DEVELOPING A GROUP MODEL FOR STUDENT SOFTWARE ENGINEERING TEAMS

A Thesis Submitted to the College of

Graduate Studies and Research

in Partial Fulfillment of the Requirements

For the Degree of M.Sc

in the Department of Computer Science

University of Saskatchewan

Saskatoon, Saskatchewan

by

Mike Winter

2004

Permission To Use

In presenting this thesis in partial fulfillment of the requirements for a Postgraduate degree from the University of Saskatchewan, I agree that the Libraries of this University may make it freely available for inspection. I further agree that permission for copying of this thesis in any manner, in whole or in part, for scholarly purposes may be granted by the professor or professors who supervised my thesis work or, in their absence, by the Head of the Department or the Dean of the College in which my thesis work was done. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to the University of Saskatchewan in any scholarly use which may be made of any material in my thesis.

Requests for permission to copy or to make other use of material in this thesis in whole or part should be addressed to:

Head of the Department of Computer Science
University of Saskatchewan
Saskatoon, Saskatchewan, Canada
S7N 5A9

Abstract

Work on developing team models for use in adaptive systems generally and intelligent tutoring systems more specifically has largely focused on the task skills or learning efficacy of teams working on short-term projects in highly-controlled virtual environments. In this work, we report on the development of a balanced team model that takes into account task skills, teamwork behaviours and team workflow that has been empirically evaluated via an uncontrolled real-world long-term pilot study of student software engineering teams. We also discuss the use of the J4.8 machine learning algorithm with our team model in the construction of a team performance prediction system.

Acknowledgements

To my family.

Table Of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Developing tools to support collaborative activity and learning has become a primary focus of the Intelligent Tutoring Systems and Computer-Supported Collaborative Learning research communities [66]. The theoretical basis for this move is summarized by William Clancey: "rather than removing people from the loop, properly designed [computational] job aids help people with different expertise and roles to work together more effectively [50]."

In traditional single-user adaptive systems a user model is kept that records the user's knowledge, behaviour and preferences. The attributes in the user model are then used by the system to adapt itself to the user's needs and interests by, for example, recommending news stories of interest to the user or sorting the menu items in a word processing program by frequency of use. In the field of Intelligent Tutoring Systems, the adaptation often takes the form of modified curriculum, a change in pedagogical style or the delivery of context-specific help.

Single-user user models have been extensively studied, but that is not the case for models of groups of users. In fact, the discussion of the general issues surrounding the modelling of groups in adaptive educational environments has been relatively small to date, in spite of the proliferation of adaptive collaborative applications [66] [28] [5]. In particular, the work that does exist tends to focus on the task skills or learning behaviour of teams working on short-term projects in highly-controlled virtual environments [65] [38]. In our literature review, we categorize existing team models into two types: generative team models and collaborative-process team models. The first type of team model takes the individual user models of the team members and combines them in some way to form the team model. The second type of team model involves the analysis of teamwork in progress, and construction of the team model based on the quality and/or quantity of that teamwork. The most common type of systems in this group analyze the conversation between group members in collaborative learning environments in order to determine the level of learning that is happening within the group.

In contrast to the limited bandwidth and breadth of previous team modelling work, our research defines, studies and discusses the biggest questions surrounding the development, content and use of group models in adaptive collaborative applications. We start the process by developing a taxonomy of group types described in the online collaborative learning literature based on the groups' time of existence, task focus and cooperation level. The examination and categorization of different groups allows us to examine hidden and unexplored assumptions surrounding group models, namely:

1. Should functional knowledge be measured at an individual level or a group level?

2. How big a role do social and teamwork skills play in group success?

3. How effectively can social and teamwork skills be measured?

In answering those questions, the factors influencing group performance and behaviour, in particular for student software engineering teams, are identified and used to construct a team model that is usable in adaptive collaborative environments, or other applications concerned with team functioning. The team model is composed of four main parts, representing the general categories of factors that affect team performance: task competencies (domain knowledge), teamwork and social skills, workflow type and contextual influences. The novel breadth of coverage of this team model and the theoretical and empirical justifications for it form the crux of our work.

We then report on an empirical test of the team model involving a large-scale pilot study on teams in a third-year software engineering class that examines whether or not the factors identified as affecting team performance are applicable in a real-world situation and should therefore be considered as part of an effective team model. The student software-engineering teams in our study were tasked with a semester-long software development project that involved the design and implementation of an educational web application using Java and XML tools. An instance of our four-part team model was generated for each of the teams within our study with data gathered from questionnaires given to the students within the groups. At the conclusion of the project, various qualitative and quantitative measures were collected that reflected the performance of the groups and these measures were then compared against the team models to see how accurate the models were. As the study was a pilot, a large part of the analysis is focused on the nature of the study and possible improvements that can be made to future studies in this area.

Finally, to demonstrate a pragmatic use of the team model, the model is instantiated with results from the pilot study and the J4.8 classification algorithm is then used to build decision trees that predict the efficacy of arbitrary student software engineering teams.

The thesis is as follows: Chapter 2 shows the research surrounding group modelling, Chapter 3 details the setup of the pilot study and composition of the team model, Chapter 4 discusses the results of the study and Chapter 5 contains general conclusions and future work.

Chapter 2

Literature Review

2.1 The Motivation for a Group Model

The first step towards the construction of a group model is to examine the nature of groups and the attributes that affect their behaviour.

The following literature review is divided into three parts. The first is a survey of different types of groups studied in the Computer Science literature through the lens of the support tools developed for the groups. The general purpose of this section is to develop a better understanding of the attributes and purpose of groups that research into collaborative computing is focusing on. The more specific purpose of this section is to isolate the type of group that is most suitable for group model development.

In the second part of this chapter, the research relating to teamwork and group behaviour will be examined. There is concentration on how team performance can be monitored and evaluated, with the techniques examined including the measuring of shared mental models and questionnaires on specific teamwork behaviours. The final part of this section will explore the psychological and organizational behavioural instruments that measure the competencies of individuals in group settings, ranging from team role determination to social aptitude tests to knowledge measurement.

The third part of this chapter will look at the possible modelling and reasoning techniques available for the construction of the group model and the current application of these techniques in collaborative applications. Constraint logic programming, fuzzy logic and Bayesian networks, and their applications to the modelling of the behaviour and knowledge of individuals and groups will be considered.

2.2 Group Types and Their Attributes

2.2.1 Teams

Teams are a very common type of group in collaborative computer science research. A team is a planned and hierarchically ordered group of people working in an organization or multiple organizations that collaborate together on a well-defined task or set of tasks [1]. Teams are usually temporally limited by the scope of their project, as they are very task-centered. Examples of teams include project groups at software design companies or student groups that are assigned to do group projects as part of their class work. Of particular interest to computer science research is the way in which computer-mediated communication (CMC) and the advent of large scale networking has led to the formation of "virtual" teams, where the physical locations of the team members are distributed across geographical environments [1]. Many computer-based collaborative work and tutoring tools have been explicitly designed to support these virtual teams, and to be surrogates for existing and effective forms of "real-word" interaction and collaboration [62][61].

Many aspects of the functioning of teams have been studied, with an equally large and varied number of tools being built to support that functioning. Although tools may be intended for use by "purely virtual" teams, they can (and have been) adopted by teams that have a higher degree of physical co-location to enhance the team's communication and co-ordination abilities.

2.2.2 Intensional Networks

An intensional network is an informal collection of collaborators who are commonly brought together to accomplish specific tasks. An example of how intensional networks operate and what types of tasks they are used for is given in [2] where "Ed", an independent television producer, discusses how he uses resources from his intensional network to compose commercials:

> Ed: And that's a multidisciplinary kind of a task when we produce them [commercials]. We typically work with an ad agency. They come to you with storyboards. They ask you to budget out how much it would cost. You give them a bid and you sign that you are going to produce it for that and then you have to hire the talent. You have to hire the camera crew. You have to hire the stage. You have to have the animation guy and you need to cue it, you need to put it together and deliver it.

This type of personal organization of resources in intensional networks occurs even though the people and resources in the network are not necessarily part of the organizer's current work group, organization or area of specialization. These changes take place as work becomes more decentralized, communication media become faster and more efficient and workers must cope with fewer colleagues in their workplace and the subsequent decline in available in-house skills [2]. An intensional network relies on a person at its locus, a person who has a catalog of past contacts in various locations that they know will be of use for a task that they are interested in completing. "Active" collaborators in the network are those who are currently engaged in some sort of work activity with the locus of the network. "Inactive" collaborators are those who still have their skills and interests known by the locus but are not currently working with the locus. The locus of the network will activate sections of the network by organizing relevant collaborators into some sort of informal work arrangement of which they are the center and coordinator. The nodes in a person's network are constantly being updated and maintained as a person moves from task to task, activating and deactivating sections of their network as necessary.

