCW '95)*
(September 11-15, 1995, Stockholm, Sweden). Kluwer Academic Publishers, Dordrecht,
1995, pp. 245-260.

[Fussell et al. 1998]
Fussell, S.R., Kraut, R.E., Lerch, F.J., Scherlis, W.L., McNally, M.M., and Cadiz, J.J. "Coordination, overload and team performance: Effects of team communication strategies". In the *Proceedings of the 1998 ACM conference on Computer supported cooperative work (CSCW '98),* (November 14-18, 1998, Seattle, Washington, USA). ACM Press, New York, NY, USA, pp. 275-284. (ISBN: 1-58113-009-0)

[Gnomemeeting 2005]
Gnomemeeting System Website. http://www.gnomemeeting.org
(Retrieved December 2005).

[Greenberg, Gutwin and Cockburn 1996]
Greenberg, S., Gutwin, C., and Cockburn, A. "Using Distortion-Oriented Displays to Support Workspace Awareness". In *People and Computers XI, Proceedings of HCI '96,* Editors: Martina Angela Sasse, Jim Cunningham, Russel L. Winder. Springer 1996, pp.299-314. (ISBN 3-540-76069-5)

[Groove Networks 2005]
Groove Networks Website. http://www.groove.net/home/index.cfm
(Retrieved December 2005).

[Gross 2004]
Gross, T. "The Past, Present, and Future of Workgroups in a Theatre of Work". In the *Proceedings of the Eleventh International Information Management Talks*, *(IDIMT 2004)* (Sept. 15-17, Budweis, Czech Republic). Universitaetsverlag Rudolf Trauner, Linz, 2004. pp. 81-90. (ISBN: 3-85487-665-3).

[Gross and Specht 2001]
Gross, T. and Specht, M. "Awareness in Context-Aware Information Systems". *In Mensch & Computer - 1. Fachuebergreifende Konferenz* (Mar. 5-8, Bad Honnef, Germany). Editors: Oberquelle, H., Oppermann, R. and Krause, J. Teubner, 2001. pp. 173-182.
Available at: http://www.uni-weimar.de/~gross/publ/mc01_gross_specht.pdf
(Retrieved December 2005).

[Gross, Stary and Totter 2005]
Gross, T., Stary, C. and Totter, A. "User-Centered Awareness in Computer-Supported Cooperative Work-Systems: Structured Embedding of Findings from Social Sciences". *International Journal of Human-Computer Interaction* 18, 3 (June 2005). Lawrence Erlbaum Associates, Inc., Mahwah, NJ, USA, 2005, pp. 323-360. (ISSN: 1044-7318).

[Gross, Wirsam and Graether 2003]
Gross, T., Wirsam, W., and Graether, W. "AwarenessMaps: Visualizing Awareness in Shared Workspaces". In the *Proceedings of the CHI '03 extended abstracts on Human factors in computing systems (CHI'03),* (April 05 - 10, 2003, Ft. Lauderdale, Florida, USA) 2003, ACM Press, New York, NY, USA, 2003, pp. 784 - 785. (ISBN:1-58113-637-4).

[Gutwin and Greenberg 2002]
Gutwin, C., and Greenberg, S. "A Descriptive Framework of Workspace Awareness for Real-Time Groupware". *Computer Supported Cooperative Work* Journal, 11, (2002). Kluwer Academic Publishers, Netherlands, 2002, pp. 411-446.

[IBM Lotus 2005]
IBM Lotus Suite homepage. http://www-306.ibm.com/software/lotus/
(Retrieved December 2005)

[IBM WebSphere 2005]
IBM WebSphere homepage. http://www-306.ibm.com/software/websphere/
(Retrieved December 2005)

[IBM Workplace 2005]
IBM Workplace Collaboration Services homepage. http://www-142.ibm.com/software/workplace/products/product5.nsf/wdocs/workplacehome
(Retrieved December 2005)

[IETF RFC 2778]: Available at http://www.ietf.org/rfc/rfc2778.txt
(Retrieved December 2005)

[Kantor, Zimmerman and Redmiles 1997]
Kantor, M., Zimmermann, B., and Redmiles, D. "From Group Memory to Project Awareness Through Use of the Knowledge Depot". In the *Proceedings of the 1997 California Software Symposium (CSS97)* (November 7, 1997, Irvine, CA, USA). UCI Irvine Research Unit in Software, Irvine, CA, November, 1997, pp. 19-26.

[Kim 2004]
Kim, E. E. "A Manifesto for Collaborative Tools". *Dr. Dobbs Journal,* May 2004. Available at: http://www.ddj.com/documents/s=9132/ddj0405e/0405e.htm
(Retrieved December 2005)

[Kirsh 2000]
Kirsh, D. "A few thoughts on Cognitive Overload". *Intellectica,* 1, 30, *(*2000). pp 19-51. Available at: http://adrenaline.ucsd.edu/kirsh/articles/overload/published.html
(Retrieved December 2005)

[Leiva-Lobos and Covarrubias 2002]
Leiva-Lobos, E. P., Covarrubias, E. "The 3-Ontology: A Framework to Place Cooperative Awareness". In *Groupware: Design, Implementation and Use: 8th International Workshop, CRIWG 2002, La Serena, Chile, September 1-4, 2002. Proceedings*, Editors: Haake, J.M. and Pino, J.A. Lecture Notes in Computer Science, Volume 2440, Springer-Verlag Berlin Heidelberg (January 2002) pp.189-199

[Li 2005]
Dr. Du Li Projects page. http://cocasoft.cs.tamu.edu/~lidu/projects/Poster05.pdf
(Retrieved December 2005)

[Lucane 2005]
Lucane - Open-Source Collaborative PlatformWebsite. http://www.lucane.org/EN/
(Retrieved December 2005)

[Mangan, Borges and Werner 2004]
Mangan, M. A. S., Borges, M. R. S., and Werner, C. M. L. "A Middleware to Increase Awareness in Distributed Software Development Workspaces". In the *Proceedings of the WebMedia & LA-Web 2004 Joint Conference 10th Brazilian Symposium on Multimedia and the Web 2nd Latin American Web Congress (LA-Webmedia'04)* (October 12 - 15, 2004, Ribeirão Preto-SP, Brazil). IEEE Press, 2004, USA, pp.62-64.

[MegaMeeting 2005]
MegaMeeting Product Homepage. http://www.megameeting.com/
(Retrieved December 2005)

[Merriam-Webster Online 2004]
Definition of the word 'awareness'. http://www.m-w.com/dictionary/awareness
(Retrieved December 2005)

[Microsoft Collaboration 2005]
Microsoft Collaboration Products Homepage. http://office.microsoft.com/en-us/default.aspx
(Retrieved December 2005)

[Microsoft Exchange 2005]
Microsoft Exchange Homepage. http://www.microsoft.com/exchange/default.mspx
(Retrieved December 2005)

[Microsoft and Groove 2005]
Microsoft and Groove Homepage. http://www.microsoft.com/office/groove/default.mspx
(Retrieved December 2005)

[Microsoft LiveMeeting 2005]
Microsft LiveMeeting Homepage.
http://www.microsoft.com/office/livemeeting/prodinfo/default.mspx
(Retrieved December 2005)

[Microsoft Office 2005]
Microsoft Office Homepage. http://office.microsoft.com/en-us/default.aspx
(Retrieved December 2005)

[MySQL 2005]
MySQL RDBMS Homepage. http://dev.mysql.com/
(Retrieved December 2005)

[Novell Groupwise 2005]
[Novell Groupwise Webpage. http://www.novell.com/products/groupwise/
(Retrieved December 2005)

[Nutter and Boldyreff 2003]
Nutter, D., Boldyreff, C. "Historical Awareness Support and Its Evaluation in
Collaborative Software Engineering". In *Proceedings of the Twelfth International
Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises
(WETICE'03)*, (June 09 - 11, 2003, Linz, Austria). IEEE Press, pp.171-176

[OMG 2005] CORBA faq from the Object Management Group's Website
http://www.omg.org/gettingstarted/corbafaq.htm
(Retrieved December 2005)

[Patterson 2004]
Patterson, D. "Show Me the Value - Improving Project Visibility Using a Web-based
Portal". Project Management Articles, Welcom. 2004.
Available at: http://www.welcom.com/content.cfm?page=484
(Retrieved December 2005)

[Prinz 1999]
Prinz, W. "NESSIE: An Awareness Environment for Cooperative Settings". *In
Proceedings of The Sixth European Conference on Computer Supported Cooperative
Work - ECSCW'99*. S. Bødker, M. Kyng, and K. Schmidt (eds.), Kluwer Academic
Publishers, 1999, pp 391-410.

[Prinz and Gross 2001]
Prinz, W. and Gross, T. "Ubiquitous Awareness of Cooperative Activities in a Theatre of Work". In Proceedings of *Fachtagung Arbeitsplatzcomputer: Pervasive Ubiquitous Computing - APC 2001* (Oct. 10-12, Munich, Germany). VDE Publisher, Berlin, Germany, 2001. pp. 135-144.(ISBN: 0-8007-2636-X).

[Ramloll and Mariani 1999]
Ramloll, R., and Mariani, J.M. " 'Moksha': Exploring Ubiquity in Event Filtration-Control at the Multi-user Desktop". In the *Proceedings of the International Joint Conference on Work Activities Coordination and Collaboration (WACC'99),* (February 22 - 25, 1999, San Francisco, California, United States). *ACM Press, New York, NY, USA, pp. 207-216.*

[Reddy, Selliah, Bharadwaj, Yu, Reddy S. and Kankanahalli 2004a]
Reddy, R., Selliah, S., Bharadwaj, V., Yu, J., Reddy, S. and Kankanahalli, S. "An Epistemological View of Collaboration", (paper and keynote address), In the *Proceedings of the Eighth International Conference on CSCW in Design (CSCWD 2004*, (May 26-28, 2004, Xiamen Province, Peoples Republic of China). IEEE Press, 2004.

[Reddy, Selliah, Bharadwaj, Yu, Kankanahalli and Reddy S. 2004b]
Reddy, Y.V., Selliah, S., Bharadwaj, V., Yu, J., Kankanahalli, S. and Reddy, S. "EkSarva: An Intelligent Collaboration Framework". In the *Proceedings of the IEEE Conference on Intelligent Systems,* (June 22- 24, 2004,Varna, Bulgaria). IEEE Press, 2004.

[Rodden 1993]
Rodden, T. "Populating the Application: A Model of Awareness for Cooperative Applications". In the *Proceedings of the 1996 ACM conference on Computer supported cooperative work (CSCW'96)*, (Boston, Massachusetts, United States). ACM Press, New York, NY, pp.87-96. (ISBN:0-89791-765-0).

[Sarma and Van der Hoek 2002]
Sarma, Anita, and van der Hoek, André (2002). "Palantír: Increasing Awareness in Distributed Software Development". In the *Proceedings of the 24th International Conference on Software Engineering (ICSE 2002), Workshop on Global Software Development of the Software* Engineering, (May 2002). IEEE Press, pp 28-32.
Available at: http://www.ics.uci.edu/~asarma/papers/IWGSD2002.pdf
(Retrieved December 2005)

[Schlichter, Koch and Bürger 1997]
Schlichter, J., Koch, M., and Bürger, M. "Workspace Awareness for Distributed Teams". *In Proceedings Workshop Coordination Technology for Collaborative Applications, Singapore, 1997,* Editors: W. Conen (ed.). Lecture Notes on Computer Science, Springer 1997.
Available at: http://www11.informatik.tu-muenchen.de/publications/html/Schlichter1997/
(Retrieved December 2005)

[Schmidt 2005] CORBA page from Douglas C. Schmidt's homepage
http://www.cs.wustl.edu/~schmidt/corba-overview.html
(Retrieved December 2005)

[Sierhuis and Clancey 2002]
Sierhuis, M. and Clancey, W. J. "Modeling and Simulating Work Practice: A human-centered method for work systems design", *IEEE Intelligent Systems*, Volume 17(5), 2002. IEEE Press.

[Simone and Bandini 1997]
Simone, C., Bandini, S. "Compositional features for promoting awareness within and across cooperative applications". In the *Proceedings of the International ACM SIGGROUP Conference on Supporting Group Work: The Integration Challenge, (GROUP'97)*, (November 16 - 19, 1997, Phoenix, Arizona, USA). ACM Press, New York, NY, USA, pp.358 – 367. (ISBN:0-89791-897-5)

[Singh and Huhns 2005]
Singh, M. P. and Huhns, M. N. *"Service-Oriented Computing: Semantics, Processes, Agents"*. John Wiley & Sons, Ltd., 2005. (ISBN: 0-470-09148-7)

[SIMPLE 2005]
SIMPLE Webpage. http://www.ietf.org/html.charters/simple-charter.html
(Retrieved December 2005)

[SIP 2005]
SIP Webpage. http://www.ietf.org/html.charters/sip-charter.html
(Retrieved December 2005)

[SIPLab 2005]
Smart Internet Programming Laboratory Website. http://siplab.csee.wvu.edu
(Retrieved December 2005)

[Steinfield, Jang and Pfaff 1999]
Steinfield, C., Jang, C., and Pfaff, B. "Supporting Virtual Team Collaboration: The TeamSCOPE System".
In the *Proceedings of the international ACM SIGGROUP Conference on Supporting Group Work (Group'99)*(November 14-17, 1999, Phoenix, Arizona, USA). ACM Press, New York, NY, USA, 1999, pp 81-90.

[Totter, Gross and Stary 1998]
Totter, A., Gross, T. and Stary, C. "Functional versus Conscious Awareness in CSCW-Systems". In *Proceedings of the XV. IFIP World Computer Congress. Telecooperation--The Global Office, Teleworking and Communication Tools* (Aug. 31-Sep. 4, Vienna, Austria & Budapest, Hungary). Oldenbourg, Vienna, 1998. pp. 67-76. (ISBN: 3-85403-121-1).

[TOWER Project Website 2005]
http://www.fit.fraunhofer.de/projekte/tower/index_en.xml
(Retrieved December 2005)

Appendix A
Awareness Model Validation Simulation Data

For further details, information and any other questions regarding the awareness model validation data collected from all the simulations runs please contact the author at vanandb@hotmail.com.

Appendix B
Awareness Simulator User Manual

For further details, information and any other questions regarding the Awareness Simulator User Manual please contact the author at vanandb@hotmail.com.

Appendix C
Awareness Simulator Programmer Manual

A preliminary version of the Programmer Manual for the Awareness Simulator Application is included. Various details such the application installation and use, instructions to compile and create a new version, instructions for following the source code, including package and file information, details of all architectural components, algorithms with sequence diagrams, design issues, bugs and messaging protocol followed in the simulator are all described. This document is intended for users who wish to understand the internal workings of the simulator and extend the implementation.

For further details, information and any other questions regarding the Awareness Simulator Programmer manual please contact the author at vanandb@hotmail.com.

About this Document

This document was created as part of the Awareness Project.

Authors:
The Awareness Project Team members:
Vijayanand Bharadwaj, Madhukar Kakaraparthi, Adarsh Nara and Satyadev Adiraju from the Lane Department of Computer Science and Electrical Engineering at West Virginia University, Morgantown, West Virginia, USA.

Acknowledgements

The team members would like to specially thank *Madhukar Kakaraparthi* for his tremendous dedication and hard work beyond the call of duty in creating this Programmer's Manual.

Instructions to Run the Application

1. System Requirements

1. Intel Pentium M processor 1.60 GHz.
2. 512 MB RAM.
3. Monitor size 14.1 inch or greater

2. Software requirements

1. Windows XP
2. Java SDK SE v1.4.2_06
3. Visual Studio .Net Framework
4. MySql 4.1
5. For XML Parsing:- common-logging.jar, oscore-2.2.1.jar
6. My Sql connector to java:- mysql-connector-java-3.1.10-bin.jar
7. Files required for SWT: - swt.jar, swt-awt-win32-3138.dll, swt-gdip-win32-3138.dll, and swt-win32-3138.dll.

Note: - All the required software except the Visual studio .Net is available along with the programmer's manual.

3. Installation and Configuration for Running the Software

Software Requirements for Running Server

1) Download and install Java SDK 1.4.2_06.
2) Download the "EXECUTABLES" folder provided with the programmers manual.
3) Download and Install Mysql 4.1 with Administrator and Query Analyzer

File Structure

```
awproject
  |----EXECUTABLES
      |-----APSimClient SetUp
                  |---------Debug
                        |---------ApSimClient.exe
                        |---------setup.exe
      |-----APSimServer Installer
                  |---------Database Installation Script
                  |---------Simulator Script
                  |---------APSimServer.bat
```

Installing Mysql

Download and extract mysql-4.1.15-win32.zip file from
http://dev.mysql.com/downloads/mysql/4.1.html. Unzip and click on the setup file for
installing mysql 4.1 . Leave all the default options except for the choosing the standard
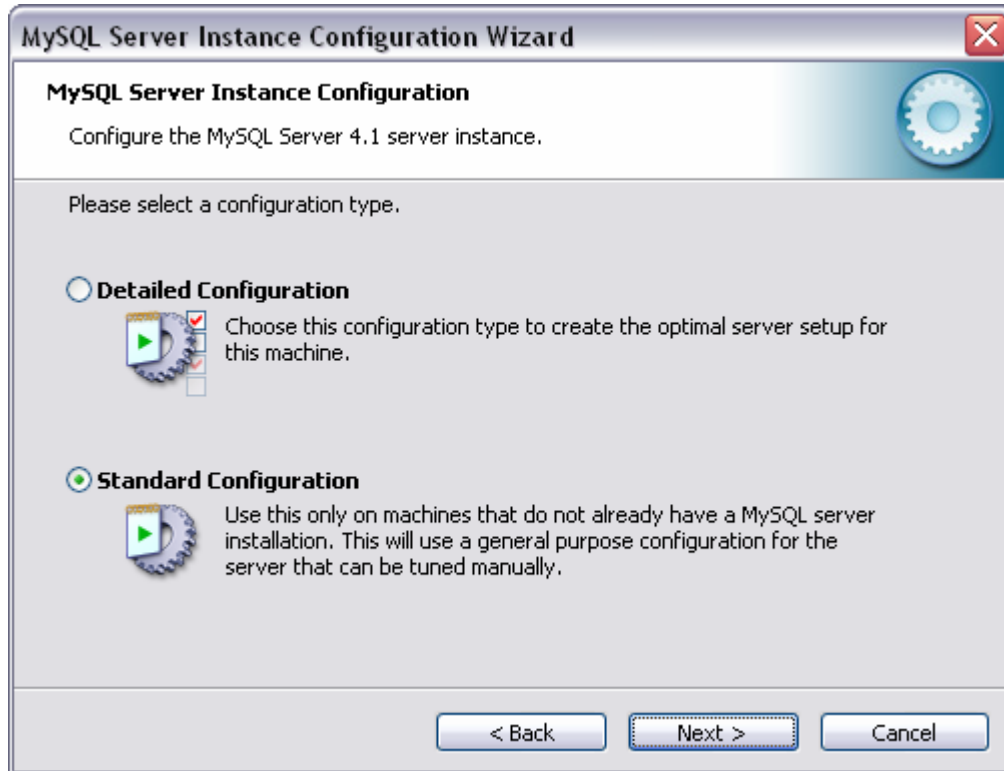configuration as shown in the screen shot below.

Figure 1

Download and install Mysql administrator and Mysql query browser. These tools help us
to use Mysql efficiently. Below is a screen shot of the Mysql Administration.
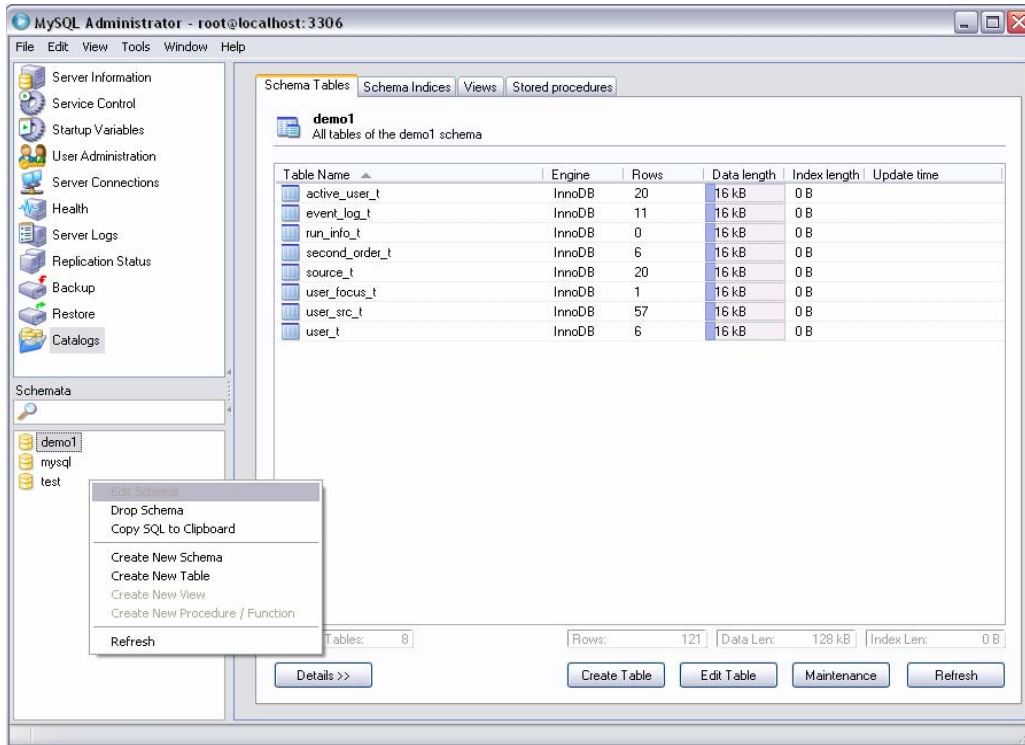
Figure 2 Administrator

The Pop-up menu can be used to create New Schema and new tables in the schema's. The screen shot shows all the tables in the schema demo1. The *"mysql"* schema which comes along with the installation is used as the default schema to connect to the MYSQL database. *"mysql"* contains meta data about all the schemas. Instructions to Run the Awareness Server:

Installing Simulator Client

In the Executables provided go to \EXECUTABLES\APSimClient Setup\Debug and we can see a setup.exe file and a APSimClient Setup.msi file. The setup.exe file is a boot strap installation files which pops open the location from which we can download and install the .NET framework. The Bootstrap installer works only in a machine with Windows XP Professional. If .Net frame work is already installed then both setup.exe and APSimClient.msi files work in the same way installing the APSimClient and adding a short cut of the Simulator Client to the All Programs. The default path where the Simulator client is installed is \program files\Default company Name\APSimClient Setup. This path is to be remembered because the collaboration files on the client side are stored in this location.

Installing Simulator Server

There is no need to install the Simulator Server.
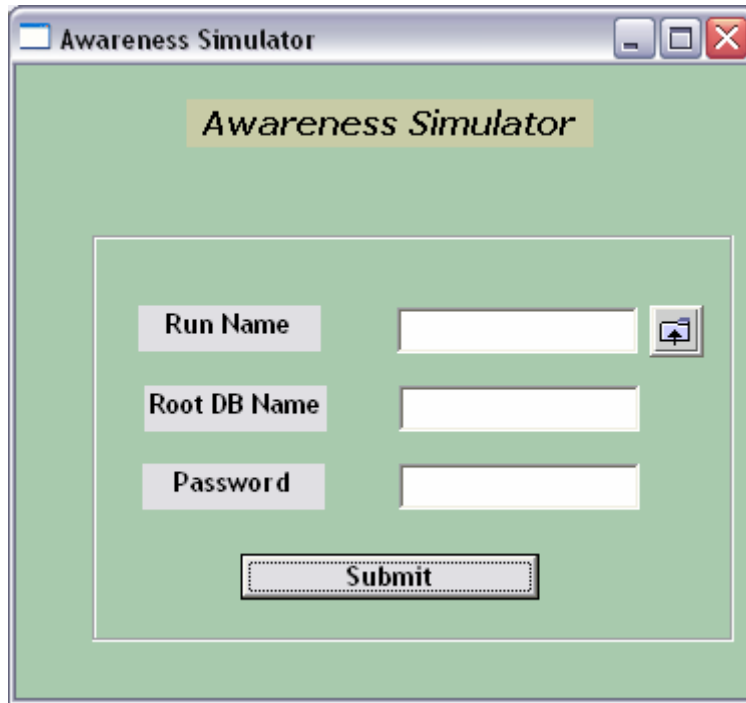
Figure 3

4. Launch and Run the Simulator:

Running Server

To Run the server we need to go to EXECUTABLES->APSimServer Installer and double click on the APSimServer.bat file. This will launch the Awareness Simulator Server. Figure 1.2 shows the Awareness Simulator Server initial login GUI. The User has to enter the following details to get started of:

1) The rootDBname/password are the credentials needed to login to the Mysql administrator.
2) runName is the name of the simulation that is being run. This will be used as the name of the database created in Mysql to store the simulation results
3) The scripts for the creation of the initial data needs to be chosen from the \EXECUTABLES\APSimServer Installer\Database Initialization Script - folder The script name is Initialization.sql. This is chosen by clicking on the button on the right hand side of the runName Label.
4) After choosing the Initialization script, submit button is clicked.

This would result in the creation of the Run Database(with name runName). With all the Initial tables created with the initial data.

Running Client

When Simulator Client is installed in the previous step a short cut is created in the Start->All Programs. To run the Simulator Client we just need to double click on the short cut added.

By Products

All the server side events are stored in the Mysql database in the table EventLog. The collaborative events on the client side are stored in a text file. These files can be later used for analysis purposes. The client side collaborative events are stored in the debug folder of the simulator client installation. The name of the file is the userIdrunName.txt. For example if the runName is demo1 and the emp has logged in. Then a file with name empdemo1.txt is created at C:\Program Files\Default Company Name\APSimClient Setup\ assuming that we have installed APSimClient at the default path.

Instructions To Compile & Run The Application

1. System Requirements

1. Intel Pentium M processor 1.60 GHz.
2. 512 MB RAM.
3. Monitor size 14.1 inch.

2. Software requirements

1. Windows XP
2. Eclipse 3.1 with visual editor (VE 1.0.1)
3. Java SDK SE v1.4.2_06
4. Visual Studio .Net
5. MySql 4.1
6. For XML Parsing:- common-logging.jar, oscore-2.2.1.jar
7. My Sql connector to java:- mysql-connector-java-3.1.10-bin.jar
8. Files required for SWT: - swt.jar, swt-awt-win32-3138.dll, swt-gdip-win32-3138.dll, and swt-win32-3138.dll.

This manual is written in great detail keeping in mind novice programmers who have little to no experience using the tools used for the development of the project, experienced programmers can skip these steps.

Note: - All the required software except the Visual studio .Net is available along with the programmer's manual.

3. Installation and Configuration for Development

Requirements for Running Server

1. Download and install Java SDK 1.4.2_06.
2. Download and install Eclipse 3.1. Install means just extract the contents. It will create a folder with name eclipse to run eclipse click on the eclipse.exe in this folder.
3. Download VE-runtime-1.1.0.1 and extract it into the eclipse folder. Visual Editor is to develop GUI in Eclipse
4. Download Mysql 4.1

File Structure

Simulator Client

\awproject\build and run folder\apsimclient
```
                              |-------------- APSimClient.csproj (project file)
                              |-------------- frmLogin.cs
                              |-------------- frmAwarenessClient.cs
                              |-------------- frmAwarenessMap.cs
                              |-------------- frmDisplayResult.cs
                              |-------------- frmResult.cs
                              |-------------- Parser.cs
                              |-------------- StringTokenizer.cs
                              |-------------- Token.cs
                              |-------------- UserController.cs
                              |-------------- UserListenerThread.cs
                              |-------------- MessageInfo.cs
                                         |--------------MessageInfo class
                                         |--------------UserSrcInfo class
                                         |--------------UserFocusInfo class
                                         |--------------FocusInfo class
```

The only file containing multiple classes is the MessageInfo.cs file all the other files contain just a single class.

*Simulator Server*

\awproject\build and run folder\awarenessWorkspace
```
                 |---------------------.metadata (Eclipse file to open the work
                                       space)
                 |---------------------awarenessproject
                                |---bin
                                     |--AwarenessCommunicator
                                            |--AMCommunicator.class
                                            |--ServerThread.class
                                     |--AwarenessController
                                            |--AMController.class
                                            |--MessageInfo.class
                                            |--ServerParser.class
                                            |--XmlParser.class
                                            |--User.class
                                     |--AwarenessModel
                                            |--AMAccessLayer.class
                                     |--AwarenessSimulator
                                            |--AMCommunicator.class
                                            |--ServerThread.class
                                |---src
```

|--AwarenessCommunicator
 |--AMCommunicator.java
 |--ServerThread.java
|--AwarenessController
 |--AMController.java
 |--MessageInfo.java
 |--ServerParser.java
 |--XmlParser.java
 |--User.java
|--AwarenessModel
 |--AMAccessLayer.java
|--AwarenessSimulator
 |--AMCommunicator.java
 |--ServerThread.java
|---commons-loggin.jar
|---jviewsall.jar
|---oscore-2.2.1.jar
|---mysql-connector-java-3.1.10-bin.jar
|---openfile.bmp
|---.classpath(Eclipse class path file)
|---.project(Eclipse project description)
**|---awarenessmanifest (manifest file for
 making jar files)**
|---awarenesssimulator.jardesc.

Installing Mysql

Download and extract mysql-4.1.15-win32.zip file from http://dev.mysql.com/downloads/mysql/4.1.html. Unzip and click on the setup file for installing mysql 4.1 . Leave all the default options except for the choosing the standard configuration as shown in the screen shot below.
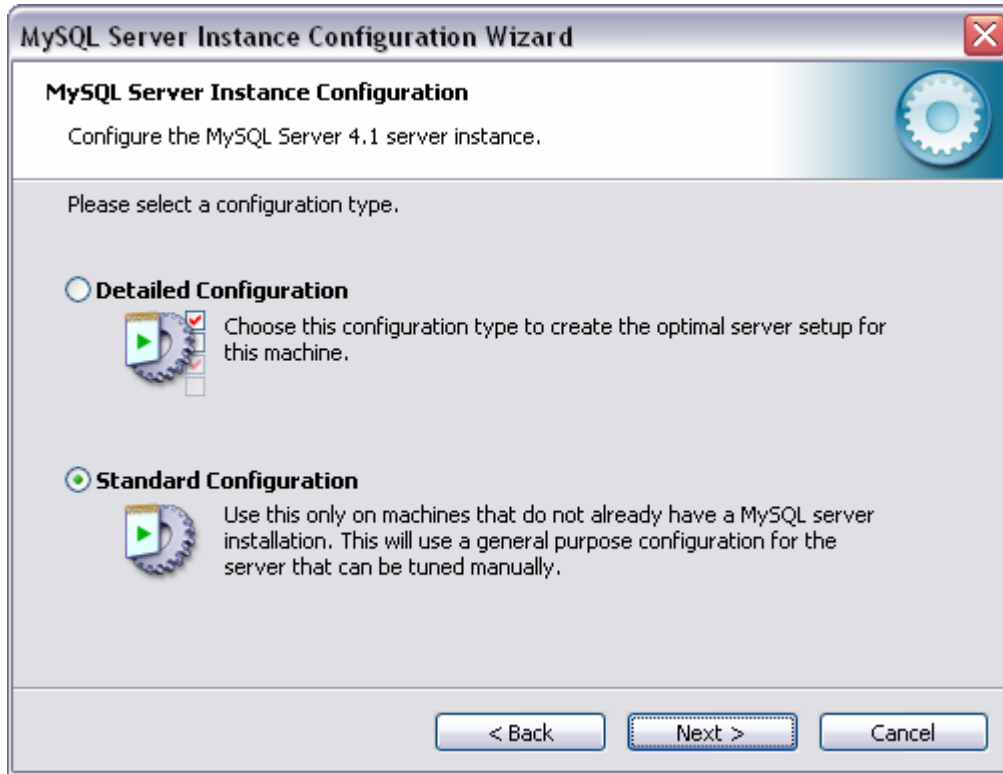
Download and install Mysql administrator and Mysql query browser. These tools help us to use Mysql efficiently. Below is a screen shot of the Mysql Administration.
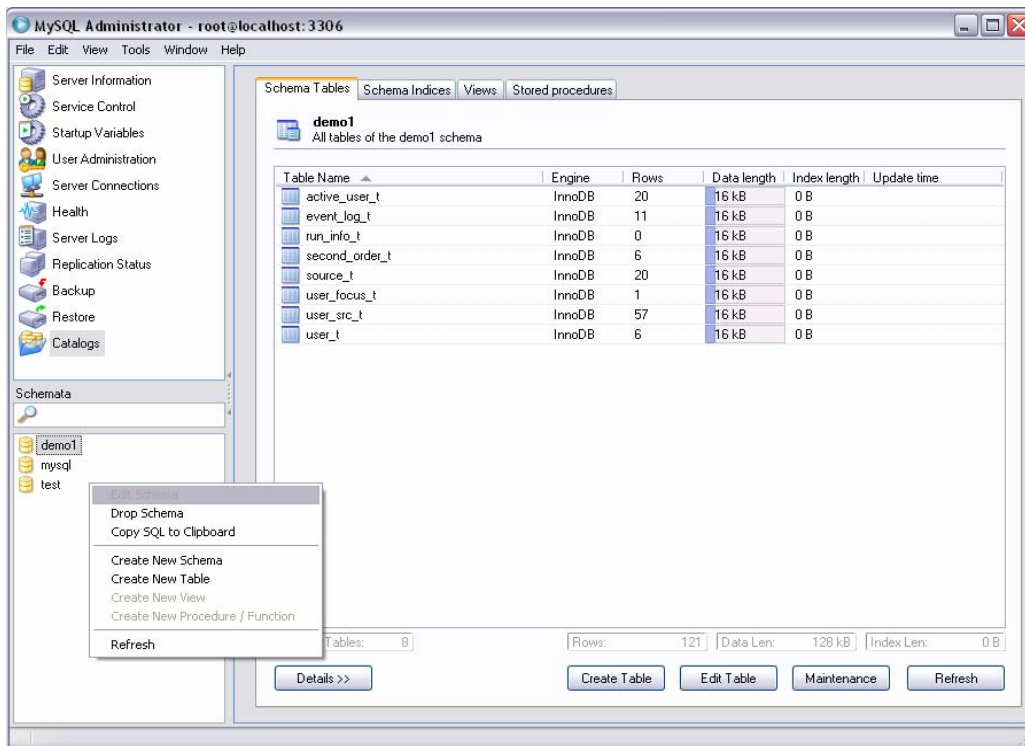

Figure 4 Administrator

The Pop-up menu can be used to create New Schema and new tables in the schema's. The screen shot shows all the tables in the schema demo1. The *"mysql"* schema which comes along with the installation is used as the default schema to connect to the MYSQL database. *"mysql"* contains meta data about all the schemas.

Configuring Eclipse

Install Eclipse. Download eclipse-SDK-3.1-win32.zip and extract it. It will create a folder with the name "Eclipse". Click on eclipse.exe in this folder to launch eclipse. Now download VE-runtime-1.1.0.1.zip and extract it into the "Eclipse" folder. Install *Visual editor* this involves installing updates from eclipse using Help -> Software Updates-> Find and Install. Choose updates for installed features. It gives a list of mirror sites choose one of them. Install all the updates. Now we are set to run the project.

Opening the Project (Awareness Server)

When you click on eclipse executable you will be given an option to choose a work space. Select the awarenessWorkspace which is given along with the programmer's manual.
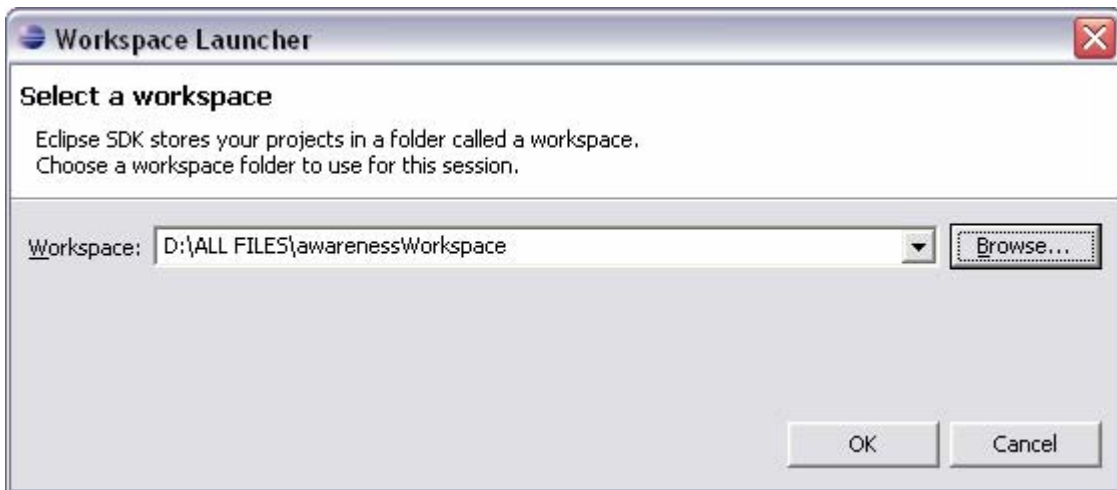

Figure 5

We would get the following screen. The right tab shows the awareness project with the following packages:-

1) Awareness Simulator
2) Awareness Model
3) Awareness Communicator
4) Awareness Controller

Each representing a module of the Architecture and is described in detail in later chapters.
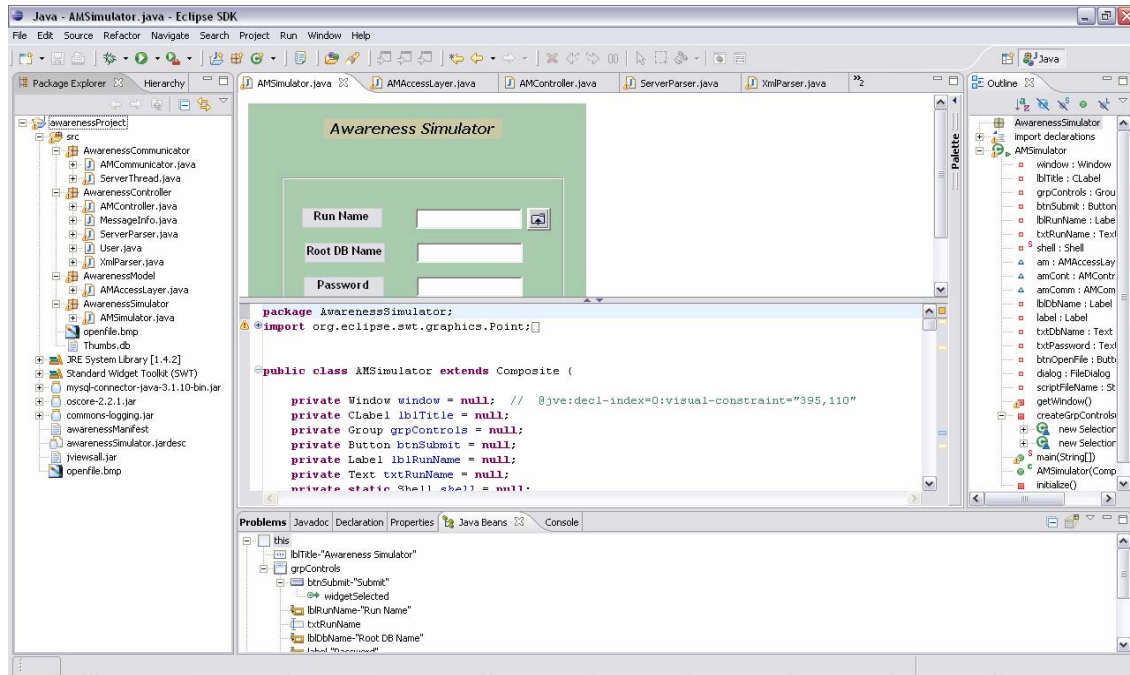

Figure 6

Compiling and Running Awareness Server:

If any changes are made to the awareness server. The application needs to be recompiled and executed. The default settings of eclipse provide for automatic compilation. To run the awareness server we need to choose the AMSimulator.java file and from menu's choose. Run->Run As->Java Bean. To run the application.

Making New Executable for Simulator Server for Submission

Once the changes are made to the existing project and a decision is made to submit the executable. A jar file should be made of the project. For this purpose select File->Export u will get a pop up menu select jarfile in that and click NEXT. This will give another pop up screen shown in figure 7 choose the options as shown in the figure. Give the jar file name as APSimServer.jar and save it at awproject\executables\apsimserver installer\simulator server\. This would replace the existing jar file. Click Next to get A Pop up Screen which looks like Figure 8. Choose the options shown in figure 8 and click Next. This would give a popup screen like Figure 9. Choose the options shown in Figure 9 and click Finish. Now we have to substitute the META-INF folder in the jar file. Open the jar file using WinRAR. This can be done by right clicking on the APSimServer.jar file and choose openWith option and choose winRar form the list of applications. This would result in the a pop up window like Figure 10. Delete the META-INF and the add the META-INF folder at \awproject.
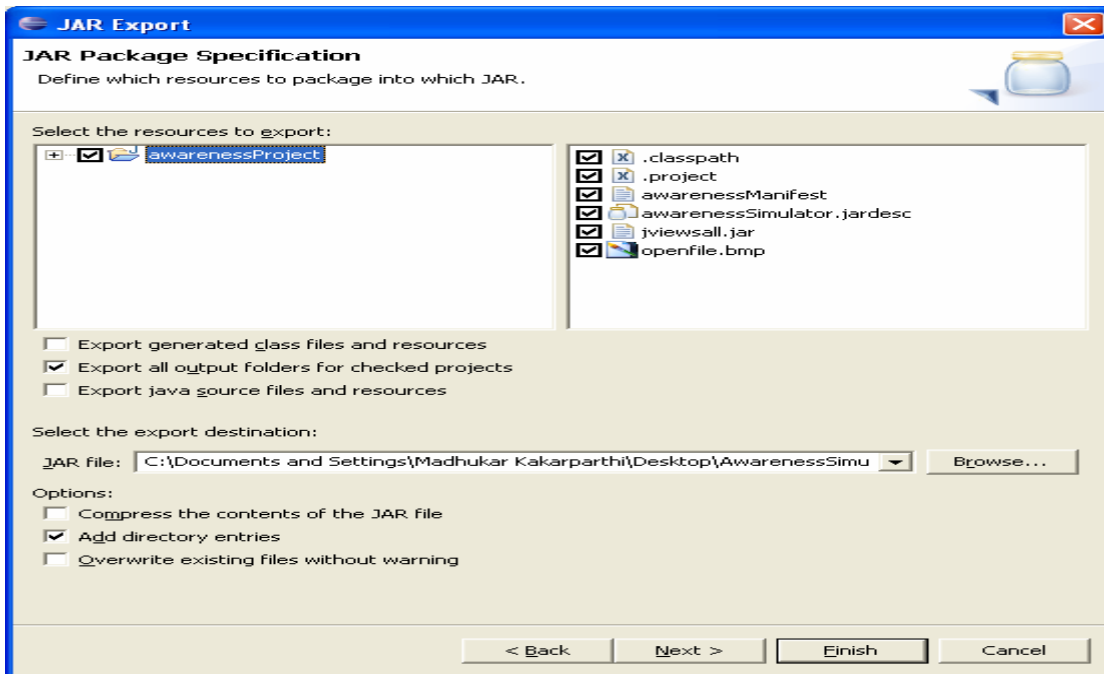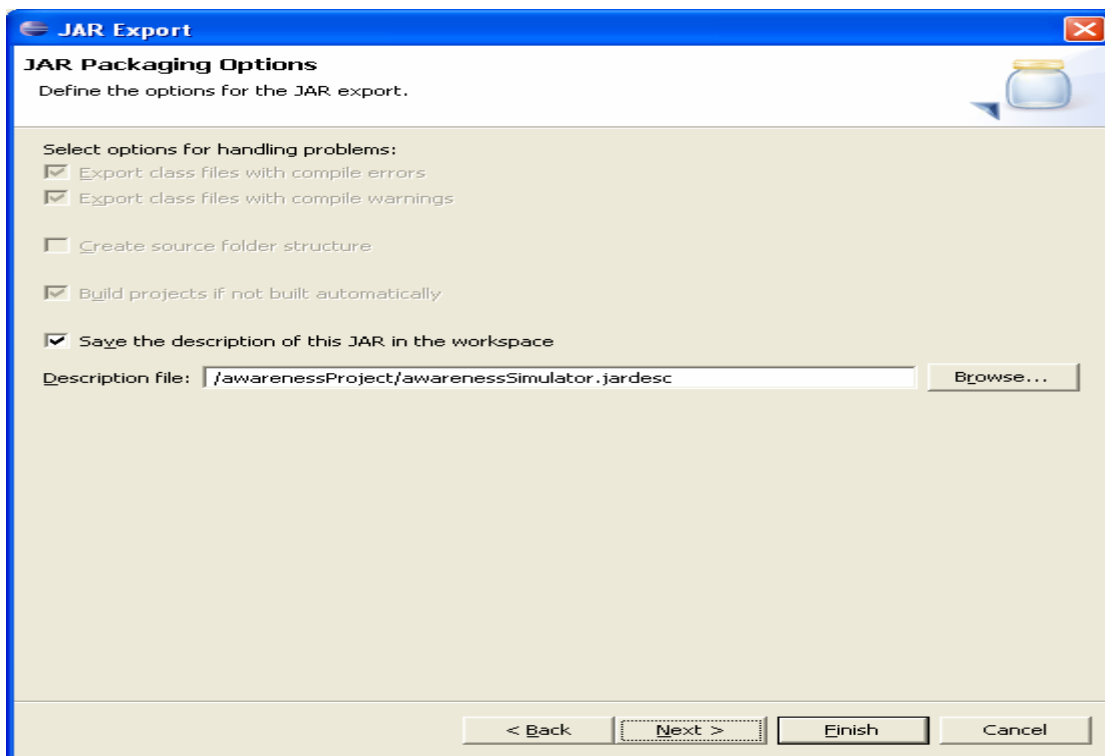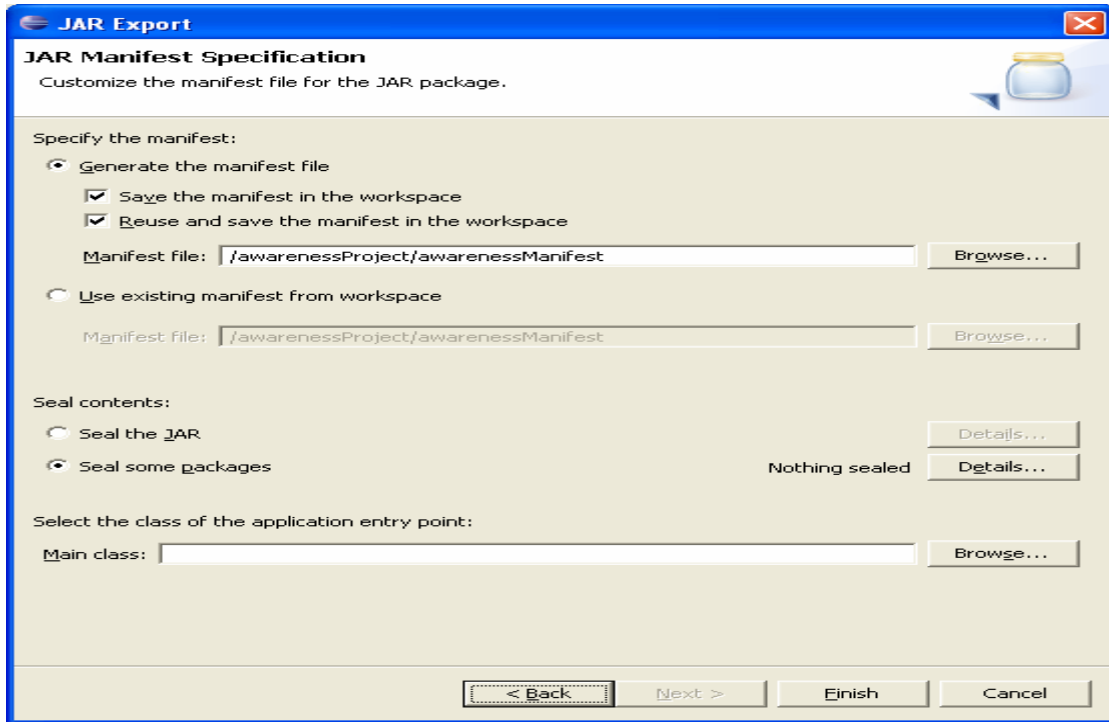
Figure7

Figure 8

Figure 9
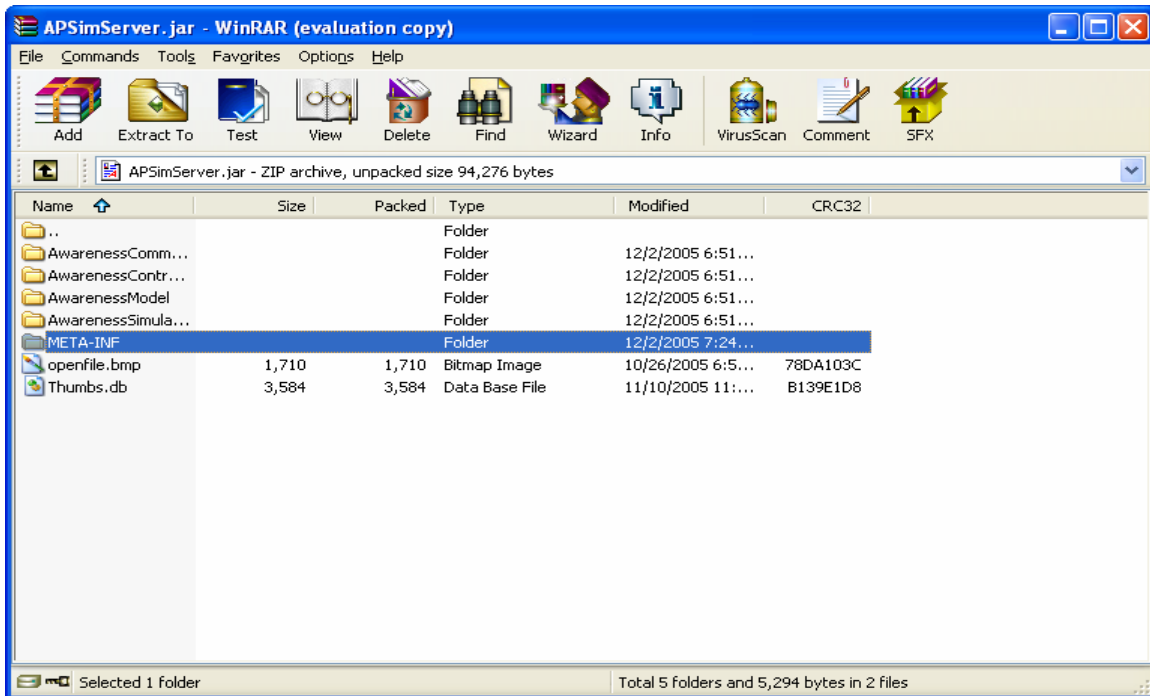

Figure 10

Opening the Project (Awareness Client)

All the source files for the Awareness Simulator Client are present in the awproject\Build And Executables\APSimClient\ folder. To open the project we need to install Visual Studio .Net and double click on the project file APSimClient.csproj. When the project is

opened it will show two project APSimClient and ApSimClient SetUp project. The second project is a setup and deployment project. If any changes are made to the 1st project second project should also be built so that we have the latest installation files. The second project is in the form of the folder \Build And Executables\APSimClient SetUp\ . This is the installation file and can be given to End Users for installation. Instructions for installation were given in details in the first section.