An activated network may serve as a type of project team, but not one in the traditional sense. First, the network will not be active longer than is necessary to complete the task that required the activation of the network initially, whereas a project team may move on to new projects. Second, the membership of an activated network is much more fluid and decentralized than that of a project team, with all communication and activity being routed through the person who is the nexus of the network. Members of the network do not always need to be aware of each other, and often aren't. Third, traditional project teams are almost always located in the same physical location, and within the same organizational structure, both attributes that are atypical of how an intensional network operates.

Similar work has been done regarding "constellations" of interaction in workplace environments [3]. As in an intensional network, an individual forms the center of a constellation work group, and the membership of the group crosses team and organizational boundaries. The work is also very task-centric and the duration of the constellation's active periods is generally limited. One major difference between constellations and intensional networks is that constellations are also defined to include information resources. This type of personal organizational structure is similar to an "electronic village" [4] that refers to the personalized collection of contacts, resources and informal community perspectives that digitally connected users use as their informational filters and potential collaboration resources and partners. Workers not only have groups of human contacts they rely on to accomplish certain tasks, they also have favourite web sites, books and other content resources that they know are appropriate to use in certain contexts and form part of their networks.

Both Salvador and Nardi identify the importance of workers constantly maintaining and updating their intensional networks or constellations. Depending on their intended use, people are aware of the existence and usefulness of these groups to a greater or lesser degree. Intensional networks and constellations, often used by people to put together tangible project teams and resources, have a well-recognized existence. On the other hand, personal electronic villages can both be used explicitly, as when someone organizes resources in their village to accomplish a specific task, much like an intensional network, or implicitly, to serve a more passive, information filtering role. Workers in the Nardi study often take opportunities to meet and contact members of their intensional networks to find out what their current work status and skill sets are. Salvador similarly talks about workers keeping their constellations "in sync". This refers to workers understanding the work contexts of members of their networks and also evaluating the flow of information and tasks that exist within the network. For example, a node in a network may be a useful and reliable worker for various tasks, or a source of information that only begrudgingly co-operates.

2.2.3 Specific-task Teams (Knots)

A specific-task team, or knot, is a group in which a limited set of people is convened to accomplish some goal in a set period of time. This type of group is often a non-strict subset of a team trying to complete some task, but is also often composed of a random and ephemeral collection of people, brought specifically together because of their relationship around a particular task. The primary difference between a knot and a team is the short period of time that a knot exists for, and its singularity of purpose. Examples of such groups online include short-term online chat groups discussing a specific topic, online meetings that are using videoconferencing tools to discuss some type of task or a two person peer-to-peer file session transfer.

An example of a collaborative system where such groups are formed is I-Help [5]. In this peer-help system, students request help on a specific concept, and the system attempts to find a suitable helper who can help the person requesting help (the 'helpee') with their information request. Once the helper is found, the two students participate in an anonymous chat and whiteboard session where the helper discusses the concept with the helpee. This demonstrates a typical use of short-term groups: to facilitate a short, random-membership interaction that is very focused around a specific task. Formation of these types of specific-task groups is greatly aided by collaborative applications that allow the gathering and coalescence of a wide variety of physically dispersed people who are interested in the same task. Also interesting is that specific-task groups are often the "base case" or foundation of many other sorts of communities. In I-Help, the help session participants are drawn from a much larger virtual community of people who also participate in online discussion forums and project teams. This aspect of specific-task formation is also seen in more prominent examples such as online auction houses or peer-to-peer file exchanges, where a large and amorphous virtual community of potential users is necessarily reduced to a specific-task group when a transaction is consummated.

There is also related work done by [6] on "knots" and "knotworking" that examines groups like flight crews or operating-room teams that are brought together for a limited temporal duration to perform a very specific task.

2.2.4 Short-term Intensional Networks

Active intensional networks in general do not have a great degree of group cohesion. The group members often do not interact with each other, as the creator of the intensional network provides most of the communication and co-ordination within the group. Intensional networks are also very task focused, as the task is the only reason for their existence, and tend to be around for a fairly long period of time, as the tasks that they are required for tend to be rather involved. However, mobile collaborative technologies have made possible an interesting sub-class of intensional networks, which are still loosely coordinated and highly task focused, but exist only over a relatively short time span. These 'Short-term Intensional Networks' eliminate the need for a central organizational figure entirely, and are entirely task focused. Korteum et al. [20] discuss 'Wearable Communities', which are groups of mobile computer users whose onboard agents negotiate with each other to achieve an efficient collaborative allocation of the users' various tasks. For example, two agents whose owners pass each other in the hallway would exchange the current task lists of their respective users, as well as a list of tasks that they are able to do, and the costs, trust information and priority information associated with each task. The agents quickly initiate a negotiation sequence to determine if the one person can complete the other's tasks more efficiently. User A may agree to take User B's journal to be photocopied if User A is heading towards the office in exchange for User B doing repairs on User A's computer.

Rather than work elements being initiated by a single figure, this type of intensional network is completely distributed and task-focused. Short-term intensional networks rely on the physical proximity and intentions of users to complete tasks, rather than any previous business or social connection. The potential usefulness of such groups is large in a world of mobile networking and computing.

2.2.5 Virtual Communities

A virtual community is a many-to-many meeting and discussion area in some sort of online "place" [63]. Examples of virtual communities include web bulletin boards, Usenet newsgroups, online role-playing environments and durable online chat rooms. The membership of a virtual community is nebulous, the temporal duration of their existence is long-term and the goals of the group are generally diverse and shifting, even if the community has been formed to deal with a specific topic or domain. Membership in a virtual community is generally self-selected from an organization or the online community at large, and the communities are self-organized, due to their open and unregulated nature. Although their existence dates back to the first e-mail and newsgroup systems, virtual communities gained prominence in the work of the author Howard Rheingold, who wrote a book called The Virtual Community: Homesteading on the Electronic Frontier, which was largely based on his experiences in the WELL online community [7]. Research in the area has intensified in recent years as content-providing Internet companies seek insights into what factors will make their virtual communities successful [8]. Work in the field tends to take the form of long-term ethnographic studies, as researchers attempt to analyze the culture and 'rhythms' of online communities in order to understand their underlying dynamics. In [10] and [9] the authors examine the functioning of a variety of successful online communities in order to draw useful heuristics regarding the design of future virtual spaces. In [9] an early type of virtual community, the MUD (Multi-User Dungeon), is examined. MUDs are text-based applications based on the Dungeons and Dragons board game where users role-play characters online and interact in a series of virtually separated locales such as forest clearings and rooms in castles. Another closely studied virtual community is SeniorNet, an America Online hosted virtual community for senior citizens, that is used as a case study by Mynatt [10]. SeniorNet consists of a set of discussion boards and chat rooms dealing with senior-focused topics that range from discussion of World War II to supporting members who are dealing with cancer. Based on her observations, Mynatt developed a series of heuristic rules to be used in developing an effective and long-lasting virtual community:

1. Online spaces need a physical context or sense of place to function effectively as a social space. Boundaries are used to partition the large communication bandwidth inherent in virtual communities into cognitively manageable chunks. In MUDs, this is accomplished by dividing the virtual world into a multitude of virtually separated locations. A similar separation is accomplished in SeniorNet with the division of discussion groups by topic: a separation that is enforced by members of the various groups.

2. If a specific user community is being targeted, the online world should be isomorphic with the physical world(s) that the users inhabit. The MUD described in [9] was targeted at a specific research institution, so the rooms were divided according to the organization of the various research groups within the organization. The isomorphism allows users to transfer social conventions from one environment to the other, making behaviour more intuitive to participants and helping ease-of-use of the virtual system. This isomorphism will likely become less and less necessary as future generations of users become more and more accustomed to dealing with virtual environments.

3. The identity of participants should be easy to ascertain, map to the physical world and be consistent throughout the virtual community. Trust, and other social bonds, are an important part of establishing social cohesion and a sense of membership in a virtual community. In SeniorNet, participants' real identities and biographies are used to enable members to throw each other virtual birthday parties, support each other through illnesses and provide a shared history when members encounter each other in various virtual contexts.

4. Different virtual communities should support different interaction "rhythms" and styles. In the SeniorNet system, different discussion forums have different paces, styles and cultures. An asynchronous message group about living with cancer has a much different conversational tone and response speed than a synchronous chat group that serves as a forum for general social interaction. A variety of interaction styles allows many different sorts of users to find a niche in the system, which increases the community's diversity and viability. Designers of a virtual community should be attuned to the changing needs and demographics of the community. The community must be flexible and adaptable in order to accommodate changing technologies and interests.