Compiling and Running Awareness Client

If any changes are made to the APSimClient project make sure to Build both the APSimClient project and APSimClient SetUp project to ensure that we have the latest setup files to be given to the End User.

Making New Executable for Simulator Client for Submission

If any changes are made to the APSimClient and the Final Executables are needed to be submitted. Then the /awproject/Build and Run Folder/APSimClient Setup folder needs to copied into executables directory to replace the existing directory. The release folder has no significance and hence can be removed.
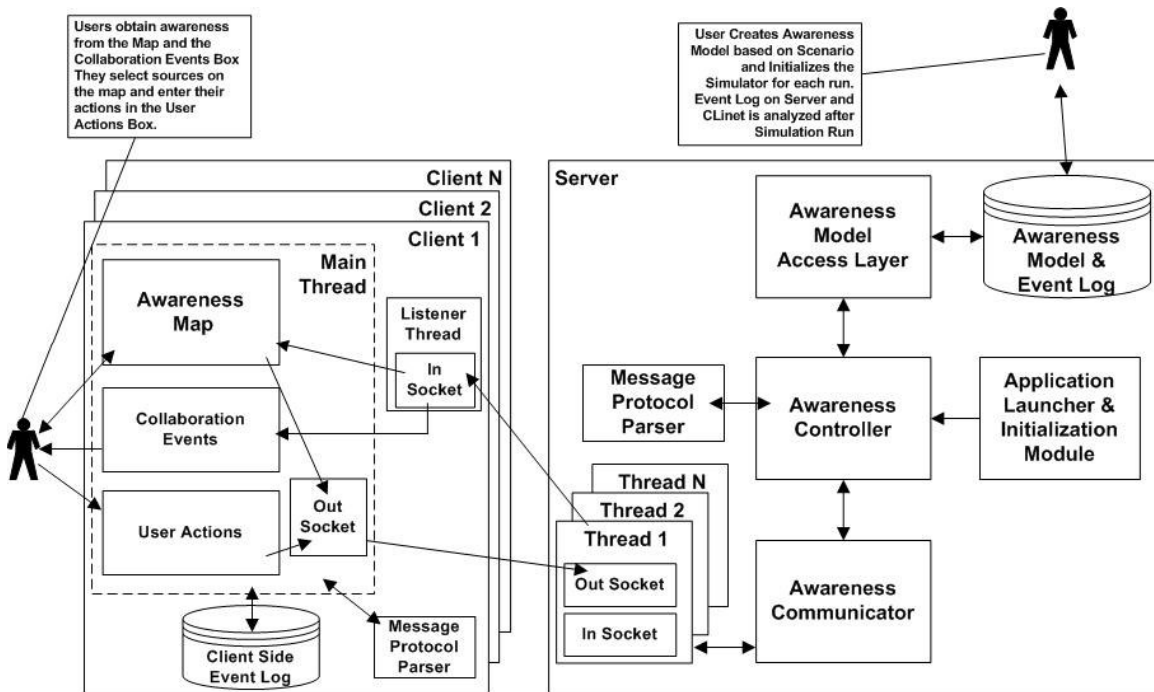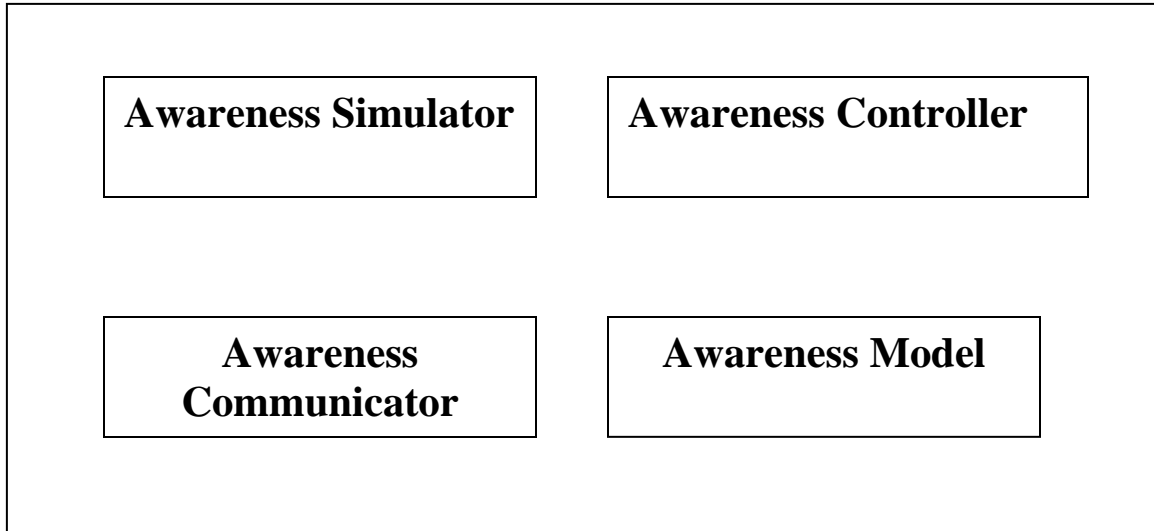
High Level Architecture

Figure 11

File Architecture View

Simulator Server File Architecture View

Server Side Package View with Files

```
┌─────────────────────────────────────────────────────────────┐
│                                                             │
│   ┌───────────────────────┐   ┌───────────────────────┐     │
│   │  Awareness Simulator  │   │  Awareness Controller │     │
│   │                       │   │                       │     │
│   └───────────────────────┘   └───────────────────────┘     │
│                                                             │
│   ┌───────────────────────┐   ┌───────────────────────┐     │
│   │      Awareness        │   │   Awareness Model     │     │
│   │     Communicator      │   │                       │     │
│   └───────────────────────┘   └───────────────────────┘     │
│                                                             │
└─────────────────────────────────────────────────────────────┘
```

Awareness Simulator Package

```
┌───────────────────────────┐
│   Awareness Simulator     │
│                           │
└──────────────┬────────────┘
               │
               │      ┌───────────────────────┐
               └─────▶│   AMSimulator.java    │
                      └───────────────────────┘
```
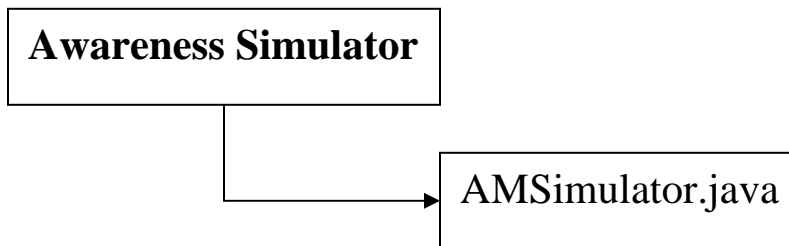
Awareness Controller Package

Awareness Controller

→ AMController.java

→ MessaeInfo.java
ServerParser.java
User.java

→ XmlParser.java

Awareness Communicator Package

Awareness Communicator

→ AMCommunicator.java

→ ServerThread.java
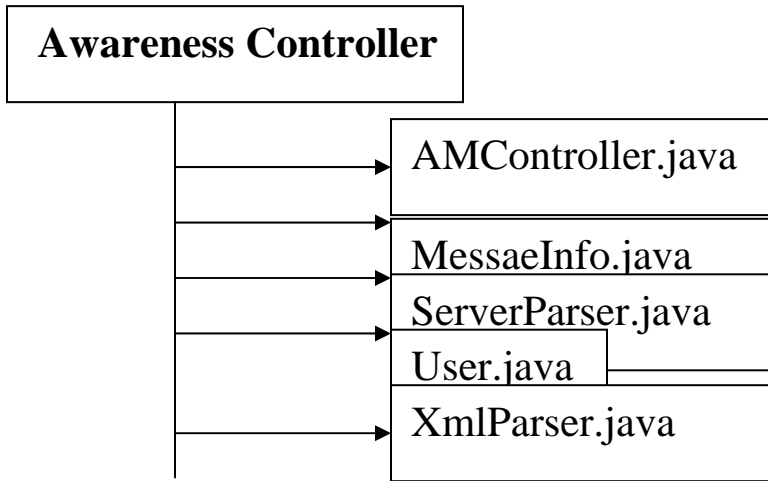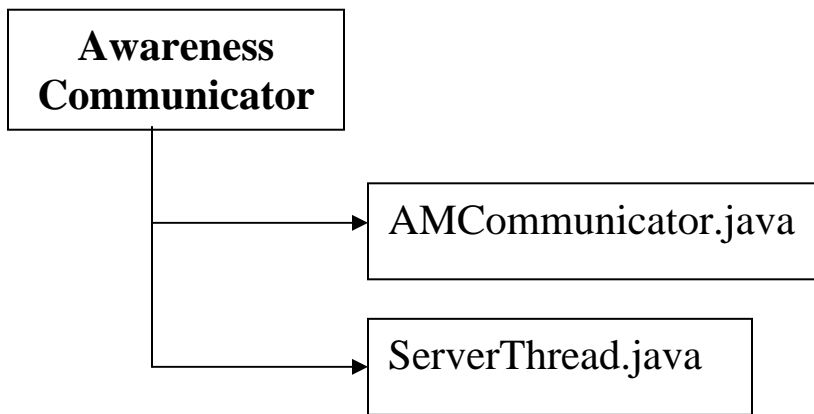
Awareness Model Package

Awareness Model

→ AMAccessLayer.java

Client Side

File View

```
┌─────────────────┐
│  APSimClient    │──────────────────────────┐
└─────────────────┘                          │
     │    ┌──────────────────────┐           │
     │───▶│  frmLogin.cs         │           │
     │    └──────────────────────┘           │
     │    ┌──────────────────────┐     ┌─────────────────────────┐
     │───▶│  frmAwarenessClient.cs│   ▶│  StringTokenizer.cs     │
     │    └──────────────────────┘     └─────────────────────────┘
     │    ┌──────────────────────┐     ┌─────────────────────────┐
     │───▶│  frmAwarenessMap.cs  │    ▶│  Token.cs               │
     │    └──────────────────────┘     └─────────────────────────┘
     │    ┌──────────────────────┐     ┌─────────────────────────┐
     │───▶│  frmResult.cs        │    ▶│  UserController.cs      │
     │    └──────────────────────┘     └─────────────────────────┘
     │    ┌──────────────────────┐     ┌─────────────────────────┐
     │───▶│  MessageInfo.cs      │    ▶│  UserLisetenerThread.cs │
     │    └──────────────────────┘     └─────────────────────────┘
     │    ┌──────────────────────┐
     │───▶│  Parser.cs           │
          └──────────────────────┘
```
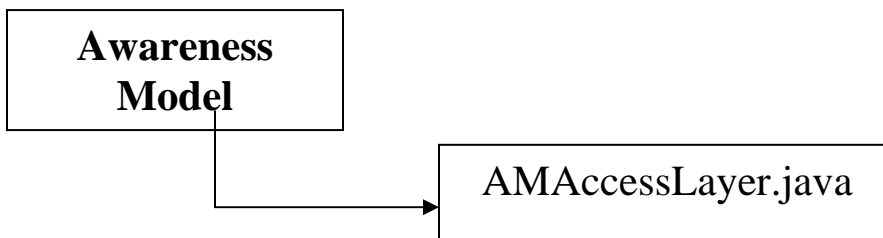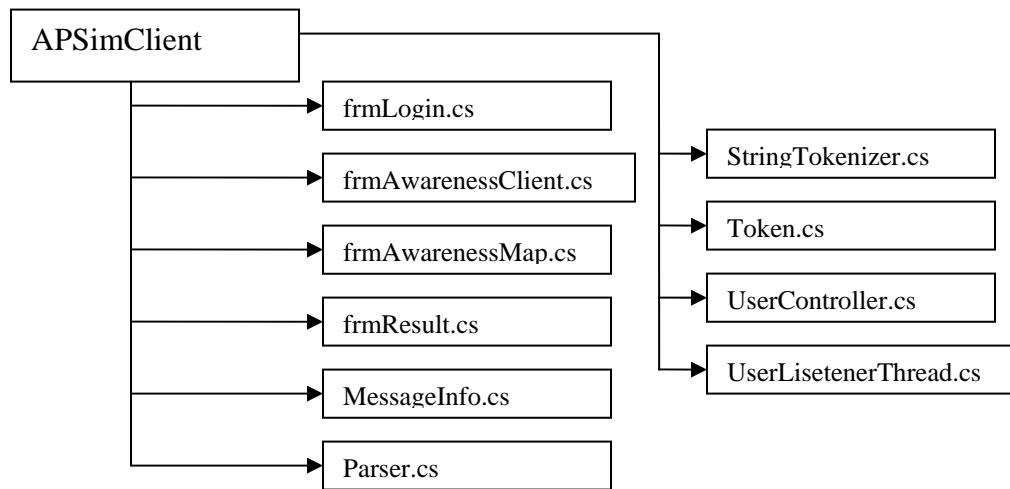
Algorithm Level

Authentication :

Client Side:

1. User Enters Login Details clicks submit.
2. Use the Ip Address to establish a connection with the Awareness Server. At the server side a thread(Server Thread) is created by the server to listen to the requests sent by this client.
3. The details are packaged into a Message and sent to the sendUserEvents function.
4. This message is packed in to a XMLMessage using the parser and is sent to the Awareness Model Server.

Server Side:

1. ServerThread(Part of the Awareness Communicator package) receives the Xml Message.
2. Sends the message to the AMController through the AMCommunicator's recieveMessage function
3. AMController uses the ServerParser to unpack the XML Message into MessageInfo object.
4. Use AMAccessLayer's UpdateEventLog method to log the incoming message.
5. handleAuthentication method is called which in turn accesses the AMAccessLayer's getPassword method to retrieve the password. Before calling the getPassword method the user details are added to the user list maintained by the Awareness Communicator.
6. password is compared and if it is equal then a Authentication reply message is sent with the reply being AUTHENTICATED the initial awareness map and the source super are also included in the Xml Message. If not a message is sent with reply as NOT_AUTHENTICATED. The message is sent by using the sendMessage method in the awareness communicator. If the user is not authenticated then the user is removed from the userList and the thread is aborted.
7. use AMAccessLayer's UpdateEventLog method to log the reply message.
8. A reply message is being sent basing on whether or not the focus is added by using the awareness communicator sendMessage function.

Client Side:

1. The User Listener Thread receives the message from the server and sends it to the User Controller.
2. The User Controller unpacks the message and depending on the message type performs necessary actions on the frmAwarenessClient (In this case updateAwarenessMap, updateSrcSuperSet, updateIpSet) and then displays the collaborative events in the colloaborative events window using the displayCollaborativeEvents function.

Add Focus

Client Side:

1. User Clicks on the Focus->AddFocus
2. Enters The foucsID, FocusName, Description and clicks AddFocus.
3. The details are sent to the userController addFocus function.
4. The details are packaged into a Message and sent to the sendUserEvents function.
5. This message is packed in to a XMLMessage using the parser and is sent to the Awareness Model Server

Server Side:

1. ServerThread (Part of the Awareness Communicator package) receives the Xml Message.
2. Sends the message to the AMController through the AMCommunicator's recieveMessage function
3. AMController uses the ServerParser to unpack the XML Message into MessageInfo object.
4. Use AMAccessLayer's UpdateEventLog method to log the incoming message.
5. addFocus method is called which in turn accesses the AMAccessLayer's insertFocusT method to update the awareness model.
6. use AMAccessLayer's UpdateEventLog method to log the reply message.
7. A reply message is being sent basing on whether or not the focus is added by using the awareness communicator sendMessage function.

Client Side:

1. The User Listener Thread receives the message from the server and sends it to the User Controller.
2. The User Controller unpacks the message and depending on the message type performs necessary actions on the frmAwarenessClient (In this case displayNewFocus, add the new focus to the focus list maintained by the user controller) and then display the collaborative events in the colloaborative events window using the displayCollaborativeEvents function.

231

Delete Focus

Client Side:

1. User Clicks on the Focus->DeleteFocus
2. Enters The foucsID and clicks DelFocus.
3. The details are sent to the userController delFocus function.
4. The details are packaged into a Message and sent to the sendUserEvents function.
5. This message is packed in to a XMLMessage using the parser and is sent to the Awareness Model Server

Server Side:

1. ServerThread (Part of the Awareness Communicator package) receives the Xml Message.
2. Sends the message to the AMController through the AMCommunicator's recieveMessage function
3. AMController uses the ServerParser to unpack the XML Message into MessageInfo object.
4. Use AMAccessLayer's UpdateEventLog method to log the incoming message.
5. delFocus method is called which in turn accesses the AMAccessLayer's deleteFromUserFocusT method and updateUserSrcTDelFocus to update the awareness model.
6. use AMAccessLayer's UpdateEventLog method to log the reply message.
7. A reply message is being sent basing on whether or not the focus is deleted by using the awareness communicator sendMessage function.

Client Side:

1. The User Listener Thread receives the message from the server and sends it to the User Controller.
2. The User Controller unpacks the message and depending on the message type performs necessary actions on the frmAwarenessClient (In this case deleteFocus, remove the focus from the focus list maintained by the user controller) and then display the collaborative events in the colloaborative events window using the displayCollaborativeEvents function.

Add Source

Client Side:

1. User Drags a source into the focus from outside the focus.
2. The details are sent to the userController addSource function with focusId and sourceId.
3. The details are packaged into a Message and sent to the sendUserEvents function.
4. This message is packed in to a XMLMessage using the parser and is sent to the Awareness Model Server

Server Side:

1. ServerThread (Part of the Awareness Communicator package) receives the Xml Message.
2. Sends the message to the AMController through the AMCommunicator's recieveMessage function
3. AMController uses the ServerParser to unpack the XML Message into MessageInfo object.
4. Use AMAccessLayer's UpdateEventLog method to log the incoming message.
5. addSource method is called which in turn accesses the AMAccessLayer's updateUserSrcT method to update the awareness model. It then retrieves the owner of the source by getSrcOwner method and sends him a notification(SOURCE_NOT) saying that his source is added to a focus by using the awareness communicator sendMessage function.
6. use AMAccessLayer's UpdateEventLog method to log the reply message.
7. A reply message is being sent basing on whether or not the source is added by using the awareness communicator sendMessage function.

Client Side:

1. The User Listener Thread receives the message from the server and sends it to the User Controller.
2. The User Controller unpacks the message and depending on the message type performs necessary actions on the frmAwarenessClient and then display the collaborative events in the colloaborative events window using the displayCollaborativeEvents function.

Del Source

Client Side:

1. User Drags a source out of the focus.
2. The details are sent to the userController addSource function with focusId and sourceId.
3. The details are packaged into a Message and sent to the sendUserEvents function.
4. This message is packed in to a XMLMessage using the parser and is sent to the Awareness Model Server

Server Side:

1. ServerThread (Part of the Awareness Communicator package) receives the Xml Message.
2. Sends the message to the AMController through the AMCommunicator's recieveMessage function
3. AMController uses the ServerParser to unpack the XML Message into MessageInfo object.
4. Use AMAccessLayer's UpdateEventLog method to log the incoming message.
5. delSource method is called which in turn accesses the AMAccessLayer's updateUserSrcT method to update the awareness model. It then retrieves the owner of the source by getSrcOwner method and sends him a notification(SOURCE_NOT) saying that his source is deleted from a focus. By using the awareness communicator sendMessage function.
6. use AMAccessLayer's UpdateEventLog method to log the reply message.
7. A reply message is being sent basing on whether or not the source is deleted using the awareness communicator sendMessage function.

Client Side:

1. The User Listener Thread receives the message from the server and sends it to the User Controller.
2. The User Controller unpacks the message and depending on the message type performs necessary actions on the frmAwarenessClient and then display the collaborative events in the colloaborative events window using the displayCollaborativeEvents function.

Switch On/Off Sources

Client Side:

1. User right clicks on the sources and chooses to switch it on/off
2. The details are packaged into a Message and sent to the sendUserEvents function.
3. This message is packed in to a XMLMessage using the parser and is sent to the Awareness Model Server

Server Side:

1. ServerThread (Part of the Awareness Communicator package) receives the Xml Message.
2. Sends the message to the AMController through the AMCommunicator's recieveMessage function
3. AMController uses the ServerParser to unpack the XML Message into MessageInfo object.
4. Use AMAccessLayer's UpdateEventLog method to log the incoming message.
5. switchSource method is called which in turn accesses the AMAccessLayer's updateUserSrcT method to update the awareness model. It then retrieves the owner of the source by getSrcOwner method and sends him a notification(SOURCE_NOT) saying that his source is deleted from a focus. By using the awareness communicator sendMessage function.
6. use AMAccessLayer's UpdateEventLog method to log the reply message.
7. A reply message is being sent using the awareness communicator sendMessage function.

Client Side:

1. The User Listener Thread receives the message from the server and sends it to the User Controller.
2. The User Controller unpacks the message and depending on the message type performs necessary actions on the frmAwarenessClient and then display the collaborative events in the colloaborative events window using the displayCollaborativeEvents function.