5. New users should always be afforded an easy way to start using the various features of the virtual community. A virtual community has a constantly shifting membership and the community needs constant renewal in order to stay viable. To allow this, new participants must be able to become acclimatized as soon as possible. This criterion applies to both the means of interacting with the community, and how the interaction takes place. A new user must be able to use the interfaces that the community provides and adapt to the rhythms of interaction within the community.

Beyond discerning general principles of successful large-scale online interaction, work has also been done on more specific types of virtual communities, and more rigorous study done of the dynamics that exist within them. Jones [19] discusses the study of 'virtual publics', which are defined as "symbolically delineated computer-mediated spaces, whose existence is relatively transparent and open, that allow groups of individuals to attend and contribute to a similar set of computer-mediated interpersonal interactions." A virtual public is a more specialized form of a virtual community, as it requires open access to all online users and only denotes one specific online communication area. [19] develops a "user contributions/user population function" which measures the relationship between a population of a virtual community and the population's contribution to a virtual public. The function uses three separate criteria in its calculation. The first is "critical mass", discussed in [18] and [17], which is a community population boundary necessary to ensure sustainable interaction in a virtual community. Critical mass estimates of the number of participants necessary to keep a community active range from eight to fifty, depending on the nature of the community and the commitment of the users of the community. The second model is concerned with the phenomenon of "social loafing", discussed in [16] and [15], which is the tendency for individuals within a group to lessen their contributions in an inverse non-linear fashion with respect to the size of the group. The third model deals informally with "information overload", or more formally, human cognitive processing limits in a group situation [14] [13]. After a certain threshold of group communication activity in a given time, humans can no longer absorb all of the communication within the group. The user contributions/user population function takes all three models into account to provide a theoretical foundation for the relationship between user-population and user-contribution. User-contribution is minimal below the critical mass required in an online community, it increases regularly after the critical mass is obtained, and levels off at or near the cognitive processing limits of members of the group. Models and guidelines such as the ones discussed above are advertised as being applicable to all virtual communities, but most of the studies on virtual communities focus on simple, text-based communities or publics, which feature social interaction styles that are relatively analogous to the real world.

The future of virtual communities lies in already successful, fully interactive, graphical worlds, such as Ultima Online [8] in which members of the community construct graphical "avatars" and interact in worlds more or less resembling our physical one. Ultima Online is an extension of early MUDs, and allows participants to portray themselves as wizards, fighters, etc., which compete and interact in the same graphical virtual space. Many complex and fascinating social phenomena have emerged within the game, such as trade guilds, online marriages and organized groups of robbers and brigands. While extreme examples of gender and identity switching occur largely within the online gaming community, work done by Turkle[11] and others details the multitudes of virtual selves that are developed by most people who maintain an online presence. New models of online social interaction and dynamics such as Turkle's are needed to model such communities where people often use their online personas to conduct extreme role-playing, build artificial realities and experiment with gender and identity-manipulation.

2.2.6 Summary of Group Types

The previous sections have qualitatively described different sorts of groups in terms of their membership, task focus and the collaborative tools and techniques used to support their work. For all the group types examined, however, there are common properties they share that can be analytically examined and used to develop a taxonomy of group types that will help to clarify future discussion of group work and behaviour.

The properties common to all groups used for this classification are:

1. The duration of existence of a group.

2. The 'task focus' of a group.

3. The length and level of intensity of the cohesion/collaboration between group members.

Table 2.1 charts the values of each of these properties for the groups examined in this section:

Table 2.1: Taxonomy of Group Types

Type of Group	**Temporal Duration**	**Task Focus**	**Group Cohesion**
Teams	Medium-Long	Medium-High	Medium-High
Intensional Networks	Short-Medium	High	Low-Medium
Knots	Short	High	High
Virtual Communities	Long	Low	High
S.T. Intensional Networks	Short	High	Low

2.3 Team Performance and Evaluation Research

To generate the content for a group model, all of the factors that affect a group's performance must be analyzed and understood. In this section, the research surrounding the evaluation of team performance is examined.

2.3.1 The Difficulties of Measuring Team Performance

The difficulties involved in measuring group performance are well documented [29], and primarily revolve around two issues. The first is how to separate the measurement of group-level behaviour from the measurement of the individual-level behaviour of team members, or more formally, the 'unit of analysis' problem. By the act of working in teams, group-level factors such as intra-group communication, task division and conflict resolution become important, whereas they are not issues for a lone individual working on a task. However, a group's performance and functioning is dependent on the skills and personalities of the individuals comprising it. Which type of measurement factors better correlate with team performance, and similarly, which best describe a team? The division between the two levels is not always clear. Consider measuring the domain knowledge of a software engineering team. Should it be measured at the group level or the individual level? Do one excellent Java programmer and three poor ones in a group imply that a group's Java programming knowledge is excellent, average or poor? The level of analysis must be taken into account before a group performance analysis is conducted.

The second difficult factor involved in measuring group performance is whether to measure the outcome of team activity or the process of team activity. The outcome of team activity disregards the actual functioning of a group and focuses on how well they achieved the tasks set out for them. Outcome-based measures of group performance focus on the speed or quality of a team's performance while disregarding the actual elements of a team's functioning. The difficulty with outcome-based measures, especially when used in uncontrolled experiments or real-life settings, is the presence of mitigating factors that contribute to a team's success but are not a measure of the team's actual ability. Outcome-based measures do not tell us if a group was successful because of positive group dynamics and excellent task division or because they gave up on their assigned task and relied on the work of friends, for example.

Alternatively, process-based measures measure the functioning and internal activity of groups. Process-based measures primarily concern themselves with the "activities, strategies, responses and behaviours employed in task accomplishment" [30] such as the task distribution and coordination within a group or the amount of help and task backup provided by team members to each other. Group functioning is often measured by assigning evaluation questionnaires or quizzes to the members of a team and/or expert observers who have been monitoring the team's activity. While providing the possibility of more accurate and useful representations of group behaviour, process-based measures are more complicated to administer and a generally accepted formal theory that correlates elements of group behaviour and group effectiveness has not yet been developed. The team behaviours that lead to the most effective outcomes have also been found to widely vary in response to the type of task a team is engaged in. Bowers [31], for example, found that increased communication between the members of a flight crew, often considered a positive factor in group performance, actually decreased their effectiveness. However, some team behaviours and processes have been found to contribute positively to group success and task completion, regardless of the task the group is undertaking. These behaviours are discussed in the next section.

The quality of team process ratings can also be variable. Questionnaires and quizzes may suffer from a lack of rigor, relevance and reliability, and the scores of raters, both expert and team member, often do not correlate well [33]. For example, Hallman et. al [32] notes that the largest score discrepancies in his Team Development Survey occur in the "Leadership and Empowerment" section, with team leaders consistently having a much more positive view of their team's performance in that category than do their subordinates! In computer-mediated collaborative environments, such as Soller's OMT tutor [38], the utterances of the team members are saved and analyzed to determine the effectiveness of team performance, but such analysis is only possible in closely monitored environments with limited task focus.

The difficulties mentioned in Hoppe's parameterization paper [28] regarding the establishment of cognitive models for group functioning are a combination of the process vs. outcome and individual vs. group issues. It is difficult to effectively measure the varied aspects of group functioning, and it is also difficult to separate the contribution of the individual from the contribution of the group when looking at the group's achievements. Generally, if effective process-based measures can be found, they are more accurate in measuring group performance and activity than outcome-based measures. However, the complexity of group interaction and the huge variety of different group types have ensured the scarcity of general and useful process-based measures. Despite the difficulties inherent in measuring team performance, some successful work has been done in defining the nature of team activity and the means of measuring it. An overview of this work is given in the next several sections.

2.3.2 The Seven Elements of Teamwork

Process-based measures of teamwork have many advantages over outcome-based measures, but before team processes can be measured, they have to be understood in terms of how they affect group behaviour. One of the best team process taxonomies is found in Dickinson et. al [34] where "seven key components of teamwork" that have a major effect on group behaviour and functioning are identified from a survey of the team performance literature. They are:

1. **Communication.** Communication is the foundation for all of the other key team skills. However, more communication does not necessarily entail better group performance. Effective groups will communicate only when necessary, so that extraneous information flow does not slow down their operation. Effective groups will also deliver appropriate information to each other at appropriate times during completion of the group task, ensuring efficient completion of the

task.