Awareness Model Message

Client Side:

1. User Enters the AMM message and clicks submit
2. The details are packaged into a Message and sent to the sendUserEvents function.
3. This message is packed in to a XMLMessage using the parser and is sent to the Awareness Model Server

Server Side:

1. ServerThread (Part of the Awareness Communicator package) receives the Xml Message.
2. Sends the message to the AMController through the AMCommunicator's recieveMessage function
3. AMController uses the ServerParser to unpack the XML Message into MessageInfo object.
4. Use AMAccessLayer's UpdateEventLog method to log the incoming message.
5. sendAMM method is called which in turn accesses the AMAccessLayer's validateAMM method to determine the validity of the AMM message. It then retrieves the interested users (by using the AMAccessLayer's getInterestedUser method) if the AMM message is valid and sends sends them the message by using the awareness communicator sendMessage function. If the message is not valid then just the reply message is sent.
6. use AMAccessLayer's UpdateEventLog method to log the reply message.
7. A reply message is being sent using the awareness communicator sendMessage function.

Client Side:

1. The User Listener Thread receives the message from the server and sends it to the User Controller.
2. The User Controller unpacks the message and depending on the message type performs necessary actions on the frmAwarenessClient and then display the collaborative events in the colloaborative events window using the displayCollaborativeEvents function.

Personal Message

Client Side:

1. User Enters the Personal message and clicks submit
2. The details are packaged into a Message and sent to the sendUserEvents function.
3. This message is packed in to a XMLMessage using the parser and is sent to the Awareness Model Server

Server Side:

1. ServerThread (Part of the Awareness Communicator package) receives the Xml Message.
2. Sends the message to the AMController through the AMCommunicator's recieveMessage function
3. AMController uses the ServerParser to unpack the XML Message into MessageInfo object.
4. Use AMAccessLayer's UpdateEventLog method to log the incoming message.
5. sendPerMessage method is called which in turn accesses the AMAccessLayer's validateUserSrcCombination method to determine the validity of the sender and receiver user-src pairs. It then sends the message by using the awareness communicator sendMessage function to the receiver user/src pair. If the message is not valid then just the reply message is sent.
6. use AMAccessLayer's UpdateEventLog method to log the reply message.
7. A reply message is being sent using the awareness communicator sendMessage function.

Client Side:

1. The User Listener Thread receives the message from the server and sends it to the User Controller.
2. The User Controller unpacks the message and depending on the message type performs necessary actions on the frmAwarenessClient and then display the collaborative events in the colloaborative events window using the displayCollaborativeEvents function.

Second Order

Client Side:

1. User right clicks on a user in the Src Super Set and clicks on SecondOrder.
2. Second Order can be done on users only. So local validation is done by using the isUsers function and if the selected node is not a user then a error message is displayed. If the selected node is a user then..
3. The details are packaged into a Message by user Controller's getsecondOrder function and sent to the sendUserEvents function.
4. This message is packed in to a XMLMessage using the parser and is sent to the Awareness Model Server

Server Side:

1. ServerThread (Part of the Awareness Communicator package) receives the Xml Message.
2. Sends the message to the AMController through the AMCommunicator's receive message.
3. AMController uses the ServerParser to unpack the XML Message into MessageInfo object.
4. Use AMAccessLayer's UpdateEventLog method to log the incoming message.
5. secondOrder method is called which in turn accesses the AMAccessLayer's getSecondOrderInfo method to retrieve the second order information of the user. The reply message contains the second order information only if the user has privilege to get the second order information. The second order information is nothing but the awareness map.
6. use AMAccessLayer's UpdateEventLog method to log the reply message.
7. A reply message is being sent using the awareness communicator sendMessage function.

Client Side:

1. The User Listener Thread receives the message from the server and sends it to the User Controller.
2. The User Controller unpacks the message and depending on the message type performs necessary actions in this case-- -displays the awareness map in a different form by calling the showFrm function in a different thread and then calls the updateAwarenessMap function in the newly generated form to display the awareness Map. It also display the collaborative events in the colloaborative events window using the displayCollaborativeEvents function.

Reverse Look up

Client Side:

1. User right clicks on a user in the Src Super Set and clicks on ReverseLookup.
2. Reverse Lookup can be done on users own sources only. So local validation is done by using the isMySource function and if the selected node is not a user then a error message is displayed. If the selected node is a source then..
3. The details are packaged into a Message by user Controller's getReverseLookup function and sent to the sendUserEvents function.
4. This message is packed into a XMLMessage using the parser and is sent to the Awareness Model Server

Server Side:

1. ServerThread (Part of the Awareness Communicator package) receives the Xml Message.
2. Sends the message to the AMController through the AMCommunicator's receive message.
3. AMController uses the ServerParser to unpack the XML Message into MessageInfo object.
4. Use AMAccessLayer's UpdateEventLog method to log the incoming message.
5. reverseLookup method is called which in turn accesses the AMAccessLayer's getReverseLookup method to retrieve the list of users who are interested in the user's source. The reply message contains the reverse lookup info that is the user id and the focus in which the source is present.
6. use AMAccessLayer's UpdateEventLog method to log the reply message.
7. A reply message is being sent using the awareness communicator sendMessage function.

Client Side:

1. The User Listener Thread receives the message from the server and sends it to the User Controller.
2. The User Controller unpacks the message and depending on the message type performs necessary actions in this case-- -displays the awareness map in a different form by calling the showFrmResult function in a different thread(this function displays a new form frmResult) and then calls the updateReverseLookup function in the newly generated form to display the ReverseLookup information. It also display the collaborative events in the colloaborative events window using the displayCollaborativeEvents function.

WISIWYS

Client Side:

1. User right clicks on a user in the Src Super Set and clicks on Wisiwys.
2. wisiwys can be done on sources only. So local validation is done by using the isSource function and if the selected node is not a user then a error message is displayed. If the selected node is a source then..
3. The details are packaged into a Message by user Controller's getWisiwys function and sent to the sendUserEvents function.
4. This message is packed into a XMLMessage using the parser and is sent to the Awareness Model Server

Server Side:

1. ServerThread (Part of the Awareness Communicator package) receives the Xml Message.
2. Sends the message to the AMController through the AMCommunicator's receive message.
3. AMController uses the ServerParser to unpack the XML Message into MessageInfo object.
4. Use AMAccessLayer's UpdateEventLog method to log the incoming message.
5. wisiwys method is called which in turn accesses the AMAccessLayer's getWisiwys method to retrieve the list of users who are all interested in this source. The reply message contains the wisiwys info that is the user id and the focus in which the source is present.
6. use AMAccessLayer's UpdateEventLog method to log the reply message.
7. A reply message is being sent by using the awareness communicator sendMessage function.

Client Side:

1. The User Listener Thread receives the message from the server and sends it to the User Controller.
2. The User Controller unpacks the message and depending on the message type performs necessary actions in this case-- -displays the awareness map in a different form by calling the showFrmResult function in a different thread(this function displays a new form frmResult) and then calls the updateWisiwys function in the newly generated form to display thewisiwys information. It also display the collaborative events in the colloaborative events window using the displayCollaborativeEvents function.

View Source Info

Client Side:

1. User right clicks on a user in the Src Super Set and clicks on View Source Info.
2. View Source Info can be done on sources only. So local validation is done by using the isSource function and if the selected node is not a user then a error message is displayed. If the selected node is a source then..
3. The details are packaged into a Message and sent to the sendUserEvents function.
4. This message is packed into a XMLMessage using the parser and is sent to the Awareness Model Server

Server Side:

1. ServerThread (Part of the Awareness Communicator package) receives the Xml Message.
2. Sends the message to the AMController through the AMCommunicator's receive message.
3. AMController uses the ServerParser to unpack the XML Message into MessageInfo object.
4. Use AMAccessLayer's UpdateEventLog method to log the incoming message.
5. wisiwys method is called which in turn accesses the AMAccessLayer's viewSourceInfo method to retrieve the list of users who are all interested in this source. The reply message contains the source info.
6. use AMAccessLayer's UpdateEventLog method to log the reply message.
7. A reply message is being sent by using the awareness communicator sendMessage function.

Client Side:

1. The User Listener Thread receives the message from the server and sends it to the User Controller.
2. The User Controller unpacks the message and depending on the message type performs necessary actions in this case-- -displays the awareness map in a different form by calling the showFrmResult function in a different thread(this function displays a new form frmResult) and then calls the updateWisiwys function in the newly generated form to display the ReverseLookup information. It also display the collaborative events in the colloaborative events window using the displayCollaborativeEvents function.

MisMatch(MM)/ MisMatchAvoided(MMA)

Client Side:

1. User Enters the MM message and clicks submit
2. The details are packaged into a Message and sent to the sendUserEvents function.
3. This message is packed in to a XMLMessage using the parser and is sent to the Awareness Model Server

Server Side:

1. ServerThread (Part of the Awareness Communicator package) receives the Xml Message.
2. Sends the message to the AMController through the AMCommunicator's recieveMessage function
3. AMController uses the ServerParser to unpack the XML Message into MessageInfo object.
4. Use AMAccessLayer's UpdateEventLog method to log the incoming message.

Client side:

1. The User Listener Thread receives the message from the server and sends it to the User Controller.
2. The User Controller unpacks the message and depending on the message type performs necessary actions in this case---displays the collaborative events in the colloaborative events window using the displayCollaborativeEvents function.
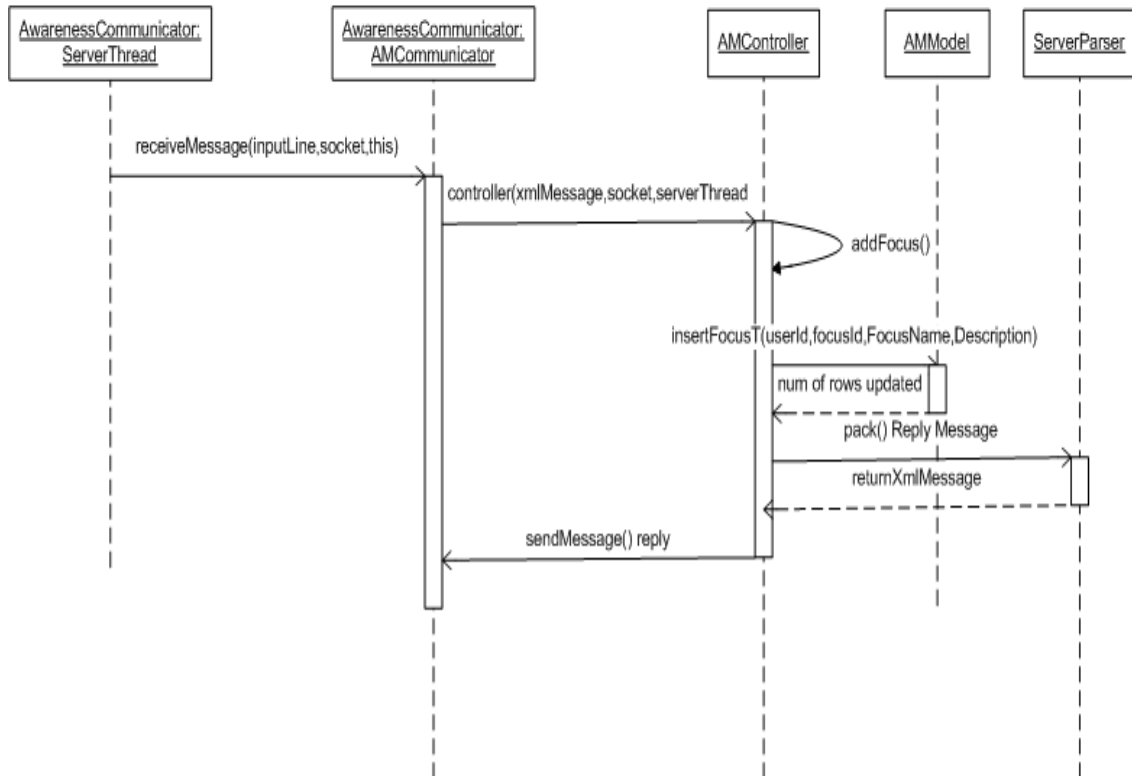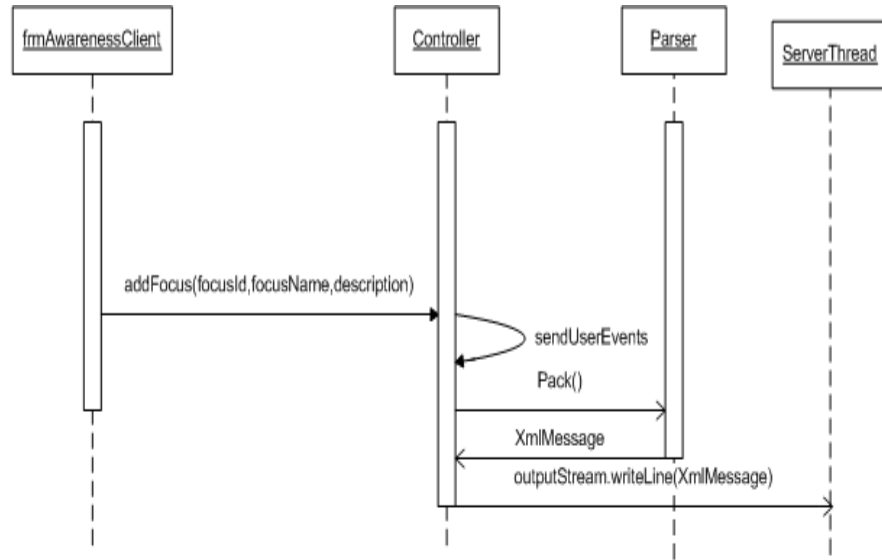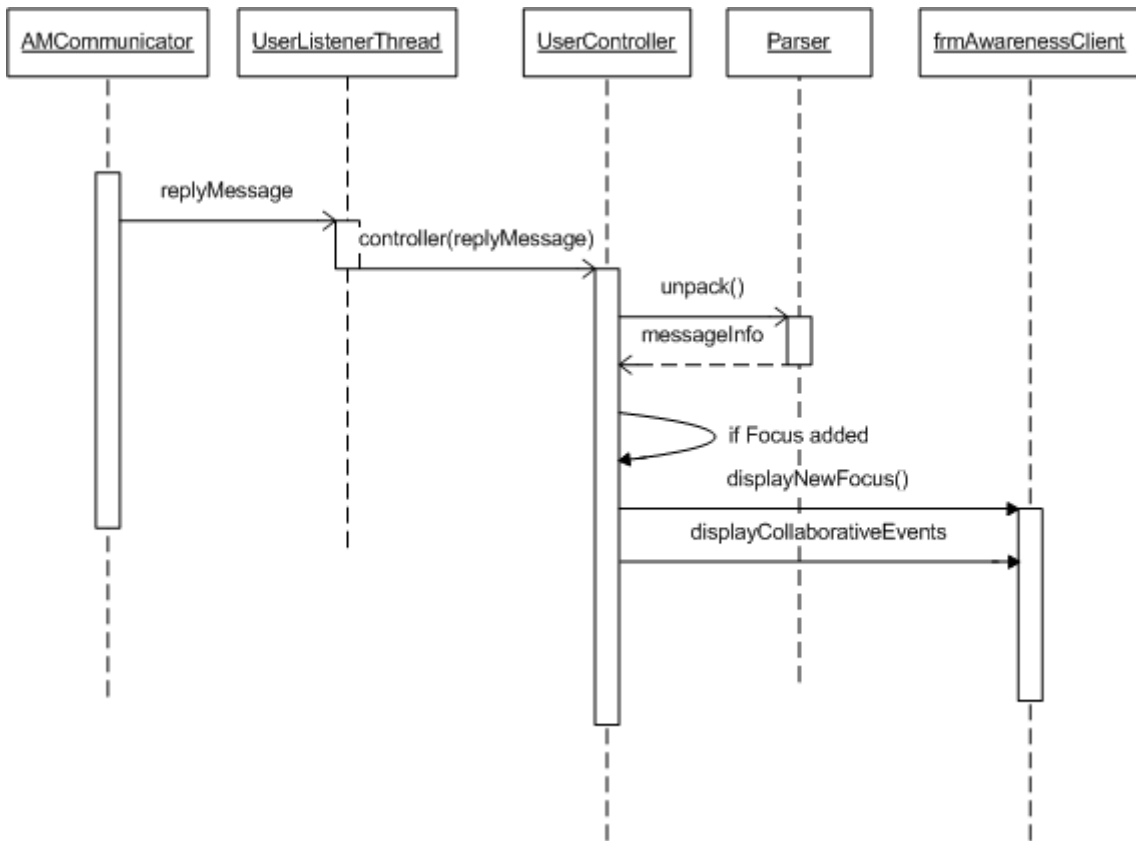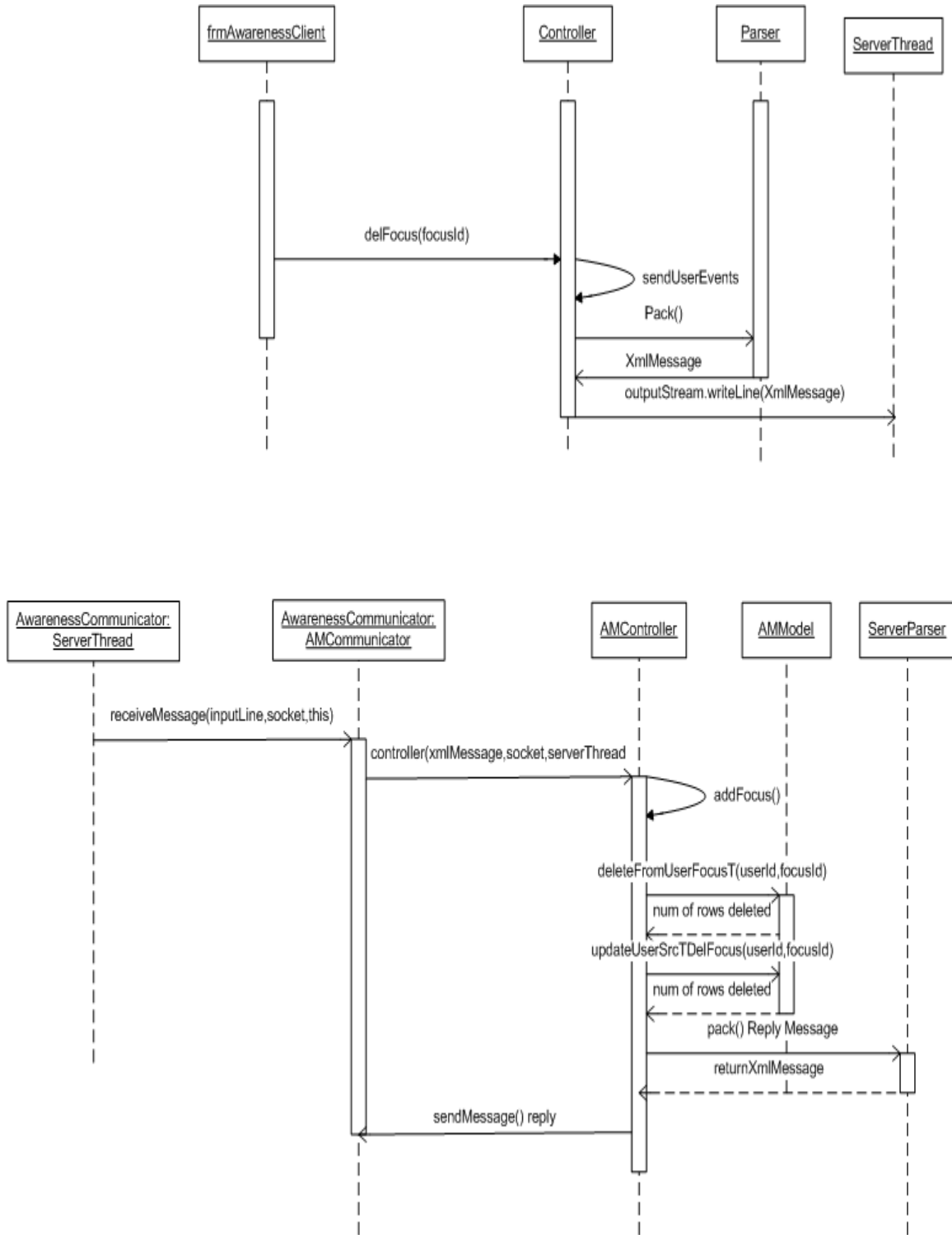
Sequence Diagrams