2. **Feedback.** Feedback behavior refers to a team member's ability to both give and receive feedback relating to their participation in a team. Feedback, to be a positive factor, must be helpful, constructive and timely. Team members should be able to incorporate feedback suggestions into their behaviour. Good feedback performance in a team implies good task and domain knowledge in the team, as feedback is often task-oriented.

3. **Leadership.** Leadership tasks in a group include providing direction, support, structure and the clarification of team goals for members of the group. Leadership behaviour is often centralized with one member of the group, but this is not always the case and good leadership behaviour should also be distributed effectively within the group. Typical leadership tasks include the resolution of conflict and proving a behavioural example for the other team members to follow.

4. **Team orientation.** Team orientation refers to the attitude that team members have towards the group's task, the group's leader, the ability of the group to perform their assigned task and the abilities of the other members of the group. Groups with good team orientation will have a high level of group cohesiveness and a shared sense of purpose.

5. **Backup.** Backup skills allow team members to assume the responsibilities of other team members. If possible, team members should have overlapping skill sets so that the absence of one team member does not render the team's task impossible to achieve.

6. **Monitoring.** Monitoring behaviour is the awareness team members have of each other's group behaviour and task performance. Effective teams generally feature high levels of awareness between the team members. Monitoring behaviour is the foundation to the backup and feedback skills.

7. **Coordination.** Coordination, much like communication, is a foundation skill for teams. Coordination refers to the ability of a team to allocate resources effectively to accomplish team goals and also to perform the other six teamwork tasks effectively and efficiently.

2.3.3 Shared Mental Models

Shared mental models are thought to be important factors in determining a team's success. They are a reflection of the 'on-the-same-page-ness', or cohesion, of a team. Shared mental models refer to the similarity between mental representations of teamwork and task knowledge held by the different members of a team. Shared mental models are an elegant tool for team research because they map effectively from individual-level attributes to a team attribute. The shared models can reflect any part of team functioning and behaviour, from goal achievement to team communication patterns and work roles. A team containing members with similar mental models of team functioning will likely be more effective, especially at problem-solving tasks [35]. Having similar mental models allows team members to quickly and accurately communicate problem descriptions, problem solutions, and important task knowledge because of shared conceptual understanding and a common task language between the team members.

There are several proposed ways to measure the different types of shared mental models that teams possess. Rentsch et. al [36]advocate measuring both shared process team schemas and task schemas. A shared process schema measures the similarity between team member's categorizations of the importance and inter-relation of teamwork-related concepts. A shared task schema measures the similarity between different team member's ordering of the sub-tasks that the group must complete to achieve its overall task goal, with the antecedent conditions for each sub-task and the assignment of particular group members to tasks included. Goldsmith et. al [37] describes a similar methodology where the process and task concepts involved in a team task are represented as nodes in a cyclic tree, with the weighted edges representing the similarity scores between the concepts as given by each individual team member. The graphs for each team member can then be compared to derive a similarity score for the mental models held by the team members.

2.4 Group Performance Factors

Hoppe's [28]work describes how to form ('parameterize') groups with members competent to complete the task given to the group. Task competency, however, is just one of the factors that determines group performance. This section presents a survey of the factors known to determine group performance, from appropriate team roles to social intelligence to relevant domain knowledge.

2.4.1 Team Roles

A team role is identified as "a pattern of behaviour characteristic of the way in which one team member interacts with another so as to facilitate the progress of the team as a whole" [40]. Several attempts have been made [51] [43] to identify various team roles and the combinations of roles that lead to effective team functioning, with the most widely recognized and used discussed in Belbin [40]. Belbin and his team measured managers for personality and critical thinking skills and then measured the performance of groups of managers in a computer-based executive management game while taking extensive notes on group processes [44]. Correlating the group process notes with the tests and the management game outcomes, Belbin identified eight separate team roles and developed a theory of which combination of these roles would lead to successful teams. The eight roles identified were:

1. **The Resource-investigator (RI)**. A resource-investigator is extroverted, enthusiastic and communicative. A good negotiator, the resource-investigator has problems sustaining interest in the team project once the initial stages have passed.

2. **The Plant (PL)**. A plant is creative, unorthodox and imaginative. The plant is effective at solving difficult problems facing the team but has trouble communicating and managing ordinary people.

3. **The Coordinator (CO)**. The coordinator is mature, confident and trusting. The coordinator is a good chairperson and helps clarify goals and promote effective decision making. The coordinator is not necessarily the smartest or most creative person in a group, and may sometimes be seen as shirking their assignments.

4. **The Shaper (SH)**. The shaper is dynamic, outgoing, highly strung. The shaper challenges the team, applies pressure and is a good problem solver. The shaper is quick to anger and easy to provoke.

5. **The Monitor-Evaluator (ME).** The monitor-evaluator is sober, strategic and discerning who sees all options and judges situations accurately. The ME lacks the drive and ability to inspire others to action.

6. **The Team Worker (TW).** The team worker is social, mild, perceptive and accommodating of others. The team worker listens well, builds consensus and averts friction, but is indecisive in clutch situations.

7. **Implementer (IM)**. The implementer is disciplined, reliable, conservative and efficient. The implementer turns ideas into actions but is often inflexible and slow to respond to new possibilities and ideas.

8. **Completer-finisher (CF)**. The completer-finisher is painstaking, conscientious and anxious. The CF searches out errors and omissions and delivers on time but is prone to unduly worry and reluctant to delegate.

   Belbin introduced a modified version of his team roles in [41], which included a new role, the Specialist (SP):

9. **The Specialist (SP)**. The specialist is single-minded, self-starting and dedicated. The specialist provides rare knowledge and skills but only contributes

to a small part of task completion and dwells on technicalities.

However, little work has been done on how this role can be measured and integrated with the other roles, and much of Belbin's new work is the proprietary property of his consulting company, Interplace Associates [33] (now, Belbin U.K.), so it will generally not be taken it into account in this survey. Along with the eight team roles, Belbin's research identified five principles for building a high-performance team: 1) Each group member must perform a functional role and a team role. 2) A balance in team roles and functional roles is needed, relative to the group's goals. 3) Team effectiveness is proportional to the team's ability to adapt to the strengths and weaknesses of the group. 4) Personality and mental abilities allow members to effectively assume some roles and not others. 5) The teamwork necessary for groups to use their technical abilities in the most optimal manner is dependent on a good mix of team roles in the group [40]. Essentially, Belbin contends that each team role must be present in a group in order for that group to be maximally effective although he also believes that team members are capable of playing two roles (primary and secondary) at the same time, if necessary. Referring to the fourth factor above, Belbin believes that people are naturally predetermined to effectively assume certain roles and not others, which means that a set of arbitrary managers cannot be placed in a group and told to assume roles, but that their personality and intelligence will determine what roles they can fill. To help determine which team roles people were best suited for Belbin developed the Belbin Team-Role Self-Perception Inventory (BTRSPI) which attempts to determine workers' preferred team roles by giving them self-evaluation tests [40]. The BTRSPI is used extensively in corporate situations in order to form highly compatible management teams [45]. The BTRSPI consists of a series of behavioural statements that relate to seven hypothetical work situations. The test-taker must determine how closely each behavioural statement describes their own preferred action in a situation, and distribute ten "points" among the statements accordingly. For example, the first question in the BTRSPI is:

WHEN INVOLVED IN A PROJECT WITH OTHER PEOPLE:

1. I can be relied upon to see that work that needs to be done is organized.

2. I pick up slips and omissions that others fail to notice.

3. I react strongly when meetings look like losing track of the main objective.

4. I produce original suggestions.

5. I analyze other people's ideas objectively, for both merits and failings.

6. I am keen to find out the latest ideas and developments.

7. I have an aptitude for organizing people.

8. I am always ready to support good suggestions that help to resolve a problem.

Unfortunately, the BTRSPI has been criticized in recent years for lacking a strong empirical or theoretical basis. Furham [45] criticizes the BTRSPI for several reasons. The first is that the BTRSPI is ipsative, rather than normative, e.g., one individual's test scores are not measurable against another individual's test scores, but only to themselves, as a taker of the test is asked to compare his relative preferences for certain actions. This can be problematic in team formation, because a worker who has a higher test score in a particular team role than another worker is not necessarily better or more suited to that role, just that the first worker has a higher relative preference for that role. The second criticism revolves around the factor structure of the BTRSPI. Belbin [40] identified four factors that would provide the separation between team roles: team leaders, intellectuals, negotiators and manager-workers. For example, Belbin classifies the Plant and Shaper roles as intellectual, so that managers with high scores on the BTRSPI questions designed to measure intellect would be classified in one of those two roles. However, factor analysis conducted in [45] on a variety of experiments did not find a four-factor structure, but rather two or three factors depending on the experiment. This casts doubt on the BTRSPI's ability to accurately determine scores on the eight separate team roles.

The final criticism lies in the BTRSPI's lack of internal consistency. The responses to the questions that attempt to measure the same concept or construct, such as the shaper role, should have a high correlation coefficient to show that each question truly is a measure of the same construct. However, the BTRSPI scores poorly on the internal consistency tests done in [45]. The lack of consistency and validity demonstrated by the BTRSPI seems to make it unusable for a serious research study, although its lack of efficacy does not mean that Belbin's proposed team roles are inaccurate or invalid, just that a better measure for determining team roles must be found.

Fortunately, Dulewicz [44] presents two personality quizzes that can act as alternative measures of team roles. The two quizzes are Form A of the 16PF [46], a general measure of personality which was actually used by Belbin in his original team roles study, and the newer OPQ test, which measures behaviour in workplace situations [47]. Both the 16PF and OPQ tests are normative, as opposed to ipsative, and psychometrically validated unlike the BTRSPI. This answers two of the criticisms contained in [45]. Dulewicz [44] presents the findings of a study in which he applied both the 16PF and the OPQ to a group of managers and compared the results. He found that the team role scores given by both tests were significantly correlated, with the exception of the ME role. This gives support to the construct validity of the team role theory, which indicates that actually existing structures are being measured. Dulewicz also performed a factor analysis of the two tests, finding that the 16PF produced four factors, which could be covered by six team roles, and that the OPQ produced three factors that could be covered by four team roles. The IM, CF and ME roles load heavily on the first OPQ factor and also have high correlation coefficients, so the roles can be considered virtually equivalent. The second factor groups together the CO and RI roles, while the third factor groups together the PL and SH roles, and contrasts them with the TW role.

Alternatively, further studies conducted by [52] follow up on Belbin's suggestion that "most competent managers should to be able to function well both in a primary and secondary team role" [40]. Fisher [52], using the 16PF questionnaire to determine team roles, found that there were two major groupings of team roles, one relating to relationship-related group functioning and the other consisting of task-related group responsibilities. The relationship team roles include the coordinator, team worker, resource-investigator and the implementer while the task team roles are plant, monitor-evaluator, completer-finisher and shaper. The study also found that workers usually have a natural secondary role of the same type as their primary role, and that the most effective work pairs have one member of each type.

2.4.2 General Measures of Social Intelligence and Teamwork Skills

While team role theories such as Belbin's are considered the most useful tools relating to the precise construction of effective teams, there are a multitude of psychological theories and assessment packages that measure a person's general social and teamwork aptitudes. Usually, these assessments are used in an evaluative function for members of an already existing team, or as a generalized social aptitude test for potential employees at a company [48]. There are many tests that focus on measuring personality traits, such as the famous Myers-Briggs test, but not as many tests that translate those individual characteristics to a team context, such as the BTRSPI. Some other examples of popular personality tests that are especially focused on teamwork and social skills include:

1 . **FIRO-B.** This test produces six scores that measure the amount in which the test-taker expresses and desires inclusion, control and affection from other people. The test is generally used for self-reflection and the possible modification of extreme behaviours identified on either end of any of the six scales.

2. **Interpersonal Communication Inventory.** This test analyzes the communi-

cation techniques of an individual based on five interpersonal personality traits: self-knowledge, listening ability, the ability to clearly express ideas, emotional coping and the ability to confide in others. It is often used in counseling, teaching and human resource situations.

3 . **Thomas-Kilmann Conflict Mode Instrument.** This test measures an individual's relative use of five different conflict resolution techniques and is usually used as a test for managers or other potential organization leaders. The test presents users with hypothetical conflicts and gives them alternatives that correspond to one of five conflict-resolution approaches: competing, accommodating, avoiding, compromising and collaborating [48].

2.4.3 Domain Knowledge

Research efforts concerning the attributes required by group members in order for a group to be successful often concentrate on teamwork skills, usually because the individuals in the pool drawn on for the experiments are assumed to already possess the necessary knowledge for task completion. However, in order better predict team performance, a more detailed understanding of the knowledge and skills each individual brings to the group must be obtained. A group, no matter how congenial, cannot complete its assigned tasks if its members do not have the skills and knowledge to do the job. Revisiting Belbin's first two principles of effective groups, gives further insight into how knowledge must be distributed amongst group members: 1) Each group member must perform a functional role and a team role. 2) A balance in team roles and functional roles is needed, relative to the groups' goals [40]. A group where every single member has the same limited skill set, or prefers to fill the same functional role will be less effective than the group where the members have a variety of skills and role preferences that are related to the task. However, ensuring domain skill heterogeneity is not as large a challenge as is developing team role heterogeneity, because new skills and knowledge can generally be learned easier than unfamiliar team roles can be performed, due to the intrinsic psychological nature of team role preference.

2.4.4 The Effect of Task Type on Group Performance

The types of tasks performed by a group have an enormous effect in determining what attributes, behaviours and skills a team will require in order to be successful. The skills and teamwork behaviours required by a group of astronauts is much different from the skills and behaviours needed from workers in a secretarial pool or a pit crew in an auto racing team. Therefore, to build a group model, it must be considered how the task being performed defines the teamwork and domain skills needed to complete the task and the relative importance of each type of skill. A useful taxonomy of task types is found in Telsuk et. al [49], who categorizes four separate types of tasks, based on how the flow of work goes through groups that are optimally configured to perform the four task types. Telsuk also determines which of work processes (domain knowledge and skills) and group behaviours (teamwork skills) have the most impact on the group's performance for each different task type. The four different work flow arrangements, which correspond to the four different task types, are:

1 . **Pooled interdependence.** This type of functioning occurs in groups that have performance that can be measured in a linear fashion by adding the contributions of each of the individual members. Coordination and cooperation skills are not essential in these types of groups, as interaction is not a critical factor in their success. Examples of groups in this category are typing pools and computer support help desks. Tasks for the group are placed in some sort of queue and each member of the group works independently on a task until it is completed. For these types of teams, the skill level and domain knowledge of the individuals composing the team are the critical factors that will determine the group's success. If the individuals in the team are competent in the skills required to complete the tasks required of them, the team will function well.

2. **Sequential interdependence.** A sequentially interdependent work group is

one that performs in an assembly line manner. A job enters such a group, and is worked on by an individual team member until his or her portion is complete. When complete, responsibility for the task is then moved to the next member of the group for further work until each member has made their contribution and the entire task is completed. Each member of the group makes a unique contribution to task completion. The movement of work through the group is not symmetrical or random, it almost always moves forward and in the same flow order. In these types of work groups, group interaction and teamwork skills gain importance because of the interactions between the various neighbouring members of the team. These interactions require coordination, communication and monitoring skills, but not the whole spectrum of group behaviours as described by [34], and group performance is still mostly dependent on the individual task competencies of the members of the group.

3. **Reciprocal interdependence.** Teams that feature this type of work flow have members that have specialized work skills and defined roles within the team, but also have a bi-directional movement of work and a large degree of collaboration and coordination on the higher-level goals of the task. A standard software engineering team provides a good example of this type of group. Usually, each member of the group will have their own area of responsibility, such as database design or interface implementation, but all of the team members are working towards a common goal of project completion. Teammates are often dependent on the work of each other as a necessary prerequisite for completion of their own tasks: for example, when an interface designer needs the application layer to provide proper data for the interface to display. Feedback and backup teamwork skills become more important as team members are often required to provide assistance or detailed evaluations to each other. Interactions between team members become more dynamic and complex, as they may involve many different aspects of team and task goals. Teamwork skills become more important in this type of group, as task success is not only

dependent on individual competencies, but on the efficient interaction of group members.

4. **Intensive.** Intensive work groups are similar to reciprocally independent work groups with the added complication that they must cope with a compressed time scale for task completion and real-time consequences of their behaviour. Examples of such groups include surgical teams, astronauts and cockpit crews. Team roles are often interchangeable, and task completion relies on the group having a very high degree of successful teamwork behaviours. For intensive work groups, task knowledge and good teamwork behaviours both must be very high for the team to succeed.

2.5 Technical Methods of User Modelling

User models represent a software user's goals, plans, skills, attitudes, emotions, beliefs and other facts about their existence. User models are the medium by which computer programs explicitly "know" and "understand" their users, knowledge that is used for program adaptation, teaching, marketing or a multitude of other purposes [21]. User models generally have a template of categories regarding the information about the user that they wish to know, and a set of relationships that describe how the categories are related. These templates are then instantiated with values reflecting the facts about particular users and used for whatever purpose the program desires. The knowledge in the user models can be gathered either explicitly by observing user behaviour using questionnaires and tests, or implicitly by using the model's structure and beliefs to infer information. A user model must be constantly refined and updated as the individual changes, and as different inferences are propagated throughout the model.