Authentication
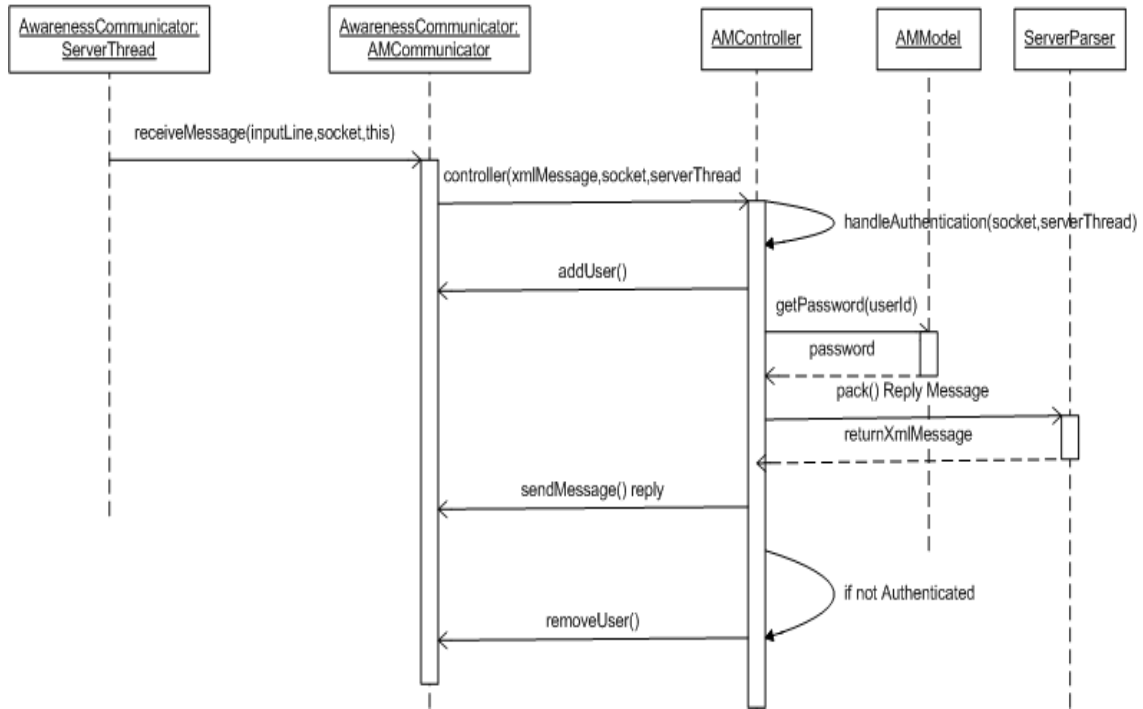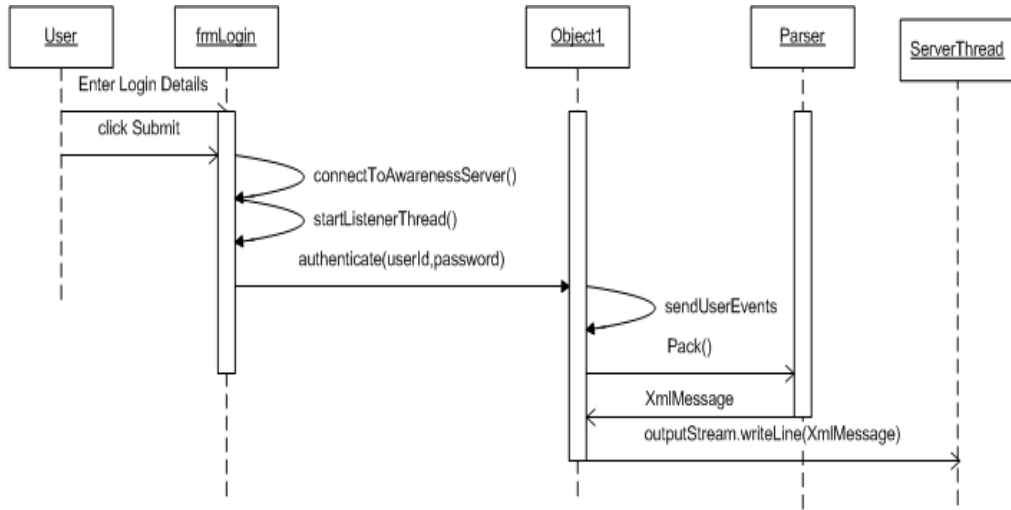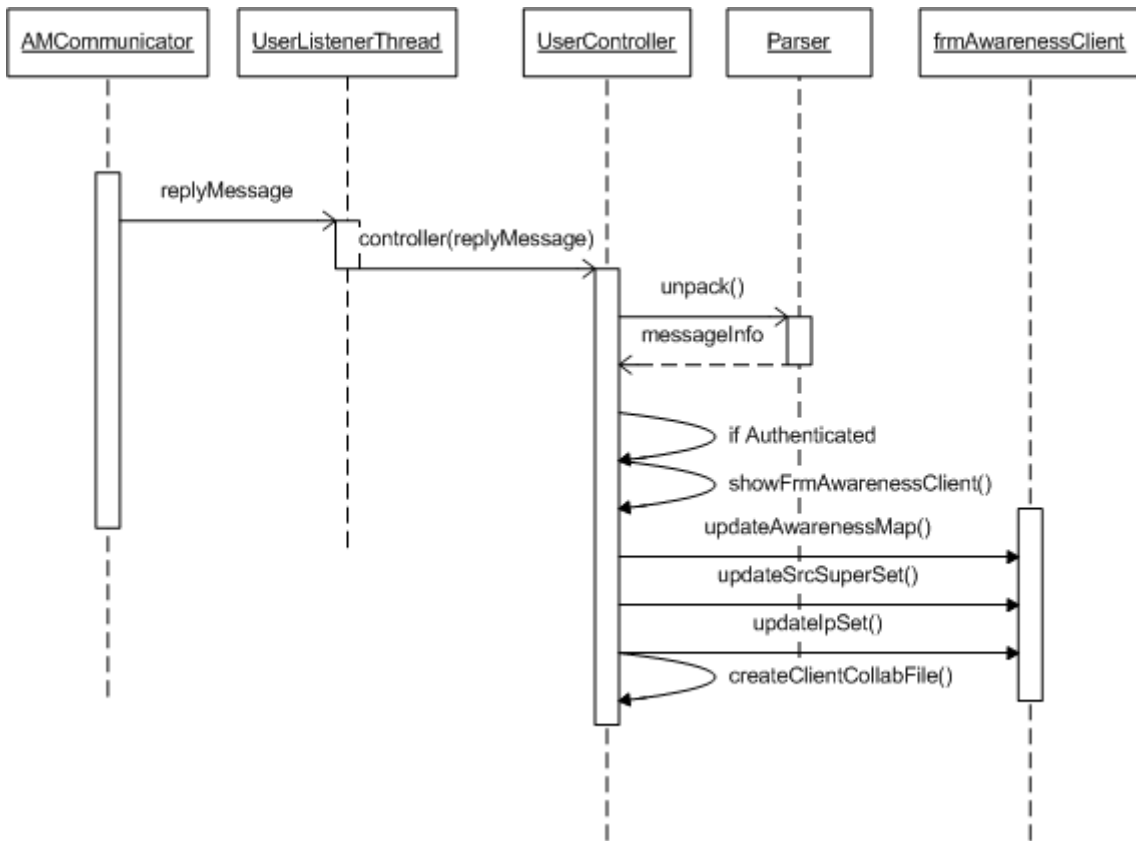
Add Focus:
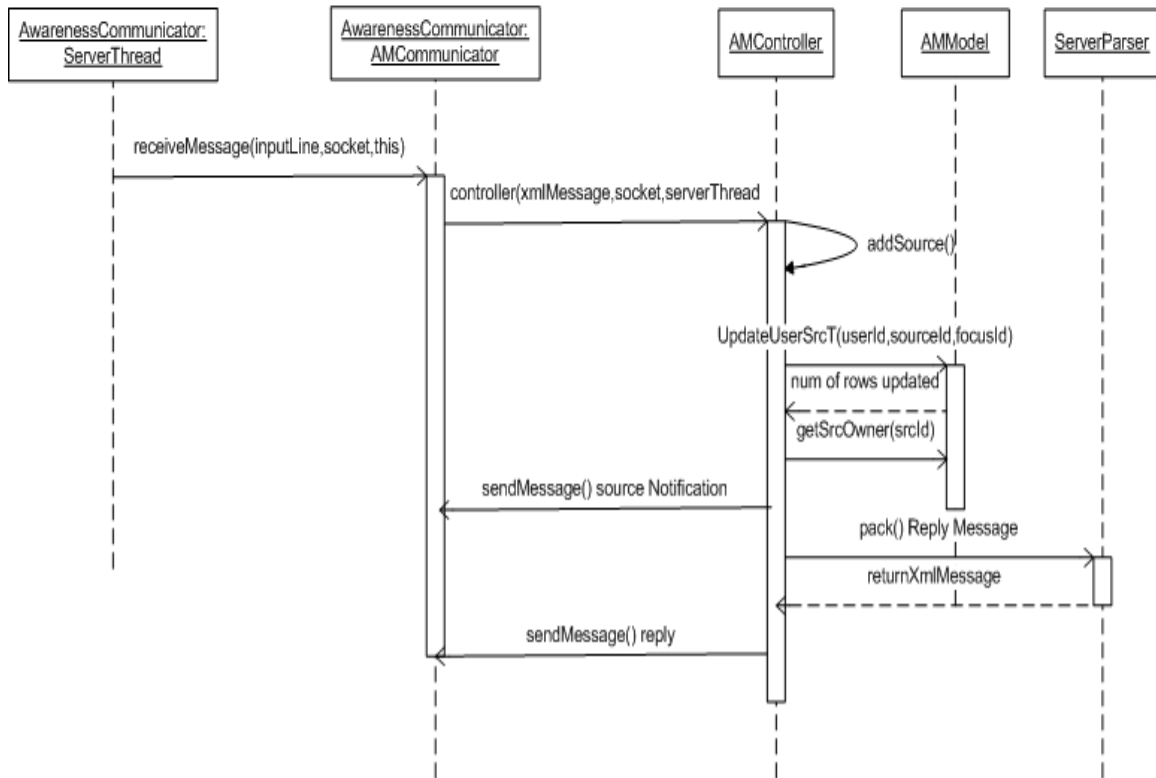
Delete Focus
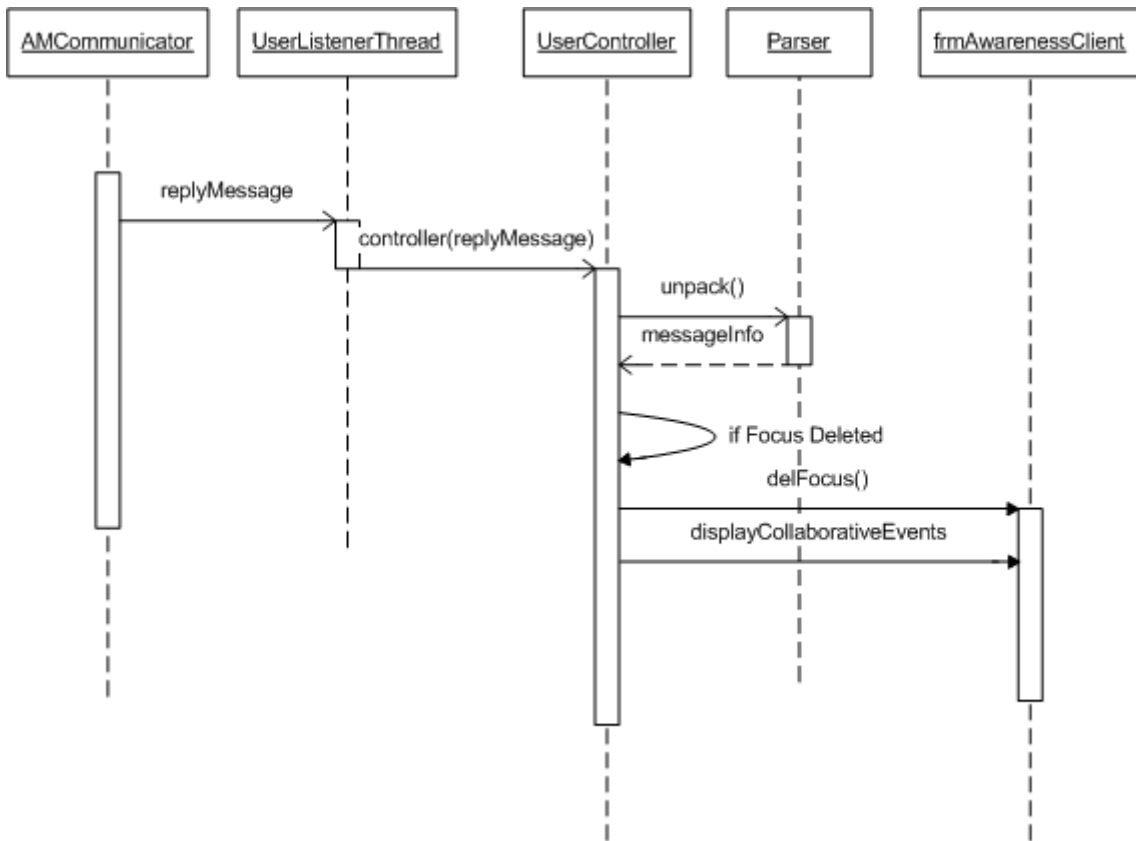
Add Source
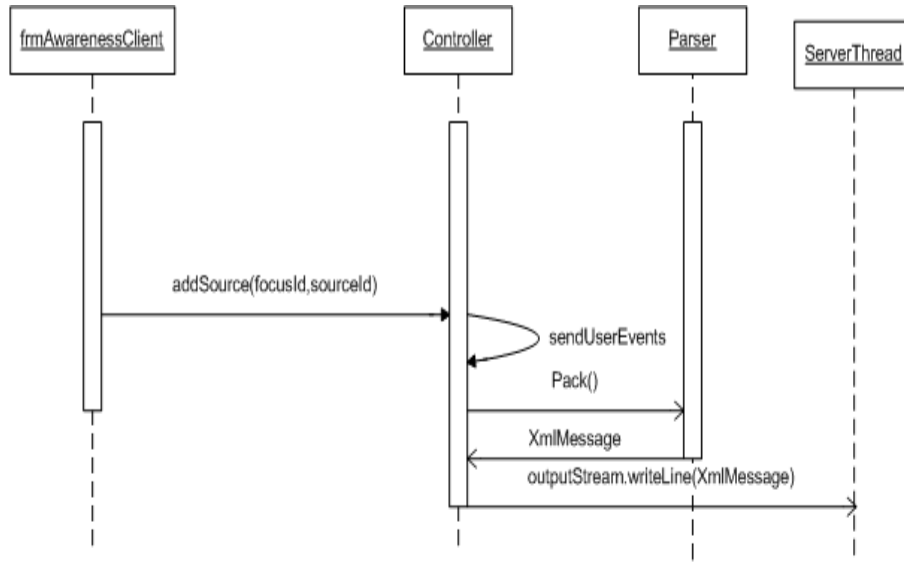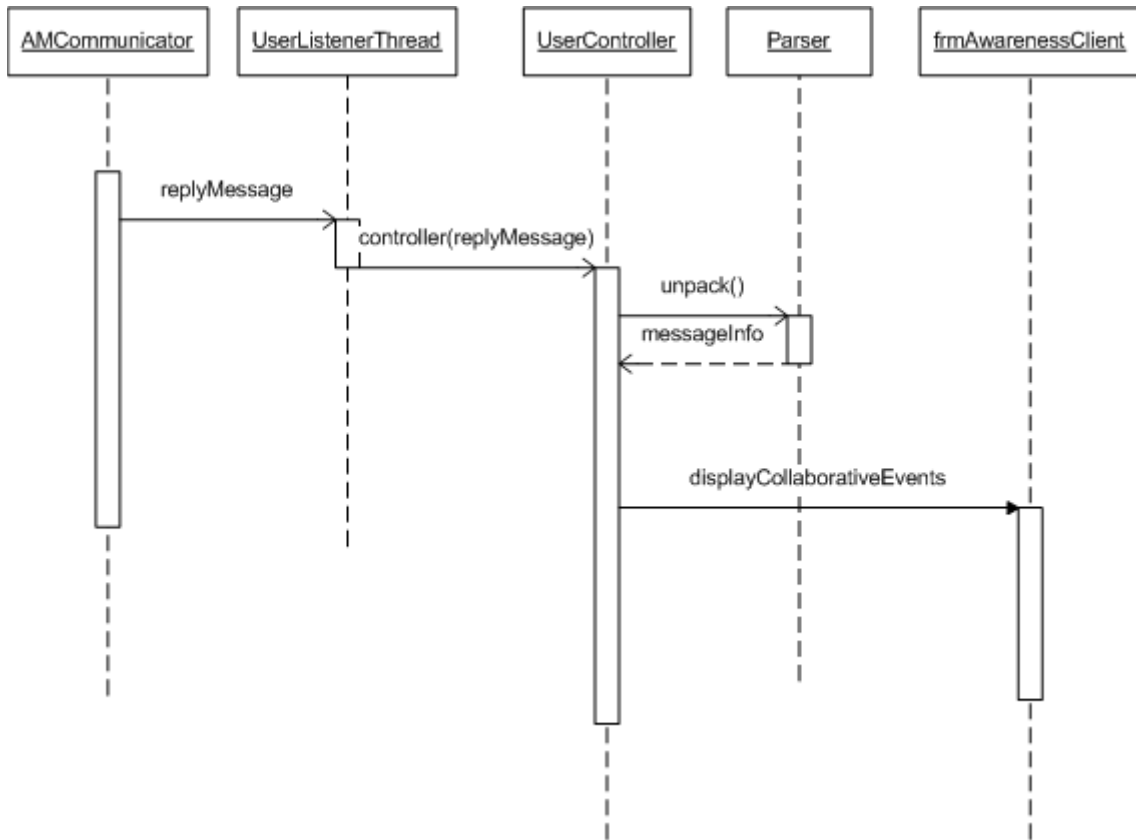
Delete Source:
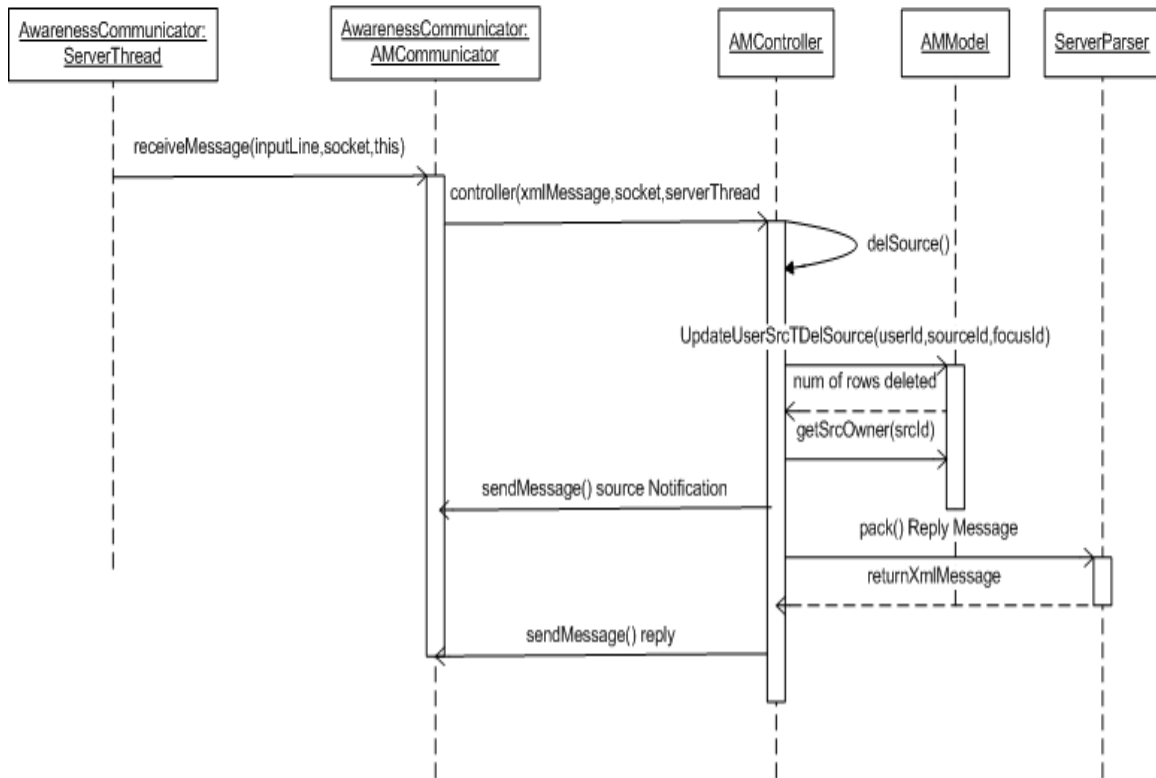
AMM Message
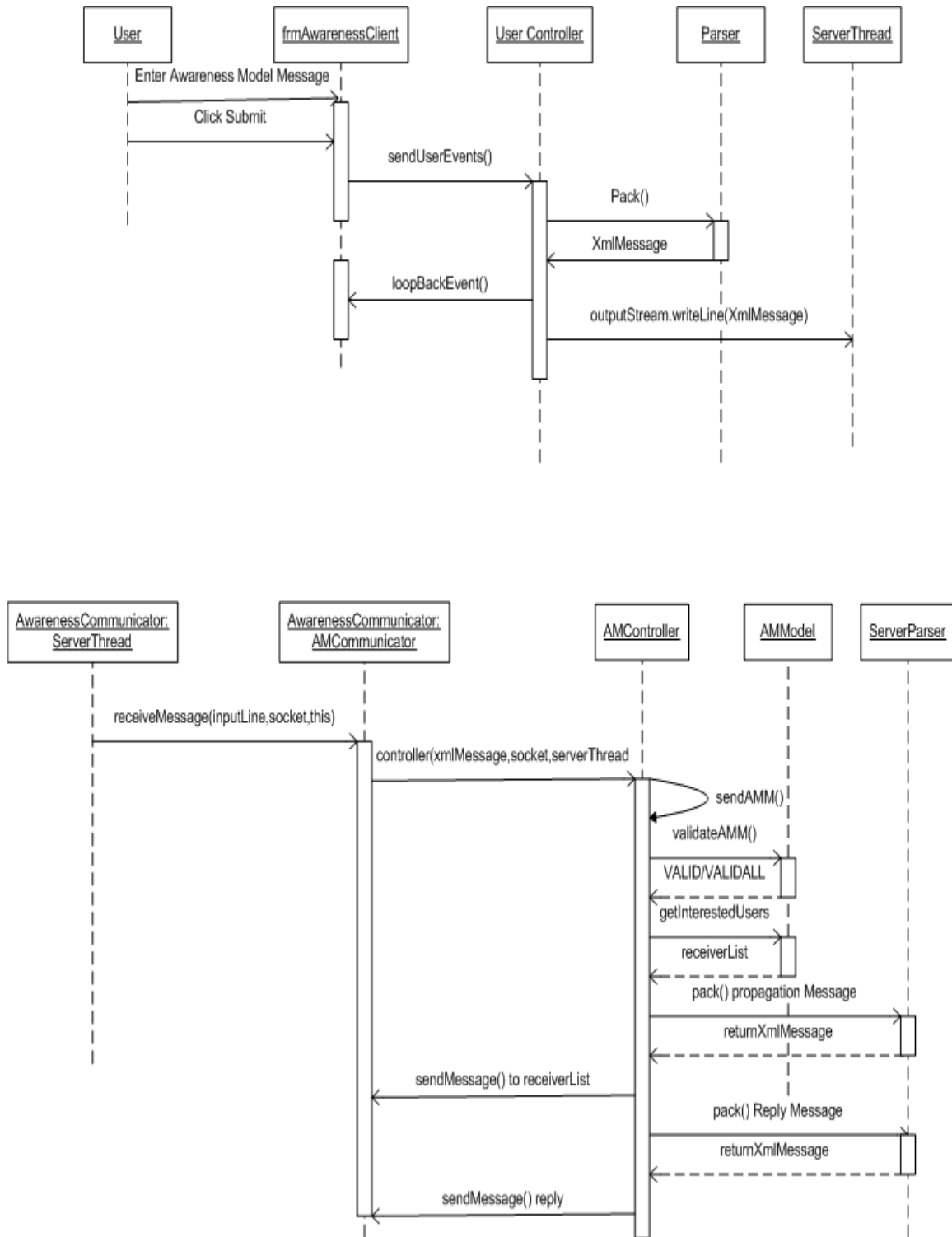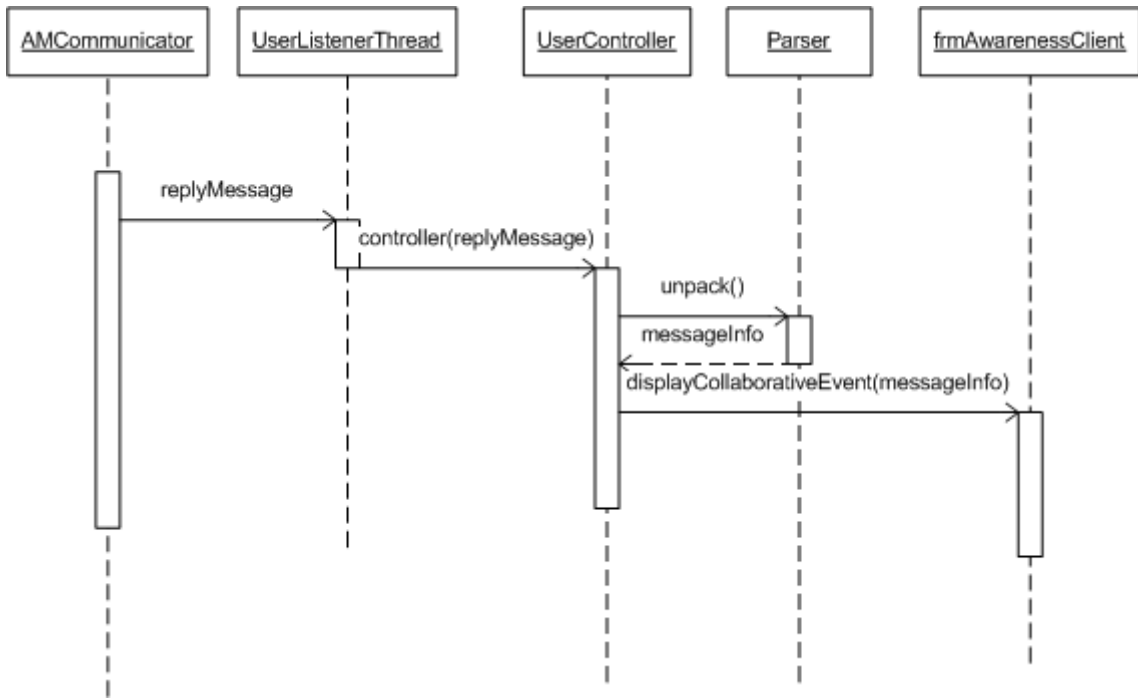
Personal Message