This section will examine a variety of user modelling techniques and their applicability to the modelling of groups and group behaviour.

2.5.1 Overlay, Perturbation and Buggy Models

Overlay models are one of the traditional ways to construct a user model, especially a student model in an intelligent tutoring system [21]. As most adaptive systems restrict themselves to a limited domain of knowledge, an expert model is constructed for the system that consists of the facts and procedures an expert would know in relation to that domain. Models of other users are then constructed as overlays on the expert model, by observing the behaviour of the users in some environment, and noting where their behaviour differs from the expert. This type of user modelling is useful in situations where simple inferences are needed about what skills and knowledge should be taught to the user.

Perturbation models are a more sophisticated extension of overlay models, in that they do not assume a user's knowledge is only a subset of an expert's knowledge. They also include some common domain errors and misconceptions not held by an expert that can augment a user's overlay model if the user's behaviour indicates that he holds those erroneous beliefs. However, it is impossible to fully model the nearly infinite number of possible misconceptions a user could hold about a certain task. This had lead to discussion that the problem of student modelling is intractable in its general form [22]. However, the same errors in a domain are often repeated over and over again by different users, allowing the "bug libraries" kept by perturbation models be effective in tracing user actions.

2.5.2 Constraint-based Models

Constraint logic programming is a popular technique in artificial intelligence for problem solving and optimization techniques, especially in complex domains. A constraint takes the form $c_0 \wedge,...,\wedge c_n$ where $n \geq 0$ and c_0 and c_n are called *primitive constraints*. Primitive constraints are defined according to the constraint domain in which the constraints are being specified. For example, in the arithmetic domain primitive constraints take the form of arithmetic equations such as $X + Y \geq 4$ and $X < 2$, with the arithmetic domain providing the relations, values and functions in the constraint. Other common domains used in constraint logic programming are Boolean logic and tree constraints. Constraint logic programming tools attempt to solve these constraints by finding sets of values that satisfy all of the primitive constraints. For example, possible solutions to the series of constraints above include [$X = 1$, $Y = 3$] and [$X = 0$ and $Y = 4$]. A series of constraints can evaluate to true, untrue, or unknown for a given *valuation*, or assignment of values to the variables in the constraints. Constraints are solved by *constraint solvers*, which are algorithms that use a variety of techniques to determine both the *satisfaction problem* (does a constraint have a solution) and the *solution problem* (what are correct valuations of the constraint). Some constraint domains have the property that the satisfiability of some constraints is inherently unknowable.

Constraints are effective in modelling and solving real world situations in which a series of equations can describe the interaction between the various components involved in the situation. Some examples of real-world domains where constraint logic programming systems are used include circuit design and power-grid design optimization. In addition to providing solutions to problems, a constraint logic system can also provide optimal solutions to a series of constraints, when given a heuristic that describes the optimality rule. The heuristic takes solutions in the constraint domain, and maps them into real number values. This heuristic function $\int$ is called the *objective function* and solutions that have a higher value when put in the objective function are preferred over those valuations with a lower value. For example, an objective function that minimizes the constraints described above, [$X + Y \geq 4$ and $X < 2$], is $1 / (Y - X)$, that minimizes the difference between X and Y, and gives a valuation of [$X = 1$, $Y = 3$].

An example application of the use of constraints in user modelling is the SQL tutor described in [22]. In this type of user model, constraints are used to represent correct solutions to problems in a particular domain. Students are observed solving problems in an ITS and their student models become collections of the constraints that are and aren't violated, which correspond to the students' incorrect and correct solutions, respectively. Mitrovic's SQL tutor uses a special form of constraints, developed in [23], called *state constraints* that are ordered pairs $<C_R, C_S>$ where C_R is the relevance condition and C_S is the satisfaction condition. The relevance condition is composed of constraints that identify in which problems the state constraint is relevant, and the satisfaction condition identifies the constraints that must be satisfied in those problems for a correct or optimal solution. In other words, if in a problem solution C_R holds, its C_S pair must be satisfied as well.

In Mitrovic's tutor, the student interactively attempts SQL query formation, and is informed of constraint violations as they submit their solutions to the tutor. Mitrovic's hypothesis, following [23], is that deep understanding of user knowledge is neither computationally feasible nor necessary. Regardless of the user's mental models that led him to commit the error, the remedy to his error should be the same, e.g. an instructional session explaining proper SQL syntax or which database table the user should be gathering data from.

2.5.3 Bayesian Network Models

Bayesian belief networks (BBN's) are used to deal with situations that require reasoning with uncertain information [24]. A BBN reflects the fact that our understanding of the world is often imperfect, whether through a lack of awareness of factors that affect a situation or an inability to gather information about those factors [25]. However, rational decisions and inferences must be made about situations, such as modelling a user's beliefs and knowledge, in which certainty is lacking, and these decisions must become statistical and probabilistic in nature. A BBN allows for reasoning about a situation by assigning probabilities of truth to beliefs and inferences regarding that situation.

BBN's were developed to deal with the complexity inherent in working with standard probabilistic reasoning methods. A domain is modeled with a collection of random variables representing entities in the problem domain that can take on a collection of values, with each value assignment having a certain probability of occurring. When each random variable is assigned a value, it is called an atomic event, which reflects a specification or instantiation of the domain, or an observation of the values of the variables in the real world. The probability of each atomic event occurring is specified by the joint probability distribution $\mathcal{P}(\mathrm{X}_0,...,\mathrm{X}_n)$ where $\mathrm{X}_0,...,\mathrm{X}_n$ are the random variables in the domain. The joint probability distribution can be represented as a n-by-n-dimensional table with each entry in the table giving the probability of the conjunction of all of the variable values occurring in an atomic event. The joint probability distribution represents the sum of all possible states of affairs in the world regarding these two variables, so the addition of the probabilities of all the possible atomic events in this domain must be equal to one (i.e. one atomic event is true).

However, working with the joint probability distribution quickly becomes cumbersome. The number of entries in the table increases exponentially at a rate of a^n where a is the number of possible values that each variable can take on and n is the number of variables in the domain. For most interesting problems there will be many variables, which increases the computing power required to deal with the problem, and raises the practical issue of defining values for each of the a^n possible atomic events.

BBN's provide a means of compressing the joint probability distribution into a usable form by using conditional probabilities. Conditional probabilities express the likelihood of a random variable assuming a value given the observed values of other random variables. The probability of Boolean variable A being true given the fact that Boolean variable B is true can be computed via the equation: $P(A|B) = P(A{*}B)/P(B), (P(B) > 0)$. However, the conditional probability of a variable value can be determined by Bayes's rule: $P(B|A) = P(A|B)P(B)/P(A)$, which defines a conditional probability by another conditional probability. Combining Bayes's rule with the conditional independence of variables allows us to form BBN's. Conditional independence between variables A and B given evidence C is expressed formally as $P(A|B \wedge C) = P(A|C)$. That is, once we know the value of C, knowing the value of B gives us no more information about the value of A. This implies that the variable C is a direct cause of variable A, and variable B only has an indirect effect on A.

Now, instead of having to specify large conjunctions of atomic events to find the probability of a variable having a certain value, we only need to know the conditional probabilities given the values of the variables that have a direct effect on it.

A BBN is a directed graph consisting of nodes representing the various probabilistic variables in the domain connected by directed arcs that represent the direct influence (whether causal or correlative) by the parent variable on the child variable. BBN's are a graphical representation of collections of conditional probabilities that relate the probability of the values for a child variable given values for the parent variables. In a causal Bayesian network, the parent(s) can be thought of as the causes of the child, although the relationship gets murkier when non-causal variables are modeled. Bayesian networks allow inferences in two directions: up and down. Making upward inferences from a child to its parents allows you to probabilistically determine (diagnose) the causes of an event you have observed, while making downwards inferences allows you to make predictions about the effects certain events will have.

One of these tables is kept for each parent/child relationship, and the conditional probabilities contained within them can be used to compute any conditional probability given any set of evidence within the domain.

Observation plays a key part in determining and updating the probabilistic relationship between the nodes in the network. When certain values of parent nodes are observed at the same time as a value for a child node, the conditional probability of that combination is increased, while the conditional probability of the other combinations is decreased. The relationships in a Bayesian network must be determined empirically, whether from existing models or experiments conducted explicitly for the purpose of knowledge representation and reasoning. BBN's are especially useful for user modelling, an area in which the models of user behaviour or belief are often incomplete, and where the evidence used to fill the models is often sparse.