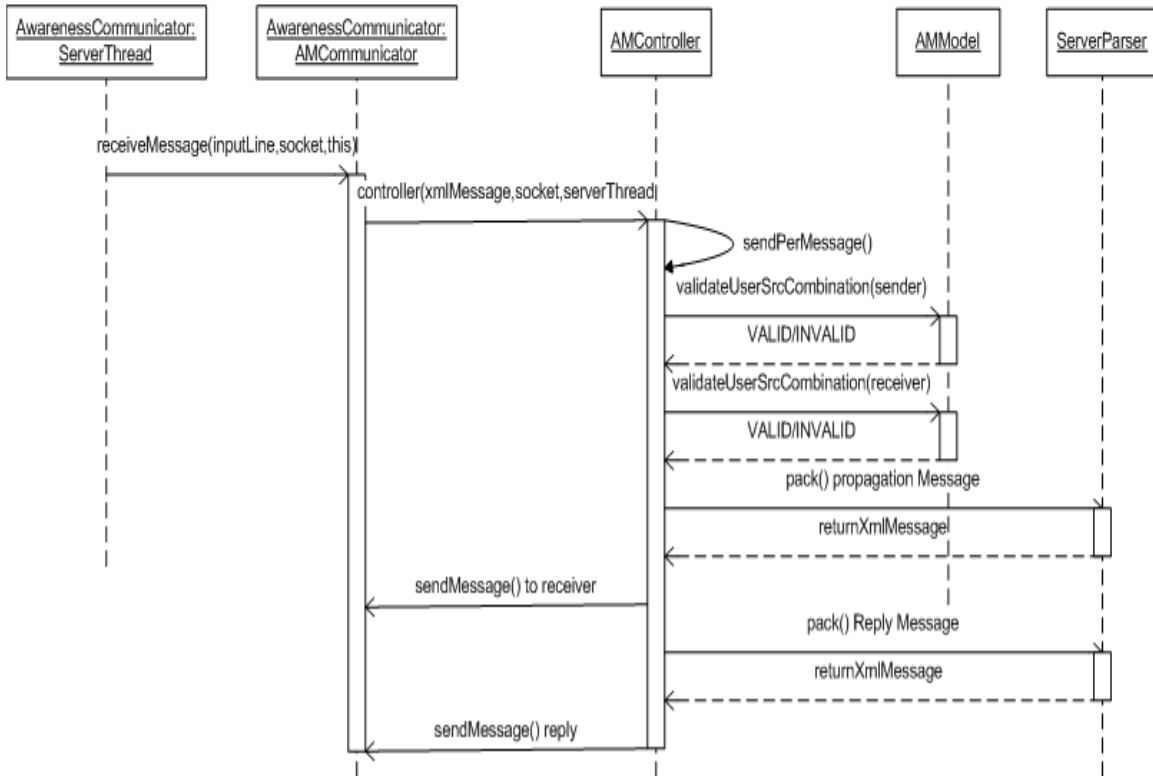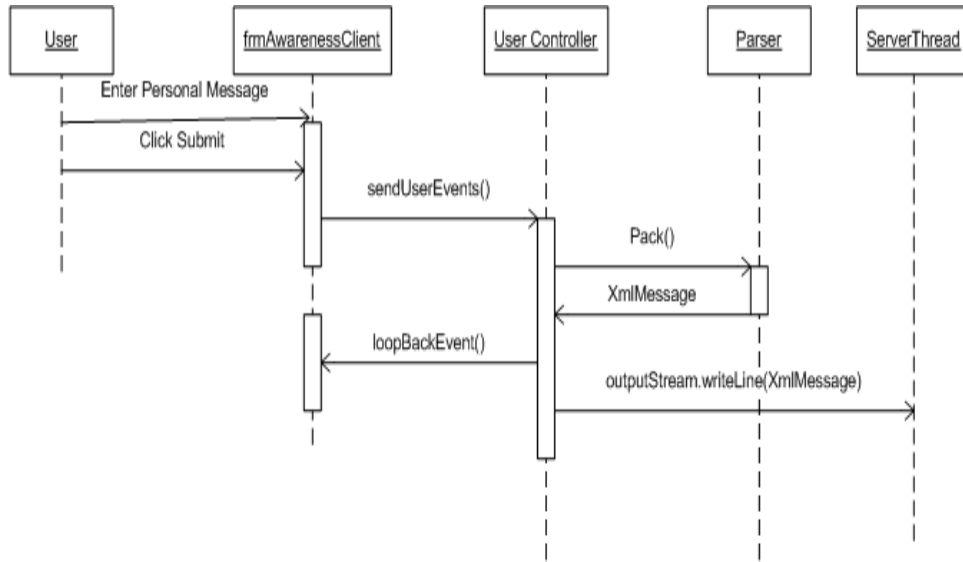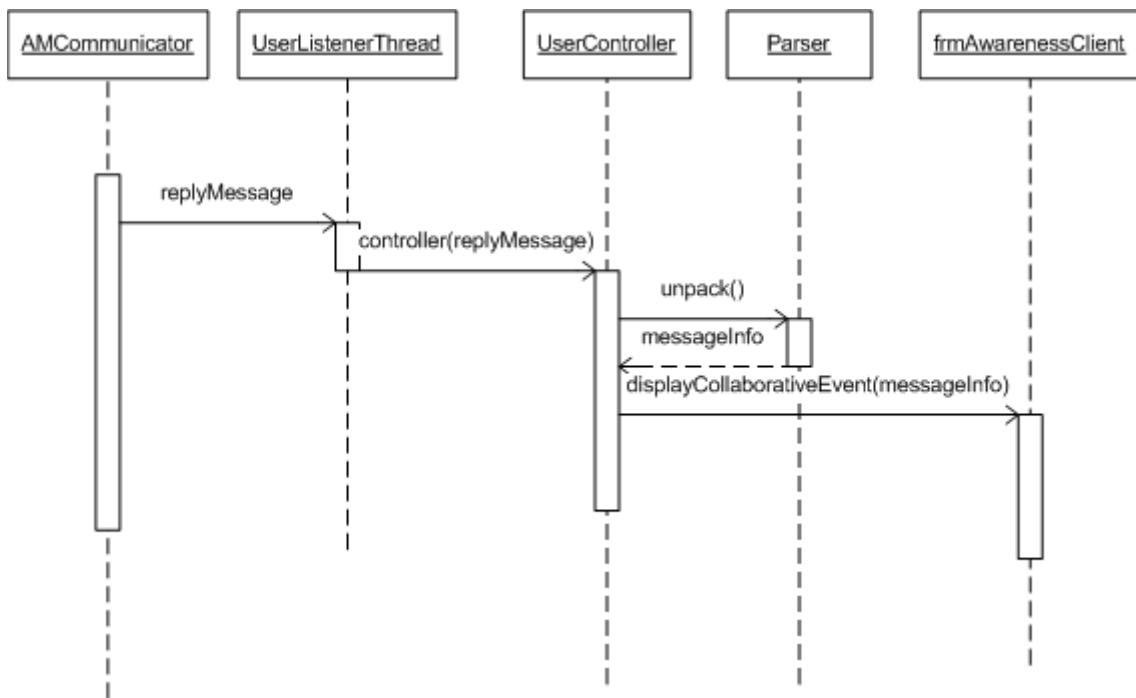
Second Order Info
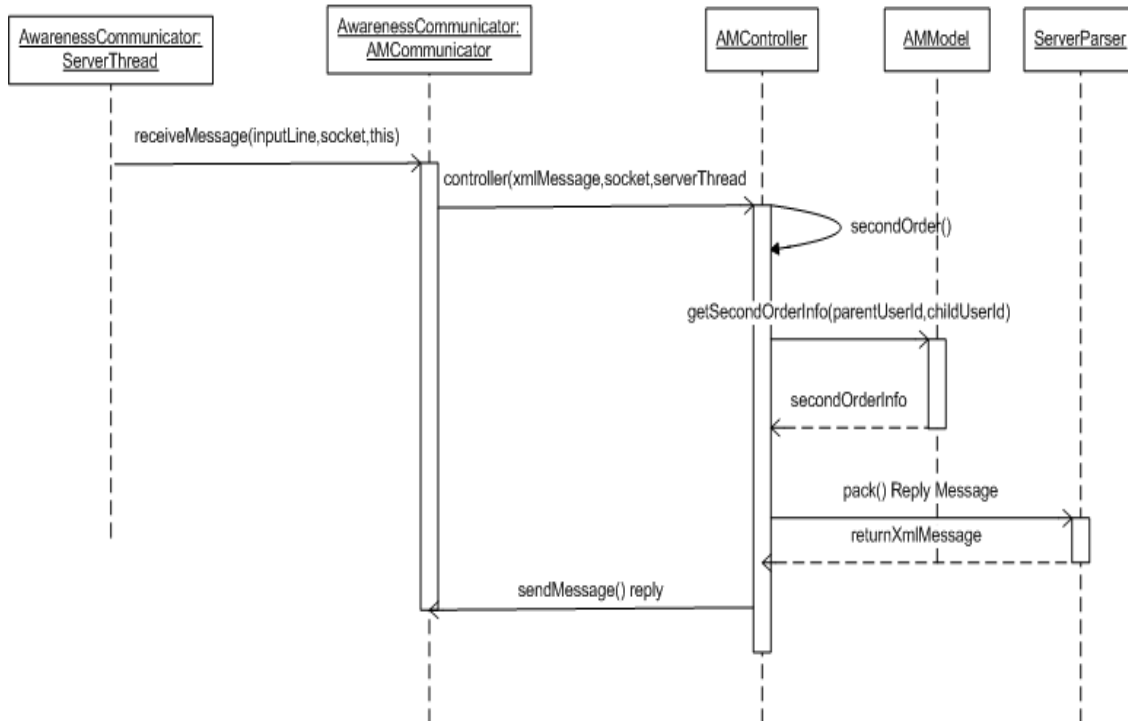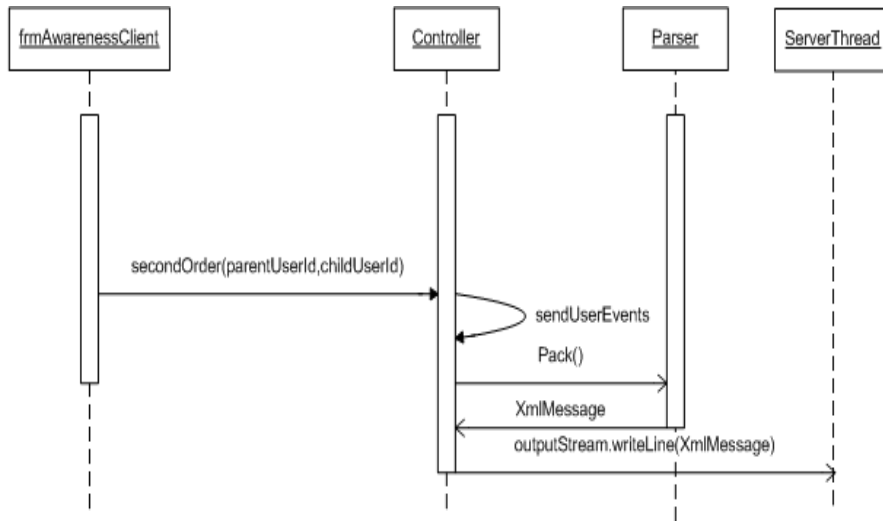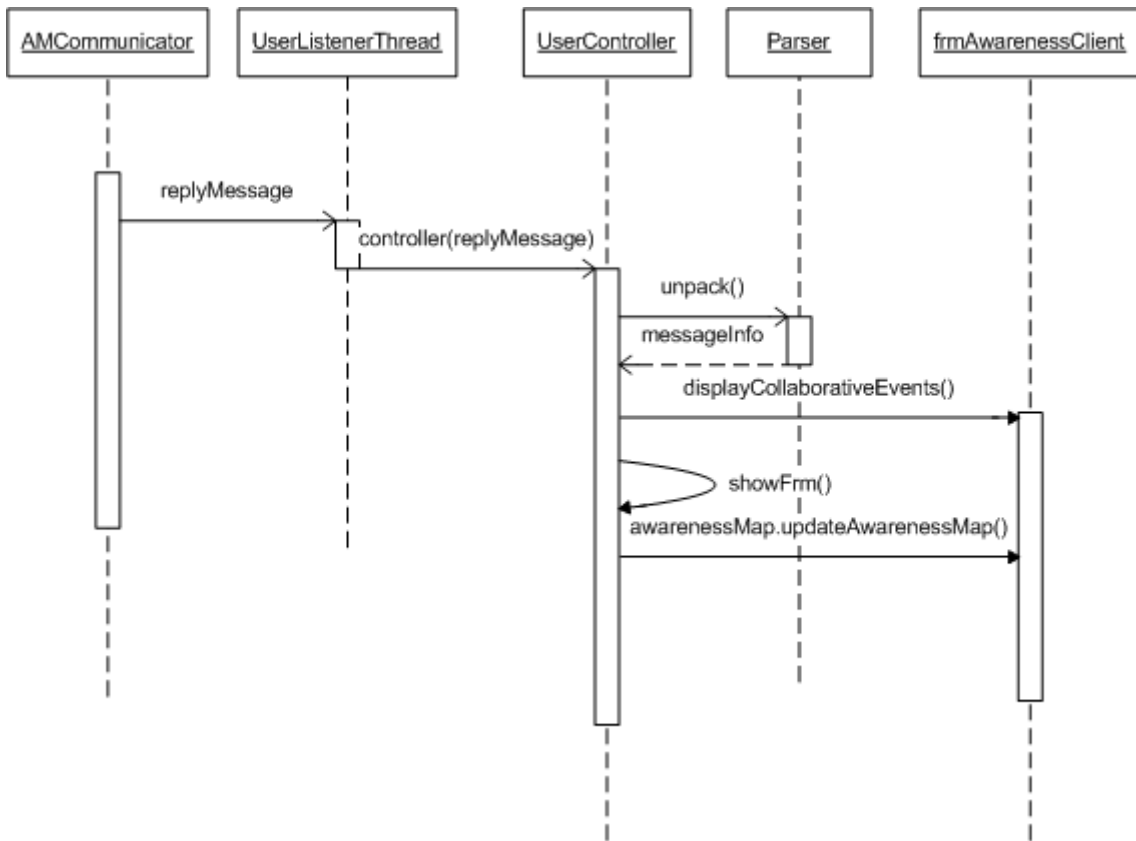
Reverse Lookup
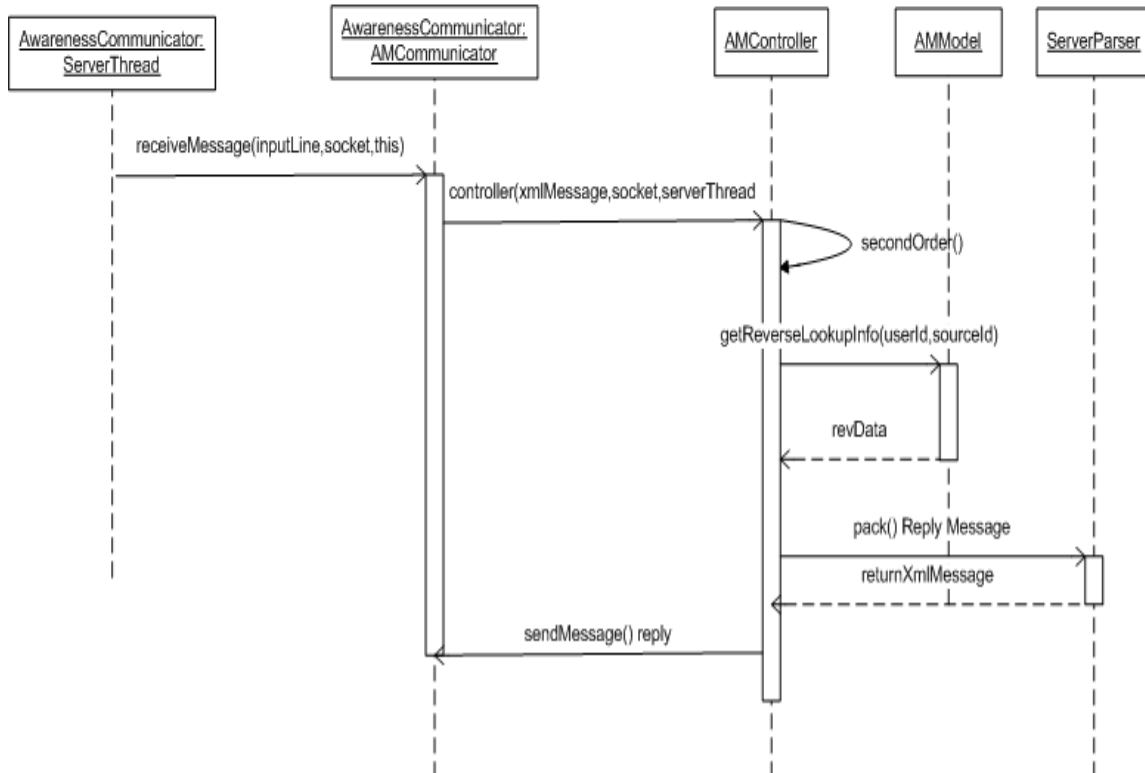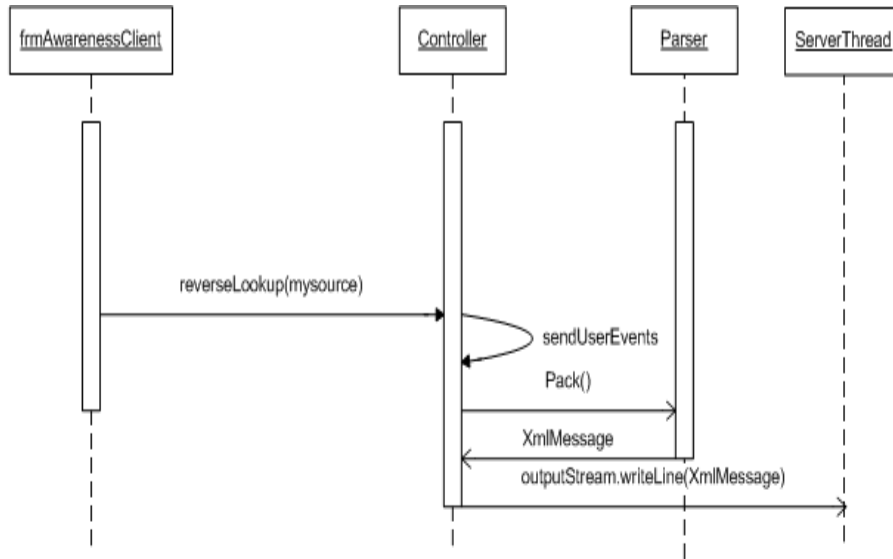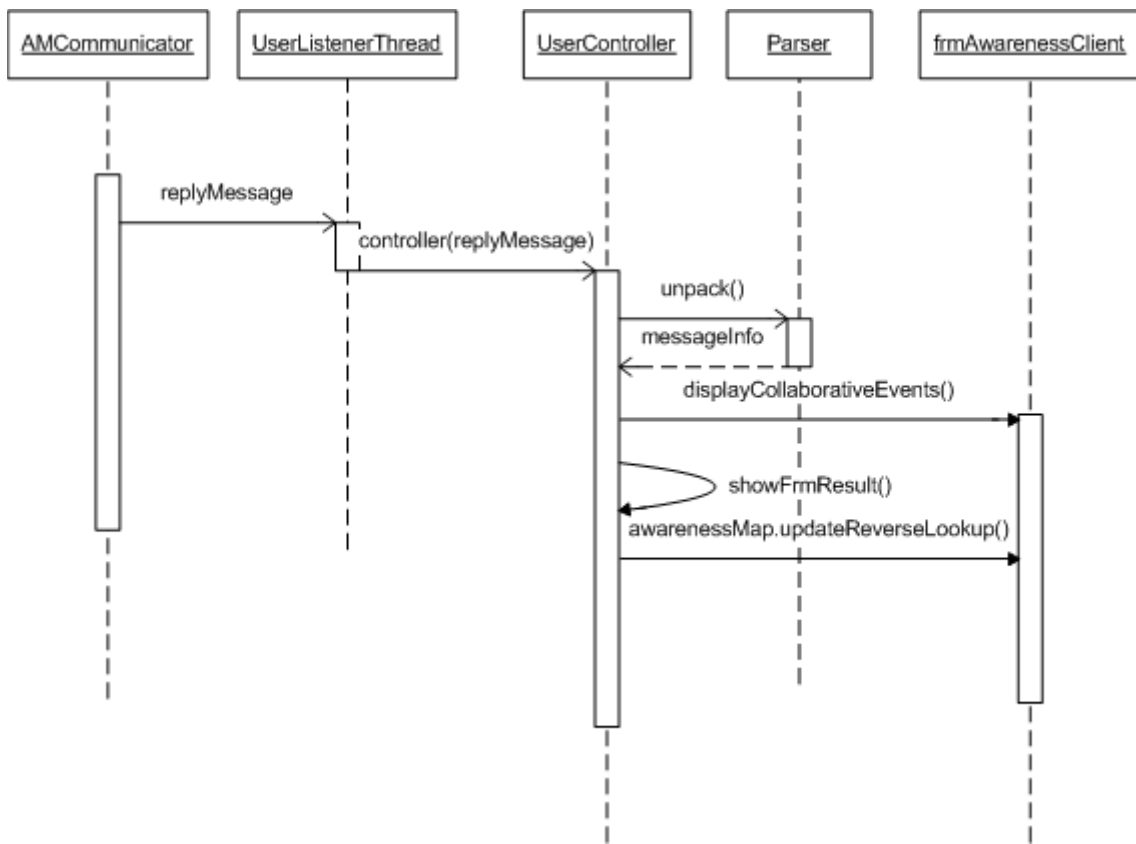
WISIWYS
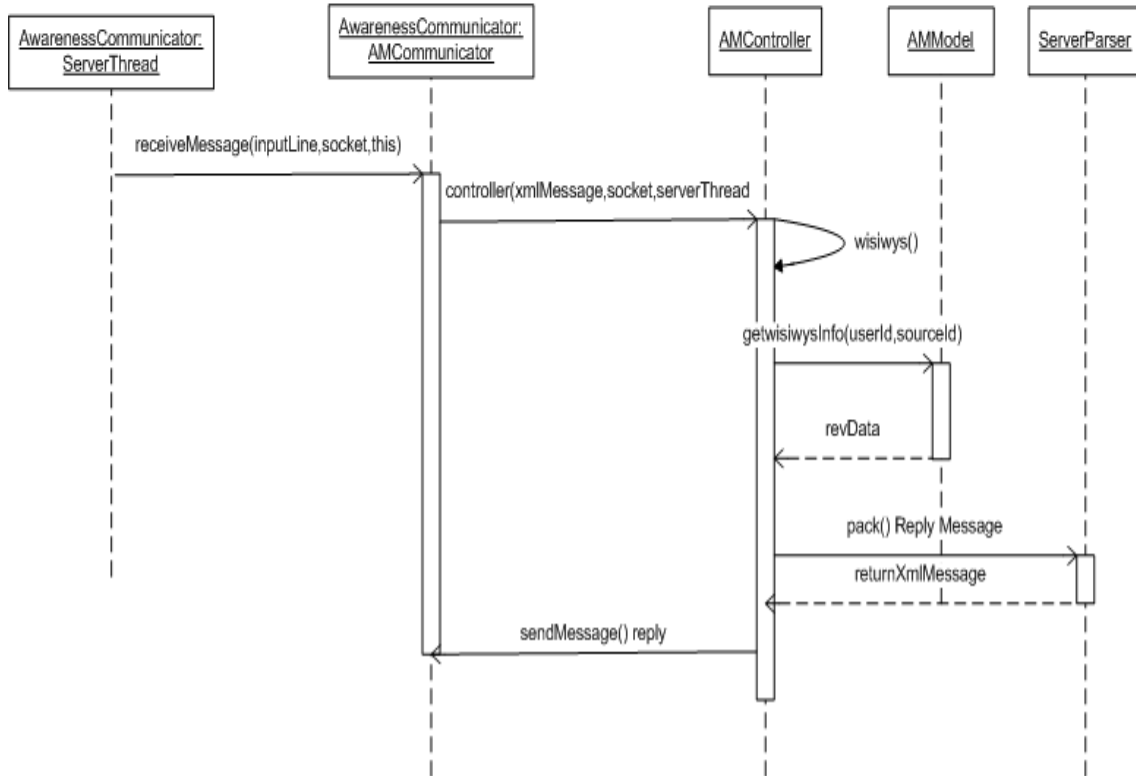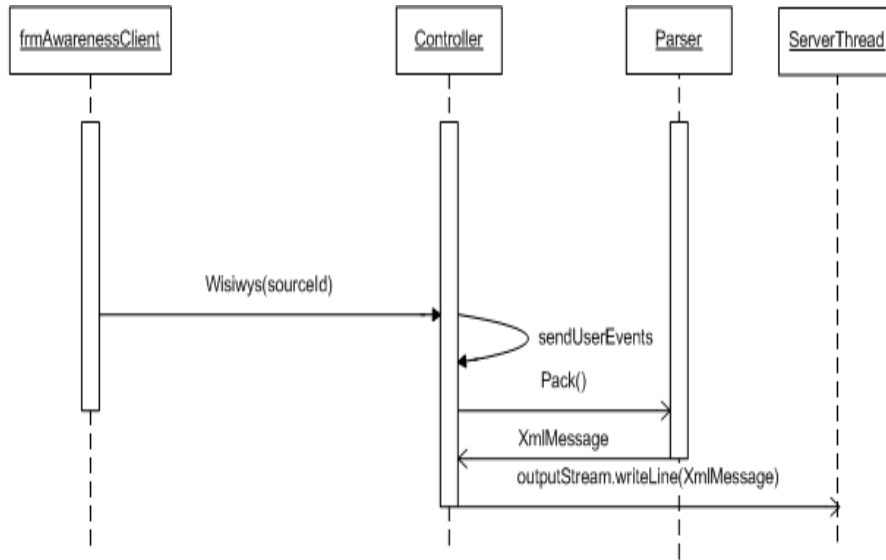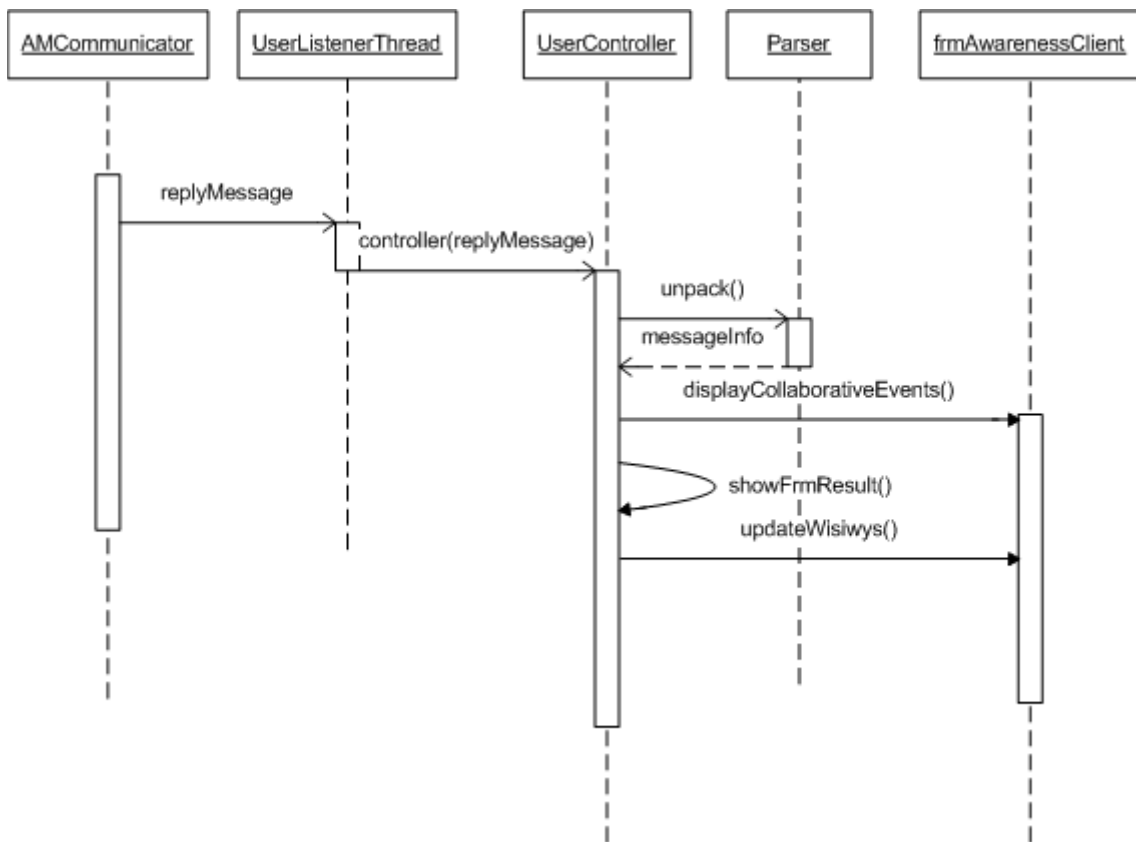
Design Issues