Bayesian user models have become very popular in the user modelling community for the reasons stated above. Some of the noted examples include the Lumiére/Office Assistant [26] system and the NewsDude [27] system. NewsDude is a web-based news system that recommends stories based on a profile of the user's interests. The long-term profile of the user's interests is kept in a Bayesian network that calculates whether or not a story will be interesting to a user given the words in the story. The words in the news stories are kept as variables in the Bayesian network and given causal links to the "interesting" variable in the network. The probability values assigned to causal links are dependent on the user's previous story selection history. The Lumiére system was developed as the precursor to the Microsoft Office Assistant and used a Bayesian model to predict the goals and needs of users as they used application software. Some of the variables in its network were: 'Assistance History', 'Competency Profile', 'Context', 'User's Goals', 'User's Acute Needs' and many more. By observing the user's behaviour while using the software, the variables in the network were instantiated with values and the Lumiére system would decided whether or not to provide assistance to the user regarding their current task.

2.5.4 Fuzzy Logic Models

Fuzzy logic-based user models are often used as a less-precise but more practical approach to dealing with uncertainty than that provided by Bayesian networks [24]. Fuzzy logic (FL) systems discretely model the degree of set membership of an object, rather than an absolute value. As discussed previously about users in an adaptive system, it is hard to discern what their knowledge and beliefs in a certain domain are, especially when the inferences must be made from a user's behaviour, and not a direct questionnaire or test. Rather than relying on generally hard-to-acquire precise probabilistic relationships between variables that a Bayesian system relies on, fuzzy logic systems rely on discrete levels of set membership to evaluate the value of variables under consideration. A FL system first defines an arbitrary set of possible degrees of membership in a class such as: [False, Unlikely, Likely, True]. This scale of set membership is then applied to the possible members of the sets that are defined in a system.

For example, in a FL system that models a group of people's feelings towards pink shirts, the system can be initialized to have three domain sets: [Loves Pink Shirts], [Neutral on Pink Shirts], and [Loathes Pink Shirts]. A default belief tuple that matches up each domain set with a set membership operator must now be defined. Initially, all users will be believed by the system to have the same attitude towards pink shirts: (<’Likely’, ’Neutral on Pink Shirts’>, <’Unlikely’, ’Loves Pink Shirts’>, <’Unlikely’, ’Loathe Pink Shirts’>). Then, just like Bayesian networks, the fuzzy logic user model relies on observations, and mappings of observations to changes in degree of set membership to update its knowledge.

In an example system, the knowledge of a user’s beliefs about pink shirts is used to make predictions about whether or not they own a pink shirt. To do this, a prediction table needs to be defined that maps the set membership of a user to the different outcomes we can predict, in this case, the ownership of a pink shirt. Unlike in a Bayesian network, where the prediction and updating mechanisms are joined, a separate update table that maps the specific observations of a user to changes in the belief in the user’s membership in our various sets must be generated.

Here are the prediction and update tables:

Table 2.2: Example Fuzzy Logic Prediction Table

	Owns Pinks Shirt	**Does Not Own Pink Shirt**
Loves	True	False
Neutral	Likely	Unlikely
Loathes	False	True

Table 2.3: Example Fuzzy Logic Update Table

	Owns Pinks Shirt	**Does Not Own Pink Shirt**
Loves	+1	Impossible
Neutral	0	-1
Loathes	Impossible	0

To take user Bob as an example, the initial knowledge of his feelings towards pink shirts, expressed as a degree of membership in each of our classes, is as follows: (<’Likely’, ’Neutral on Pink Shirts’>, <’Unlikely’, ’Loves Pink Shirts’>, <’Unlikely’, ’Loathe Pink Shirts’>). So, after the initialization of unlikely, likely, and unlikely to Bob's love, neutral and loathing feelings towards pink shirts respectively, the prediction is made that Bob is likely to own a pink shirt, before any observations are made of Bob's behaviour. After observing that Bob owns a pink shirt, the belief of Bob's membership in the various sets is updated by following the set membership updating rules in the first column of the update table, where the number in the table is the discrete number of degrees we move our belief. The updated belief is now: (<'Likely', 'Neutral on Pink Shirts'>, <'Likely', 'Loves Pink Shirts'>, <'False', 'Loathe Pink Shirts'>).

Notice that because of the observation, it is now known that Bob's membership in the set of people who Loathe Pink Shirts is impossible. The values of True/False are absolute in fuzzy logic systems, as they are in standard logic. Once it has been established that membership in a set is true or false, further observations can not change its value. If it is observed that Bob owns another pink shirt, our belief in his set membership will now be: (<'False', 'Neutral on Pink Shirts'>, <'True', 'Loves Pink Shirts'>, <'False', 'Loathe Pink Shirts'>). Once the truth of Bob's membership in one of the sets is established, he is excluded from the other sets.

2.6 Group Modelling

While the modelling of individuals is a well-studied problem, the modelling of group-level attributes has not been researched to the same extent. The research that has been done tends to split into two different camps. The first type of approach focuses on taking the user models of the individuals comprising a group and translating them into a group context. The second style of group modelling generally ignores individual attributes and just measures group process behaviour such as conversational acts or actions in a collaborative environment. The following sections examine each approach in detail.

2.6.1 Generating Team Models from Individual Models

One of the seminal papers in the field of group modelling is Hoppe's [28] paper, "The Use of Multiple Student Modelling to Parameterize Group Learning". Hoppe acknowledges that the analytical monitoring and evaluation of group work, and in particular group learning, is hard to do and that the mappings from individual models to group models are not linear and straightforward (many of the same issues are discussed in the review of the team performance literature in 2.3). Therefore, Hoppe discusses the need to narrow the scope of group modelling projects by finding a series of partial and pragmatic solutions that improve group learning performance in ways that don't rely on a complete cognitive model of group performance and learning.

The partial solution that Hoppe proposes is the "parameterization" of group work through the use of existing individual models to generally define the "initial conditions" of a group. By using the convergence of the group members' individual user models as a guide to the performance of a group, Hoppe hypothesizes that relatively accurate predictions can be made about the various competencies of that group. Hoppe develops three crucial assumptions that guide how the parameterization will work in group learning situations such as trying to find a compatible match for a peer help session or in developing a suitable problem for a cooperative learning session. The three assumptions are:

1. **The Complementary Criterion.** Interaction between a student with good knowledge of a particular topic and a student with low knowledge of the same topic should result in the improvement of the second student's competence with the topic if they are working on a problem together that requires knowledge of that topic.

2. **The Competitiveness Criterion.** Two students with similar knowledge on topics related to a particular problem will stimulate each other to learn those topics better while working on the problem because of competitive pressure.

3. The Problem-Selection Criterion. A learning group should be given problems that cannot be solved by an individual member of the group, but that require the cooperation and knowledge of all of the group members.

The parameterization criteria and rules are formally defined through a series of predicates that hold regardless of the method by which the individual user models are constructed, in order to keep the parameterization process generalizable to all methods of user modelling. Hoppe's formalization is as follows:

$\mathcal{K}$ is the set of knowledge elements in the problem domain.

$\mathcal{T}$ are topics that are subsets of the knowledge contained in $\mathcal{K}$.

$\mathcal{P}$ is a problem that needs knowledge of one-to-many $\mathcal{T}$'s to solve.

knows$(\mathcal{S}, \mathcal{T})$ is true if student $\mathcal{S}$ knows topic $\mathcal{T}$.

has_difficulties$(\mathcal{S}, \mathcal{T})$ is true if student $\mathcal{S}$ has trouble with topic $\mathcal{T}$.

$\mathcal{S}, \mathcal{SA}, \mathcal{SB}$ are students being modeled by the system.

Different user models will provide different logical equivalences for these predicates. An overlay model system provides the following definitions:

$\mathcal{SM}(\mathcal{S})$, is the student model of student $\mathcal{S}$ which is a subset of $\mathcal{K}$ and leads to:

knows$(\mathcal{S}, \mathcal{T}) \leftrightarrow \mathcal{T} \subseteq \mathcal{SM}(\mathcal{S})$ and

has_difficulties$(\mathcal{S}, \mathcal{T}) \leftrightarrow \mathcal{T} \nsubseteq \mathcal{SM}(\mathcal{S})$ and

solvable$(\mathcal{P}, \mathcal{S}) \leftrightarrow$ knows$(\mathcal{S}, \mathcal{T})$ for each $\mathcal{T}$ required to solve $\mathcal{P}$

Complementarity Criterion

can_help$(\mathcal{SA}, \mathcal{SB}, \mathcal{T}) \rightarrow$ knows$(\mathcal{SA}, \mathcal{T})$ and has_difficulties$(\mathcal{SB}, \mathcal{T})$

Competitiveness Criterion competes$(\mathcal{SA}, \mathcal{SB}) \rightarrow$ for t in $\mathcal{T}$: knows($\mathcal{SA}$,t), knows($\mathcal{SB}$, t)

Problem Selection Criterion Problem $\mathcal{P}$ is adequate for learning group (S_0,..., S_k) if not exists i in 1,...,k so that solvable$(\mathcal{P}, \mathcal{SM}(\mathcal{S}_i))$ & solvable($\mathcal{P}$, $\cup$ i in (1,..., k) $\mathcal{SM}(\mathcal{S}_i)$).