In this section we go through the coding part and explain each part in detail.

Initialization of the Awareness Server:-
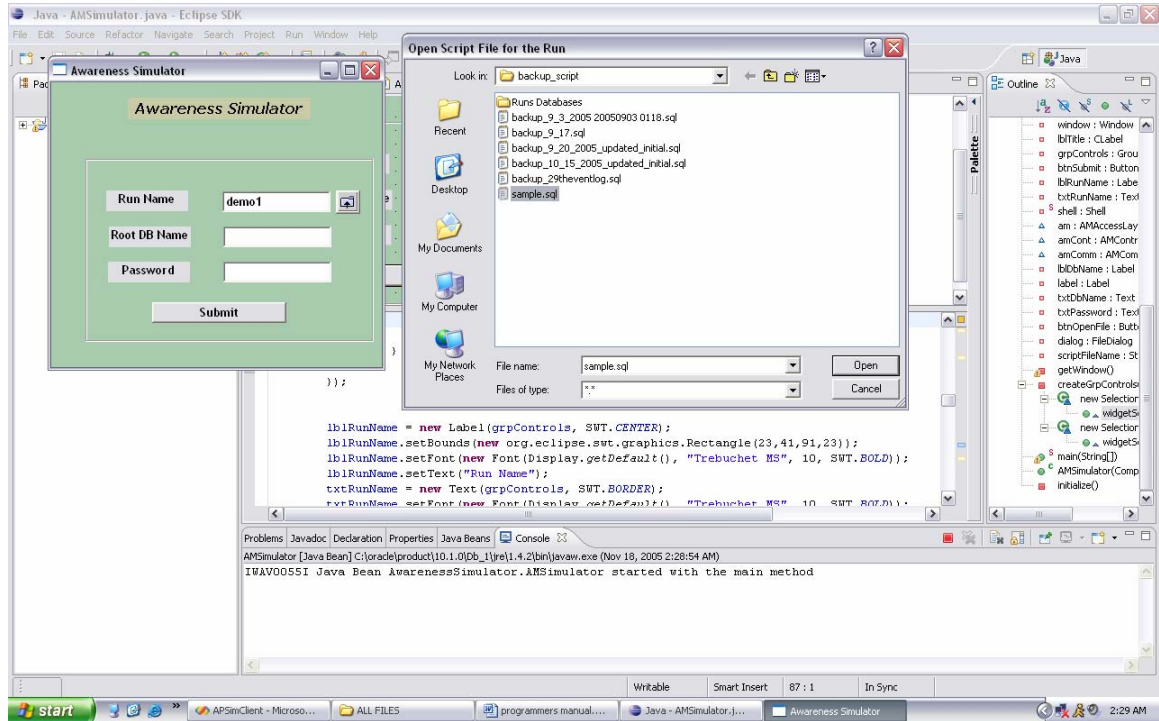
When the user runs the Awareness Server he is presented with a window asking for RunName, Root DB Name and the password. The Run name is the name you want to give to the current simulation run. The Root DB Name and the password are the details you give to logon to your Mysql adminis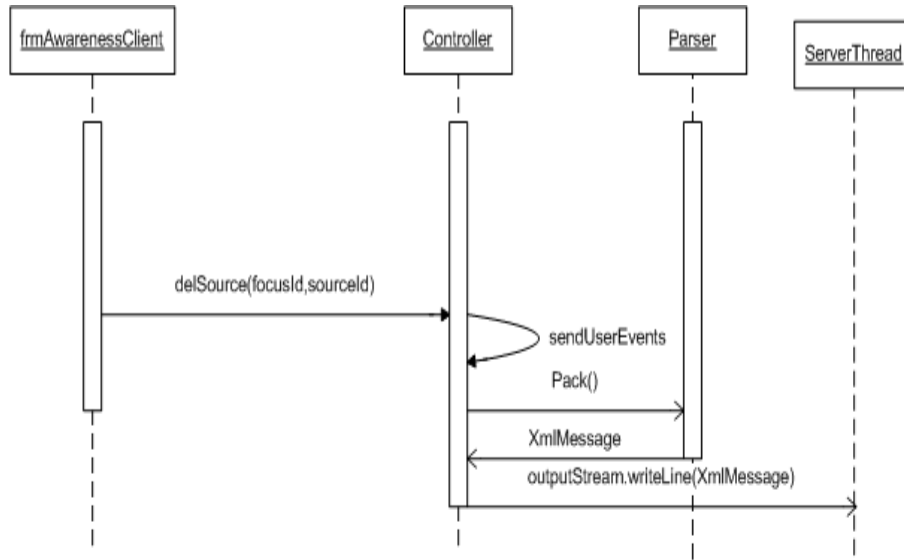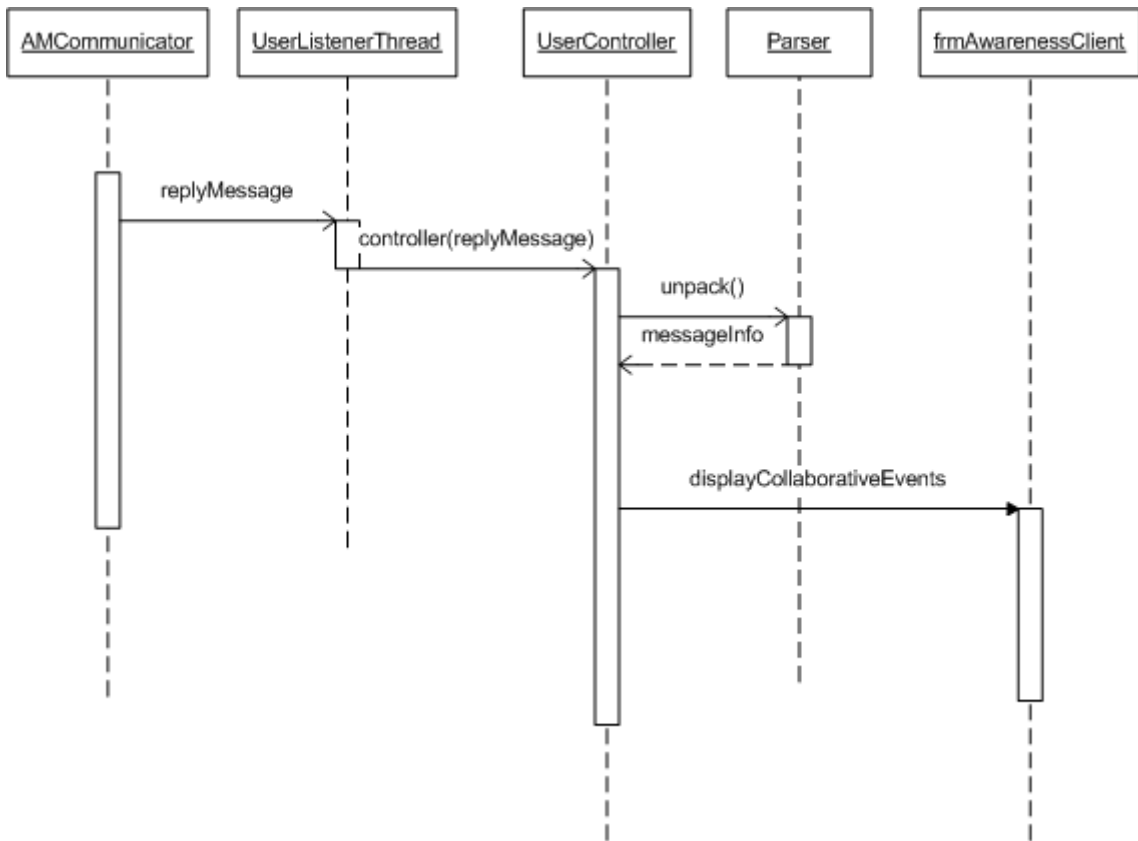trator. The user needs to click on the button on the right side of the Run Name text box and select the script containing the statements to create the new database with the initial data. After the User Enters all these details he clicks on Submit button.

The following code is called which is in the btnSubmit's widgetSelected function.

```
if(!txtRunName.getText().equals(""))
{
String runName =      txtRunName.getText();
String dbName = txtDbName.getText();
String password = txtPassword.getText();
am = new  AMAccessLayer(runName,dbName,password,scriptFileName);
amCont= new AMController();
```

263

amComm= new AMCommunicator();
amCont.setReference(am,amComm);
amComm.setReference(amCont);
amCont.start();
}

This code basically initializes the Awareness Server. It initializes the AMAccessLayer, AMController and the communicator. It sets reference between the accessLayer the controller and also between the controller and the communicator. The controller is started which in turn starts the communicator.

Important point to note is that when the communicator starts the server. The server is hard coded to listen on port 8090.

Server will be listening on port 8090 and if a client request comes then it creates a serverThread for it. The communication will take place between the client and the serverThread. For each client accepted a separate serverThread is created.

Awareness Client:-

When the User opens the simulator client he is presented with a login window in which he enters his user Id, Password and the Ip address of the Awareness Server to be connected and clicks submit. The name of the form that is presented is frmLogin. The code file associated is frmLogin.cs. When the user clicks Login button *btnLogin_Click* function is called. A connection is established with the awareness Server using the IP address. An object for the User Controller i.e., the uCon is created. Reference is set in the user controller to refer to the frmLogin as *"parent"* variable to refer to the form. The networkStream(for InputStreams and OutputStreams) details of the client are sent to the User Controller because it handles all the communication on behalf of the client.

If the connection is established then a Listener thread is started whose duty is to listen to the messages sent by the awareness server and communicate them to the User Controller.

The Way thread is started in C# is

Thread userLThread = new Thread (
            new ThreadStart(userListenerThread.run)
            );
userLThread.Start();

ThreadStart indicates the starting point of the thread. In this case we indicate that the run method in the userListenerThread object as the Starting point. userLThread.Start() starts

the execution of the run method in a different thread. It is important to understand the concept of threading to understand the project.

After the Listener Thread is spawned an authentication message is sent to the server. Any message sent to the Server is send through the User Controller. The user name and password are sent to the user controller authenticate method which then adds appropriate tags and send it to the sendUserEvents method. This method makes use of the Parser to pack the message into XML Message and send it to the Awareness Server.

Server Side:-

Server Thread's run Method receives the message calls sends it to the communicator's *receiveMessage* method which sends the message to the controller's *controller* method. The method *unpacks* the XML message into a messageInfo object using the parser. Then calls the appropriate method. In this case the *handleAuthentication* method.

In the *handleAuthentication* method a User object is created which contains User ID, Input and output streams and is added to the userList which is a vector of User objects maintained by the Communicator.
Password is retrieved using the AMAccessLayer's *getPassword* method.
The password is compared with the password entered by the user and if it matches the user is send a XML message with the Awareness Map and the src Super Set. Otherwise a reply is sent with reply that the user is not authenticated and the user is removed from the userList and the Thread is stopped. The messages are explained in detail later.

Client Side:-

The UserListenerThread will receive a reply message from the Awareness Server. The Listener Thread sends this message to the User Controller. The Controller now unpacks this message using the parser into messageInfo object. The reply message(assuming that the user is authenticated) is of this format:-

<Message>
<MessageType>AUTH_REPLY</MessageType>
<UserId>sup</UserId>
<Reply>AUTHENTICATED</Reply>
<RunId>demo1</RunId>
<AwarenessMap>
 <FocusList>
  <Focus>
   <FocusId>nofocus</FocusId>
   <SrcId>s6</SrcId>
   <SrcId>s1</SrcId>
   <SrcId>s5</SrcId>
  </Focus>

```
 </FocusList>
</AwarenessMap>
<SrcSuperSet>
        <UserId>all</UserId>
        <SrcId>s1</SrcId>
        <Active>yes</Active>
        <UserId>emp</UserId>
        <SrcId>s10</SrcId>
        <Active>no</Active>
...
</SrcSuperSet>
</Message>
```

Unpacking of this XML Message involves
1) Getting the reply into reply variable of the MessageInfo object
2) Getting the awareness map information in a ArrayList named awarenessMap. awarenessMap is a ArrayList which holds objects of type FocusInfo. The FocusInfo object consists of two fields. The FocusId and the srcList which is a ArrayList which holds the srcId's of sources in that focus.
3) Getting the source super set information in a ArrayList named srcSuperSet. srcSuperSet is a ArrayList which holds objects of type UserSrcInfo. UserSrcInfo objects consist of three fields. The User Id, Src Id and Active.

Now the User Controller makes use of the reply variable to display the frmAwarenessClient.

Design Issues:-
    The process of displaying the new Form is done in a separate thread. When the form is displayed in the same thread we are not able to perform any operations on the form so we kept the process of displaying the form in a separate thread. Also we need to update the displayed form. For this purpose we need a reference to the newly displayed form. This reference is being set in the newly spawned thread. So we sleep the current thread so that the reference is being set in the other thread.

User Controller makes use of the array lists to display the awareness map and the source super set.

It calls the updateAwarenessMap function to display the awarenessMap and the updateSrcSuperSet function to display the srcSuperSet. Please refer to the code to have a better understanding of how the source super set and the awarenessMap are displayed. After the frmAwarenessClient is updated we create a file to store all the collaborative events.

Issues while displaying source super set in the TreeView Control

The sourcesuper set displayed is a treeView control. The User
Controller which is running on a different thread and needs to update
the frmAwarenessClient's userList (tree control) to represent the
latest state of source super set. But the problem is that the tree
control cannot be updated from a different thread, it should be updated
from the same thread. So here we are making use of a delegate function.
The User Controller call the updateSrcSuperSet function. Which in turn
call the treeListDelegate. the treeListDelegate is associated with the
updateTree Function. By using the BeginInvoke method the execution of
the update is delegated to the thread which created the control. Hence
solution to the problem.

Issues while displaying the initial awareness map.

We assumed a few things to make the display of awareness map easy.

      1) The size and positions of the awareness map are fixed.
      2) There can be a maximum of 4 sources for each foci.
      3) There are a maximum of 12 sources for each User in the IP set.

The way the initial awareness map is displayed is:-

The awarenessMap arrayList contains information about the foci and the sources in the foci. So we take each focus and display the sources in it (positions pre determined). Please refer to the code for complete understanding.

Issues While displaying Enhanced Awareness:-
    The Enhanced Awareness i.e., Second order information, Reverse Lookup, Wisiwys, Source Info are all shown in a new form. The process of displaying the new Form is done in a separate thread. When the form is displayed in the same thread we are not able to perform any operations on the form so we kept the process of displaying the form in a separate thread. Also we need to update the displayed form. For this purpose we need a reference to the newly displayed form. This reference is being set in the newly spawned thread. So we sleep the current thread until the reference is being set in the other thread.

Agent Script

The User has the choice to choose between manual simulation or deploy a agent script When the User is provided with the Simulator Client. He is given an option to Run Script at the top right hand corner. On clicking that he will have to select the script to run. A sample Script is store at /awproject/AgentScript/agentscript.txt. The client will execute the messages from the script. This script should consist of text messages which conform to the messaging protocol.

Bugs

The SrcSuperSet is refreshed whenever a source is switched ON/OFF. When the refresh is being done the user cannot perform any operations on the SrcSuperSet. If he performs any action then he gets an error message. Click continue to continue the operation.

Messaging Protocol

Protocol is defined as a set of syntactic and semantic rules for exchanging information that includes (a) syntax of the information; (b) semantics of the information; and (c) rules for the exchange of information. The message protocol format used for the awareness emulator is explained below in detail. It mainly consists of three types of messages. There are three types of message:-

- Request Message:- This is an XML Message sent by the client requesting some information from the awareness model or making changes to the awareness model. The ADD_FOCUS, DEL_FOCUS, ADD_SOURCE, DEL_SOURCE, AMM ,GEN, PER, GET_SECOND_ORDER, DO_REVERSE_LOOKUP all come under this type.

- Notification/Propagation Message:- When a User performs a change in the Awareness model and this changes needs to be propagated to other users then a notification or a propagation message is sent to them. An example of a notification scenario would be ascertaining the status of the users. When a users switches on his source, a notification is sent to all the users interested in that user, source pair. An example of a propagation message is an Awareness Model propagation message. When a user sends a awareness model message as a source. The message needs to be propagated to all the users interested in that user, source pair. An Awareness model propagation message is sent to all the interested users. The notification/propagation messages include. SWITCH_NOT – Notification of a source being switched ON/OFF, SOURCE_NOT – When a users source is

added to other users Focus. AMM_PROP – Awareness message sent to interested

users. PER_PROP- Personal message sent to the interested user.

- Reply Message:- This is an Xml Message sent back to the user to indicate the

  success/failure of the requested awareness Event. An example would be

  ADD_FOCUS_REPLY which indicates whether or not the change is made to the

  Awareness model.

Message Types

   1. Authentication:- The User sends an XML Message of the format. The text format of the message is

<div align="center">*AUTH userId Password*</div>

    XML Message is
      <Message>
      <MessageType> AUTH </MessageType>
      <UserId>   USERID   </UserId>
      <PassWord>   PWD   </PassWord>
      </Message>

  The Awareness Controller then receives this message, contacts the Awareness Model and sends another XML Message to the client.
Text format
    AUTH_REPLY Authenticated/NotAuthenticated RunId awarenessMap srcsuperset

      <Message>
      <MessageType> AUTH_REPLY </MessageType>
      <Reply> Authenticated/NotAuthenticated <Reply>
      <RunId> RunId </RunId>

      <AwarenessMap>
      <FocusList>
      <Focus>
      <FocusId> FocusId </FocusId>
      <SrcId> SrcId1 </SrcId>
      <SrcId> SrcId2 </SrcId>
      </Focus>
      </FocusList>
      </AwarenessMap>

      <SrcSuperSet>
      <UserId> userId </UserId>
      <SrcId> SrcId </SrcId>
      <Active>Active </Active>
      …
      …
      </SrcSuperSet>
      </Message>

2) Adding Focus:- When the User performs a gui action to create a focus. The User will be sending a XML Message of the format

        <Message>
        <MessageType> ADD_FOCUS </MessageType>
        <UserId>   USERID   </UserId>
        <FocusId>   FOCUS   </FocusId>
        </Message>

The Text Format of the message passed to the packer is
                      ADD_FOCUS userId FocusId

   The Awareness controller then receives the message through the communicator and makes modification to the Awareness Model. It then sends a XML Message to the client indicating that the awareness model was changed successfully.

        <Message>
        <MessageType> ADD_FOCUS_REPLY   </MessageType>
        <FocusId>   FOCUS   </FocusId>
        <Reply>   ADDED   </Reply>
        </Message>

The text format of the message passed to the packer is
             ADD_FOCUS_REPLY focus ID ADDED/NOT_ADDED

3) Deleting Focus:- When the User performs a gui action to create a focus. The User will be sending a XML Message of the format

> <Message>
> <MessageType> DEL_FOCUS </MessageType>
> <UserId>   USERID   </UserId>
> <FocusId>   FOCUSID   </FocusId>
> </Message>

The Text Format of the message is

<div align="center">*DEL_FOCUS userID FocusId*</div>

   The Awareness controller then receives the message through the communicator and makes modification to the Awareness Model. It then sends a XML Message to the client indicating that the awareness model was changed successfully.

> <Message>
> <MessageType> DEL_FOCUS_REPLY   </MessageType>
> <FocusId>   FOCUS   </FocusId>
> <Reply>   DELETED   </Reply>
> </Message>

The Text format of the message is

<div align="center">*DEL_FOCUS_REPLY Focus Id DELETED/NOT_DELETED*</div>

4) Adding Source:-When the User adds a source to a focus. The user wil be sending the following XML message -


```
<Message>
<MessageType>  ADD_SOURCE          </MessageType>
<UserId>            USERID                </UserId>
<FocusId>          FOCUS                 </FocusId>
<SourceId>        SOURCE               </SourceId>
</Message>
```

The Text Format of the message passed to the packer is
                        ADD_SOURCE userId FocusId SourceID

   The Awareness Controller then receives the message through the Awareness Communicator and makes modification to the Awareness model. It then sends a Xml Message to the client indicating that the awareness model was changed successfully.


```
<Message>
<MessageType>  ADD_SOURCE_REPLY          </MessageType>
<FocusId>     FOCUS                          </FocusId>
<SourceId>    SOURCE                           </SourceId>
<Reply>            SOURCE_ADDED   </Reply>
</Message>
```
The Text Message is:
         ADD_SOURCE_REPLY focus ID SourceID ADDED/NOT_ADDED

5) *Deleting Source*:-When the User adds a source to a focus. The user wil be sending the following XML message -

```
<Message>
<MessageType> DEL_SOURCE </MessageType>
<UserId>     USERID              </UserId>
<FocusId>   FOCUS          </FocusId>
<SourceId>   SOURCE         </SourceId>
</Message>
```

The Text Format of the message is

DEL_SOURCE userID FocusId SourceId

The Awareness Controller then receives the message through the Awareness Communicator and makes modification to the Awareness model. It then sends a Xml Message to the client indicating that the awareness model was changed s successfully.

```
<Message>
<MessageType> DEL_SOURCE_REPLY        </MessageType>
<FocusId>    FOCUS                        </FocusId>
<SourceId>   SOURCE                       <SourceId>
<Reply>       SOURCE_DELETED   </Reply>
</Message>
```

The Text format of the message is

DEL_SOURCE_REPLY Focus Id SourceId DELETED/NOT_DELETED

6)Awareness Model Message :- When the user types a Awareness Model message the message is sent to the all the users who are interested in the source the user is logged in as.

AMM source_id AMMessage

is what the user types in the user Events where source_id is the source as which the user is sending the message

An XML Message is sent to the Awareness Controller of the format.

```
<Message>
<MessageType>  AMM                </MessageType>
<UserId>        USERID            </UserId>
<SrcId>         SOURCEID           </SrcId>
<AMMessage>  AMMESSAGE   </ AMMessage>
</Message>
```

The Awareness Controller then communicates the message to the Users who have the source in their focus by contacting the Awareness Model. It then sends a message to the user of the format. Where UserId and SrcId are the From UserId and SrcId .

```
<Message>
<MessageType>   AMM_PROP         </MessageType>
<UserId>          USERID          </UserId>
<SrcId>           SOURCEID         </SrcId>
<AMMessage>    AMMESSAGE         </AMMessage>
</Message>
```

The Awareness Controller also sends a reply back to the user who sent the Awareness Model Message. The reply consists of details of the users who received the message.

```
<Message>
<MessageType>   AMM_REPLY         </MessageType>
<UserId>          USERID          </UserId>
<SrcId>           SOURCEID         </SrcId>
<RecvUserList>   RECVUSERLIST   </RecvUserList>
<AMMessage>    AMMESSAGE         </AMMessage>
</Message>
```

276

7) *Personal Message* :- Personal message can be sent by any user to any user. He needs to know the source of the user to which the message should be sent.
 He types in

PER sourceId recvUserId recvSourceId PerMessage.

 The sourceId is the source as which the user is sending the source, recvUserId and recvSourceId are the receiver's User id and Source Id.
 An XML Message is sent to the Awareness Controller of the format.

```
<Message>
<MessageType>   PER                 </MessageType>
<UserId>        USERID              </UserId>
<SrcId>         SOURCEID            </SrcId>
<RecvUserId>    RECVUSERID          </RecvUserId>
<RecvSrcId>     RECVSOURCEID        </RecvSrcId>
<PerMessage>    PERMESSAGE          </ PerMessage>
</Message>
```

The Awareness Controller then communicates the message to the Users who have the source in their focus by contacting the Awareness Model. It then sends a message to the user of the format.

```
<Message>
<MessageType> PER_PROP    </MessageType>
<UserId>    USERID              </UserId>
<SrcId>  SOURCEID            </SrcId>
<RecvUserId>    RECVUSERID      </RecvUserId>
<RecvSrcId>     RECVSOURCEID    </RecvSrcId>
<PerMessage>    PERMESSAGE      </ PerMessage>
</Message>
```

The Awareness Controller also sends a reply message to the sender informing whether this message is sent to the intended user or not.

```
<Message>
<MessageType>  PER_REPLY            </MessageType>
<UserId>        USERID              </UserId>
<SrcId>         SOURCEID            </SrcId>
<RecvUserId>    RECVUSERID          </RecvUserId>
<RecvSrcId>     RECVSOURCEID        </RecvSrcId>
<Reply>         REPLY               </Reply>
<PerMessage>    PERMESSAGE          </PerMessage>
</Message>
```

8) *Viewing Source Info*:- When the user wants to view a source's information. He performs an action on the GUI .Then a message of this format is generated.

```
<Message>
<MessageType>  VIEW_SOURCE_INFO   </MessageType>
<UserId>     USERID                  </UserId>
<SourceId>   SOURCEID                </SourceId>
</Message>
```

The Text format of the message is

VIEW_SOURCE_INFO SourceId

The controller then contacts the awareness model and returns the result to the user
In the format.
```
<Message>
<MessageType> VIEW_SOURCE_INFO_REPLY        </MessageType>
<UserId>           USERID                 </UserId>
<SourceId>         SOURCE              </SourceId>
<SourceName>  SOURCENAME           </SourceName>
<Description>   DESCRIPTION            </Description>
<PreferredMed>  PREFERREDMEDIUM   </PreferredMed>
<CP> cp</CP>
<IP> ip   </IP>
</Message>
```

The Text format of the message is

*VEW_SOURCE_INFO_REPLY SourceID SourceName Description Preferred Medium
Cp Ip*

278

9) View Second Order Info:- When the user wants to view the foci of the users in his focus. He performs some GUI action for which the following XML Message is generated.

    <Message>
    <MessageType> GET_SECOND_ORDER </MessageType>
    <UserId> UserId </UserId>
    <UserInFocus> UserId </UserInFocus>
    </Message>

The Text format of the message is

Get_Second_order UserID UserInFocus

The Awareness Controller then Receives the Message through the Communicator and then contacts the Awareness Model to retrieve the second order information of the user. This information is then sent to the user in the form of a XML Message.
    <Message>
    <MessageType> GET_SECOND_ORDER_REPLY </MessageType>
    <UserId> UserId </UserId>
    <UserInFocus> UserId <UserInFocus>
    <Reply>   REPLY </Reply>
    <AwarenessMap>
    <FocusList>
    <Focus>
    <FocusId> Focus1 </FocusId>
    <SourceId> s1</SourceId>
    <SourceId> s2</SourceId>
    </Focus>
    </FocusList>
    </AwarenessMap>
    </Message>
The Text format of the message is

GET_SECOND_ORDER_REPLY UserID UserInFocus reply AwarenessMap

10) Reverse Lookup:- When the user wants to view the other users who have the source in their focus. He performs some GUI event. For which the following XML Message is generated and is sent to the Controller through the Communicator.

<Message>
<MessageType> DO_REVERSE_LOOKUP </MessageType>
<UserId> UserId </UserId>
<SourceId> SourceId </SourceId>
</Message>
 The Text format of the message is

DO_REVERSE_LOOKUP UserID SourceId

Now the communicator determines if the user has the privilege and then sends the following information.

<Message>
<MessageType> DO_REVERSE_LOOKUP_REPLY </MessageType>
<SourceId>   SOURCEID   </SourceId>
<UserList>
<User>
<RevUserId> RevUserId </RevUserId>
<RevFocusId> RevFocusId </RevFocusId>
</User>
….
</UserList>
</Message>

 The Text format of the message is

DO_REVERSE_LOOKUP_REPLY SOURCE_ID USERLIST

11)WISIWYS:- The User can know Which other users are seeing the source he is looking at. He can do a WISIWYS on a source to get a list of users who can see that source.

The Controller receives a WISIWYS message of this format
<Message>
<MessageType> WISIWYS </MessageType>
<UserId> UserId </UserId>
<SourceId> SourceId </SourceId>
</Message>

The Text format of the message is

WISIWYS UserID SourceID

The controller contacts the awareness model and returns a message of this format

<Message>
<MessageType> WISIWYS_REPLY </MessageType>
<SourceId>   SourceId   </SourceId>
<UserList>
<User>
<WysUserId>   wysUserId   </WysUserId>
<WysFocusId> wysFocusId   </WysFocusId>
</User>
….
</UserList>
</Message>

 The Text format of the message is

WISIWYS_REPLY SourceId UserList

12)Switching Sources:-

The user can switch on his sources by sending the following XML Message.

```
<Message>
<MessageType> SWITCH   </MessageType>
<UserId>   USERID     </UserId>
<SrcId    SOURCEID </SrcId
<Action>  ON/OFF         </ACTION>
</Message>
```
The Text format of the message is

SWITCH UserID SourceId ON/OFF

The Users who have the source in their Source Super Set get a notification

```
<Message>
<MessageType> SWITCH_NOT </MessageType>
<UserId> USERID </UserId>
<SrcId> SOURCEID </SrcId>
<Reply> SOURCE_ACTIVATED/SOURCE_DEACTIVATED </Reply>
<SrcSuperSet>
<UserId> userId </UserId>
<SrcId> SrcId </SrcId>
<Active>Active </Active>
</SrcSuperSet>
</Message>
```

The Text format of the message is

SWITCH_NOT UserID SourceId SrcSuperSet

The user who switched the source will get a SWITCH_REPLY message

```
<Message>
<MessageType> SWITCH_REPLY </MessageType>
<UserId> USERID </UserId>
<SrcId> SOURCEID </SrcId>
<Reply> SOURCE_ACTIVATED/SOURCE_DEACTIVATED </Reply>
<SrcSuperSet>
<UserId> userId </UserId>
<SrcId> SrcId </SrcId>
<Active>Active </Active>
</SrcSuperSet>
</Message>
```

The Text format of the message is

SWITCH_REPLY UserID SourceId SrcSuperSet

13) *MisMatch/MisMatchAvoided* :- The user might find a mismatch or a mismatch Avoided. He types a message of format

<div align="center">

MM/MMA Rational.

</div>

This is sent as a XML Message of the format

```
<Message>
<MessageType> MM/MMA</MessageType>
<UserId> USERID </UserId>
<Rational> rational </Rational>
</Message>
```

The Text format of the message is

<div align="center">

MM/MMA UserID rational

</div>

The message gets logged in and a reply message is sent back to the user

```
<Message>
<MessageType> MM_REPLY/MMA_REPLY</MessageType>
<UserId> USERID </UserId>
<Rational> rational </Rational>
</Message>
```

The Text format of the message is

<div align="center">

MM_REPLY/MMA_REPLY UserID rationa

</div>

Appendix D
Awareness Simulator Source Code

For details, information and any other questions regarding the Awareness Simulator application source code please contact the author at vanandb@hotmail.com.

Curriculum Vitae

Vijayanand Bharadwaj
vanandb@hotmail.com

Education

Ph.D. **Computer & Information Sciences**   **West Virginia University USA, December 2005**
M.S.   **Computer and Information Sciences**   **West Virginia University USA, May 2000**
B.E.   **Electronics Engineering**   **Bombay (Mumbai) University India, May 1997**

Doctoral Dissertation: *Supporting Awareness in Heterogeneous Collaboration Environments*
Master's Thesis: *Web-Based Workflow in Secure Collaborative Telemedicine*
(https://eidr.wvu.edu/eidr/documentdata.eIDR?documentid=1267)

I. Professional Experience

1. Research Experience
Graduate Research Assistant at the following institutions at West Virginia University.
August 2004 – Present   Smart Internet Programming Laboratory (SIPLab, CERC)
November 2003 – August 2004 Advanced Power & Electricity Research Center (APERC)
May 2000 – November 2003   Virtual Medical Campus (VMC)
August 1997-May 2000   Concurrent Engineering Research Center (CERC)
June 1996 – May 1997   Crompton Greaves Ltd., Bombay, India (Trainee)
I have worked on various projects sponsored by both U.S. federal and other agencies such as NLM, DHHS, Dept. of Homeland Security, DOE, NASA, NSF and Bell Atlantic. Brief descriptions of all projects provided below. I have authored several research papers, participated in conference discussion panels and served on the Program Committee of an IEEE conference (details below).

2. Research Guidance Experience
Since mid-2003 assisting advisor Dr. Y.V. Ramana Reddy in guiding the graduate research component of students at LDCSEE, WVU. Ten students with whom I have worked with, have completed their Master's research and graduated. Eight current students are expected to graduate in May 2006. My work involves aiding them in defining their research topic & requirements, project design, leading the group in project development activities and evaluation of their work.

3. Teaching Experience
I have assisted Dr Y.V.R. Reddy in teaching specific topics in Computer Networking and Artificial Intelligence (including creation & grading of exams & term projects). I have also taught a specific topic of Computer Systems (Interprocess communications) for Dr. Bojan Cukic at WVU. I am a Group Fitness Instructor at the WVU, Student Recreation Center. Since 2001 I have been teaching a variety of fitness classes. I am certified by the National Exercise Trainer's Association (NETA).
(http://www.sa.wvu.edu/studentrecreationcenter/weight&fitness.shtml)
Video of Teaching CS Course & Research Presentation available here (others on request):
http://siplab.csee.wvu.edu/CS554/lectures/09202005_L5P1_Streaming.wmv
http://siplab.csee.wvu.edu/research/EkSarva/CDVE2005/CDVE2005_Streaming.wmv

II. Research Interests

Primary: Computer Supported Cooperative Work, Awareness Systems, Human-Computer Interaction, Interoperability in Group Work, Distributed Computing. Software Engineering, including architecture and design of N-tier information systems, Maintainability of systems.

Secondary: Knowledge Transfer in Academic Research Groups, Intelligent Agents and Workflow Systems, Service-Oriented Architectures, Computer Science Education Curriculum Development & Teaching Methodologies, Information Visualization of Large Volume Data and Interoperability issues, Arrangement of Information Spaces, Some interest in Simulation Science (agent-based & conventional) and Software Quality Assurance (Reliability).

III. Projects

1. EkSarva – *A Workflow-Centric Context-Aware Collaboration Framework (August 2003-Present): http://siplab.csee.wvu.edu/research/EkSarva (Supported by Bell Atlantic endowment)*
Currently at SIPlab I am part of team involved in the design and development of a collaboration environment that improves the quality of collaboration by use of intelligent software agents, adaptive workflow and context-awareness. I am involved in designing an expert-systems like paradigm in enabling dynamic workflow. Collaboration processes are modeled as objects and rules and the workflow engine enacts the processes with humans in the loop. Initial version of a rapid prototype included using Protégé a knowledge-base editor (Java based ontology creator) as a core component.

2. Awareness Project – *Supporting Awareness in Heterogeneous Collaboration Environments (June 2004-Present):*
As part of my dissertation, I have been studying the impact of heterogeneous environments on awareness propagation in collaborating groups. Having identified the problems in awareness propagation, I have devised an awareness model to integrate a variety of information sources and empower users with the ability tailor the quality of awareness information. The model was validated by creating an Awareness Simulator application along with graduate students at SIPlab. Some technologies used in this research effort were Java, .NET C# and RDBMS (MySQL).

3. TEMS – *Testbed for Energy Market Simulations (June 2003- August 2004)*
http://www.aperc.wvu.edu (Sponsors U.S. Dept. of Energy and WVU EPSCoR Program)
At APERC I was part of an interdisciplinary research effort involving computer scientists, decision scientists, economists and electrical engineers from WVU and WVTech. Our objective was to study various issues related to power quality, supply and demand in a deregulated power industry. Apart from analytical methods economists use decision theory for such analysis. To facilitate the application of decision theory to deregulated power industry I was responsible for creating an agent-based simulation environment. A three-layered architecture for the simulator was specified, with the three layers corresponding to the electrical, economic and human decision aspects that are prevalent in the power industries today. My work included an extensive literature review of agent-based simulation of power markets, and the specification of the TEMS concept with some design decisions. A simple prototype was also implemented in Java using the RePAST toolkit.

4. IKB-LMS for VMC – *Integrated Knowledge Base & Learning Management System for the Virtual Medical Campus (May 2000 – November 2003)*
http://www.vmc.wvu.edu (Sponsors DHHS, DOJ, now under Dept of Homeland Security).
The Virtual Medical Campus at WVU, is responsible for creating and disseminating training material (courses & content) for Emergency First Responders (Firefighters, EMT and EMS). The training is for emergency preparedness and response for disasters due to Weapons of Mass Destruction (WMD). The Integrated Knowledge Base and Learning Management System are two essential components in the aiding Subject Matter Experts and instruction designers in the design, creation, dissemination of training material, the testing, certification and continuing education for Emergency First Responders. I was responsible for assisting researches from the sociology department at WVU, in creating surveys about the state of emergency preparedness which were sent out to first responders in all fifty states of the USA. I was part of a team creating requirements (software and system) for the IKB component of the VMC. Apart from requirements creation I was responsible for creating software for certain modules of the IKB and activities such as gathering and classification of content used by SMEs to create test oracles for the IKB software, as well assisted the VMC staff in a live simulated exercise "Operation Mountain Thunder" to test emergency preparedness of local first responders. Our partners in the initial phase of requirements elicitation were Electronic Data Systems (Virginia).

5. CBRE - *Component based Software Reliability Estimation (May 2000- May 2002):*
(Sponsors NASA, and CAREER award from NSF)
The objective of this project was to determine the quality of software in terms of reliability when the software system was in the design phase as opposed to waiting for the testing phase as done traditionally. I was responsible for creating and running simulations of reliability models that were devised and applied to applications created using software components. With prior reliabilities of the components known the aim of the model was to predict the total reliability of entire system. This project was part of larger research effort in software quality assurance. Simulations were conducted using the Mathematica package.

6. Secure Collaborative Telemedicine - *Web-based Workflow System (August 1997- May 2000)*
http://www.cerc.wvu.edu/nlm/telemedicine/Default.htm (Sponsor National Library of Medicine)
This project investigated the manner in which healthcare could be provided to patients in an efficient and secure manner by using various technologies such as middleware, the Internet & WWW (growing rapidly in popularity then), smart card technology and secure encryption to facilitate electronic medical records, remote monitoring while employing healthcare standards such as HL7. Applications were created and deployed so that healthcare providers in cities such as Huntington (WV) could examine and administer care to remote patients in rural areas. Smart cards were used to hold patient health records and interfaced with databases. Technologies used were the Schlumberger Smart Card API and Microsoft SQL Server. I was responsible for creating a web-based workflow system for healthcare professionals and insurance providers to request and approve patient benefits. The workflow automated various aspects such as entering patient information (data was automatically read in from smart cards), client-side scripting for validation and email, all using secure protocols. The system was built using, Active Server Pages and HTML running on Microsoft's Internet Information Server that interfaced with the backend RDBMS. To ensure secure access the web-server was enabled with the SSL protocol. The workflow included Microsft Exchange Server component to provide secure messaging. Later I extended the system so that mobile healthcare providers could also request benefits authorization using hand-held devices. This was done using Java 2 Mobile Environment (Midlets) and Servlet Technology.

IV. Selected Publications

Bharadwaj, V., Reddy, Y.V.R. and Reddy, S., **"Integrating Awareness Sources in Heterogeneous Collaboration Environments"**, in *Proceedings of the 2nd International Workshop on Computer Supported Activity Coordination, CSAC 2005 In conjunction with International Conference on Enterprise Information Systems (ICEIS) 2005*, Miami, U.S.A., May 2005.

Bharadwaj, V. and Reddy, Y.V.R., **"A Framework to Support Collaboration in Heterogeneous Environments"**, in *ACM SIGGROUP Bulletin, Special issue on community-based learning: explorations into theoretical groundings, empirical findings and computer support*, Volume 24, Issue 3, December 2004, ACM Press, New York, 2004.

Bharadwaj, V., Reddy, Y.V.R., Chandrmouli, A., and Reddy, S., **"A Layered Architecture to Model Interdisciplinary Complexity in the Deregulated Power Industry,"** in *Proceedings of the 2004 IEEE Power Systems Conference and Exposition, (PSCE 2004),* New York, NY, October 10-13, 2004.

Reddy, R., Selliah, S., Bharadwaj, V., Yu, J., Reddy, S. and Kankanahalli, **"Eksarva: An Intelligent Collaboration Framework"**, *in Proceedings of the IEEE International Conference 'Intelligent Systems',* Varna, Bulgaria, June 22-24, 2004.

Bharadwaj, V., Reddy, Y.V.R., Kankanahalli, S., Reddy, S., Selliah, S. and Yu, J., **"Evaluating Adaptability in Frameworks that Support Morphing Collaboration Patterns"**, in *Proceedings of the 13th IEEE International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises (WETICE-2004) Evaluation of Collaborative Information Systems and Support for Virtual Enterprises (ECE) Workshop,* Modena, Italy, June 14-16, 2004.

Reddy, R., Selliah, S., Bharadwaj, V., Yu, J., Reddy, S. and Kankanahalli, S., **"An Epistemological View of Collaboration"**, Keynote Address, *in Proceedings of the Eighth International Conference on CSCW in Design (CSCWD 2004)*, Xiamen Province, Peoples Republic of China, May 26-28, 2004.

Bharadwaj, V., Raman, R., Reddy, R., Reddy, S., **"Empowering Mobile Healthcare Providers via a Patient Benefits Authorization Service"**, *in Proceedings of Tenth IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE 2001)*, Massachusetts, USA, June 20 - 22, 2001.

H. Singh, V. Cortellessa, B. Cukic, E. Gunel, V. Bharadwaj, **"A Bayesian Approach to Reliability Prediction and Assessment of Component Based Systems"**, *in Proceedings of the 12th International Symposium on Software Reliability Engineering (ISSRE'01)*, Hong Kong, China, November 27-30, 2001.

V. Professional Activities

Program Committee Member for the "Workshop on Evaluation of Collaborative Information Systems and Support for Virtual Enterprises (ECE)" track of the 14th IEEE International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises (WETICE-2005), Linköpings universitet, Linköping, Sweden. http://hemswell.lincoln.ac.uk/wetice05/index.php#pc

VI. Technologies & Skills

Methodologies: Experience in Software & System Requirements Elicitation, Analysis and Design.
Object-Oriented Design Methodology, Use of Unified Modeling Language (UML), N-tier Architecture design and development.

Technologies: J2EE (including Enterprise Java Bean, Servlets and JSP), Microsoft .NET, XML, Active Server Pages (Classic ASP 2.0), PHP, ActiveX Controls, Dynamic HTML, M.S. Active Data Objects (ADO), Collaboration Data Objects (CDO). Familiarity with Secure Sockets Layer and Internet Security. Some familiarity with Resource Description Framework (RDF), Semantic Web Concepts (OWL) and mobile technology J2ME.

Knowledge-Based tools: Ontology Editor Protégé, some familiarity with JESS, JENA.
Simulation Environments and Agent-based Toolkits: Some familiarity with Matlab, Mathematica, RePAST (Java counterpart of SWARM), MADkit, BRAHMS

Programming Languages: Java, C/C++, Ada, some familiarity with Fortran, Pascal.
Scripting Languages & Web: VBscript, JScript, JavaScript, HTML, Cascading Style Sheets
Database Technologies: SQL (Worked using Orcale 8i, Microsoft SQL Server 6.5, MySQL 4.0)
Development Environment & tools: Microsoft Visual Studio, (.NET, C++,) Microsoft FrontPage, MS Office Suite, MS Exchange, Novell's Groupwise, Visual Interdev 6.0, Paintshop Pro, some knowledge of Eclipse IDE, and others.
Server Technology: Basic System Administration of servers, Windows XP, 2000 Professional, Linux, Microsoft Internet Information Server 4.0, Microsoft Exchange Server 5.5, Microsoft SQL Server 6.5, Microsoft Personal Web Server
Operating Systems: Windows XP/2000 Professional/98/ NT 4.0, Sun Solaris, Linux.

VII. Relevant Courses Taken

Global Knowledge Networks, Internet Technologies, E-Commerce/E-Services, Software Performance Engineering, Software Reuse, Applied Microeconomics, Fault-Tolerant Computing, System Lifecycle & Configuration Management, Operating Systems Theory. Complete list of courses is available on request.

VIII. Awards Received

Received a National Science Foundation (NSF) travel grant from Florida International University to the Doctoral Consortium at the International Conference on Enterprise Information Systems (ICEIS 2005) Miami Florida. Among the twelve selected out of fifty applicants in the USA.

Received a partial travel grant to present a poster at a competition held as part of the IEEE Power Engineering Society General Meeting (PESGM 2004 International Conference) Denver 2004.

IX. Membership to Professional Societies

Institute of Electrical and Electronics Engineers IEEE

Association of Computing Machinery ACM

Upsilon Pi Epsilon UpE Computer Science Honor Society

American Society for Quality (ASQ)

X. Certifications

Group Fitness Instructor Certification from the National Exercise Trainer's Association
Certified to teach a variety of fitness classes including Step Aerobics, Cycling, Kickboxing, Total Body Muscle Conditioning, Abs & Back and Hi-Lo (High-Intensity Low-impact Cardio).

XI. References

Available upon request.