Hoppe's criteria can be used to formulate a new parameterization rule, that is the inverse of the problem selection criterion, which we call the group selection criterion.

Group Selection Criterion Group $\mathcal{G}$, consisting of students (S_0,..., S_k) is an

appropriate group to complete Problem $\mathcal{P} \rightarrow$ Solvable($\mathcal{P}$, $\cup$ i in (1,..., k) $\mathcal{SM}(\mathcal{S}_i)$)
Or, if for each $\mathcal{T}$ required to complete $\mathcal{P}$ There exists an student i in (S_0,..., S_k) such that knows($\mathcal{S}_i, \mathcal{T}$)

To state our new criterion in natural language: a group is an appropriate one to solve a particular problem if the individual user models of the group members indicate that every task contained within the problem can be completed by at least one of the group members. Using Hoppe's parameterization rules, individual user models can be lifted out of their initial contexts and used to seed groups with the students who possess the appropriate knowledge to complete the group's assigned task.

Hoppe acknowledges that his parameterization rules are only useful as heuristics, and do not take into account any notion of group dynamics or any external factors affecting group performance, an observation that is also valid for the new Group Selection Criterion. However, Hoppe has developed an important heuristic for translating the knowledge possessed by individual group members, as represented in their separate group models, into a measure of the knowledge possessed by a group, that can then be used as a valuable base for a model of the group as a whole.

Quignard [56] et. al. extend Hoppe's group formation work in an interesting way by developing an algorithm to put together two-person groups for the explicit purpose of encouraging argumentation. Students in their collaborative physics tutor are asked to solve a problem which is then analyzed and graded by the tutor. The analysis of each student's solution is then used to give each possible student pair an argumentation score, with the higher the score indicating a higher likelihood the two students will argue about their solutions. The score is a combination of three elements: conceptual obstacles, which measure whether or not the students solved the problem in a similar fashion, normative obstacles, that determines whether or not one of the students' solutions violates rules in the problem domain, and solution correctness, which tries to find a reasonable distances between the correctness of the two students' solutions: not too far apart, and not too close. The algorithm then uses an unspecified optimization procedure to determine the best set of pairs amongst all the students online, and seems to have been successful in producing argumentation in most cases.

A similar system is developed in Baker et al.'s [57] CONNECT physics CSCL environment where three human judges rate seven facets of the problem solutions of students who are using the tutor. The CONNECT system then forms pairs of students with maximum semantic distance between their solutions (and minimum variation between each pair's score), as determined by the seven rated elements, so as to promote argumentation between the pair.

A related approach that relies more heavily on pedagogical background is found in Supnithi et. al's [53] discussion of Opportunistic Group Formation(OGF) which is an agent negotiation process that attempts to place users in learning groups where both the individual learning needs of the user and the social role obligations of the user in the learning group can be fulfilled simultaneously.

OGF is carried out via personal agents in a CSCL environment. These agents monitor a user's learning situation and try to form a collaborative learning group when they feel that it is appropriate for their respective learners. The agent also negotiates the appropriate social role and learning goals for its user within the context of the collaborative learning group. The situations that trigger a collaborative learning request are: when the agent detects a learning impasse on the part of the learner, when the agent feels that the learner needs a review of the material that has been presented, or when the agent feels that it would be appropriate for the learner to approach a new topic in a collaborative fashion.

Using their learner model, agents negotiate the group setup and the social/group role for their learner based on which theory of collaborative learning, and its proscribed methods of attaining knowledge, best matches the current collaborative learning possibilities as defined by the needs and abilities of the other users current available to participate in group work. OGF uses a three-level learning goal ontology (described further in [58]) to carry out the negotiations between the agents, moving from the high-level learning goals of each agent to the appropriate tactics to achieve that learning goal to the actual collaborative learning situation which must be organized to carry out those tactics. The links between the levels of the ontology that are used by the negotiation mechanism of the agents are determined by learning theories such as Vygotsky's Zone of Proximal Development [54].

Another work that combines social roles and task knowledge is found in Singley et al. [60] who describe the use of a Bayesian team model that has components for both learning team roles and domain knowledge. They identify five team roles common in learning groups: Observer, Apprentice, Specialist, Leader and Coach. These team roles appear in various common permutations in collaborative learning environments. Some of the more helpful learning role combinations are formalized into Collaborative Learning Schemas (CLS), including: Modeller-observer, in which the Coach solves a problem while the rest of the team watches, and the Great Escape, where several Specialists collaborative to solve a difficult problem.

The Bayesian model is integrated into a math tutor called 'Algebra Jam', which is a collaborative environment for solving algebra problems. The Bayesian model observes the interactions of the learners with the Algebra Jam interface and makes inferences about their domain knowledge, learning role knowledge, learning role competency and other factors based on the events learners generate on the interface. Human tutors can then examine the Bayesian network of each learner and suggest helpful CLS's, with learners taking up roles that will enhance their learning.

Serra Jr. et. al [64], discuss the use of genetic algorithms working with learner models containing information on leadership capabilities, team member preference, cooperative learning behaviour and task knowledge to form optimal groups in their MATHNET CSCL environment.

2.6.2 Group Performance Modelling: Collaborative Conversations and Actions

Work by McManus, Soller and others [39][38] has led to a better understanding of how effective learning groups function, and what aspects of group functioning should be studied to determine the performance of a group. Soller et. al [38] studied the learning performance of several different groups using a collaborative learning tool, the Collaborative Learning Interface (CLI), to complete a software domain-modelling project using an object-oriented methodology called Object Modeling Technique (OMT). The Collaborative Learning Model [39] was developed to identify the factors observed during collaborative learning: participation, social grounding, active learning conversation, performance analysis/group processing and promotive interaction, all of which we describe below. Participation, especially when all members of a group are contributing, is a self-explanatory factor. Social grounding skills are those that allow successful turn-taking and role playing activities in group conversation and activities. Active learning conversation skills include knowing how to question, educate and motivate team members, as well as being able to resolve conflict and settle disputes within the group. Group processing occurs when a group evaluates and examines its own behaviour and performance, while making changes to the way it works, if necessary. Promotive interaction describes the spirit of teamwork that occurs when team members support each other, and put the goals of the team above their own individual needs.

Based on the Collaborative Learning Model, Soller [38] developed the Collaborative Learning Conversation (CLC) Skills Network, in which certain conversational acts are identified as essential to collaborative learning and problem solving. These conversational acts, which Soller groups into three high-level categories: Active Learning, Conversation and Creative Conflict, each have their own distinctive sentence opening phrases. The CLI forces students using the OMT editor to communicate with each other, via a chat window, by first picking one of the sentence opening phrases associated with the CLC Skills Network and then completing the rest of their utterance in natural language. This system allows researches to easily place learner communication within the CLC taxonomy without having to rely on uncertain and hard natural language understanding techniques.

Doing a small study on two groups of students asked to learn OMT skills through the OMT editor, Soller found that the team that used a higher percentage of Active Learning conversational acts had a more self-rated productive learning experience. The projects turned in by the two groups at the end of the study were of similar quality, but the group with the higher Active Learning activity was seen to be more successful as a learning group due to the better educational experiences of its members.

Continuing the research, Soller [61] and her co-authors have explored the use of Hidden Markov Models on the captured dialog within the OMT groupware system (now called Epsilon). They were able to train the models to relatively successfully detect when in the learner conversations successful and unsuccessful knowledge sharing events were going to take place.

Mühlenbruck and Hoppe [59] take a similar approach by not modelling the learners in a CSCL environment at all, but by modelling their actions in a collaborative card-placement activity. Their system detects such collaborative actions as conflict creation, focus shifting or task aggregation and can adapt its presentation of the problem to assist the users in their current learning goal. Their system does log the various collaborative actions by each user, but does not keep an explicit learner model, or pre-adapt the system to meet a learner's specific needs.

Chapter 3

Group Model and Experimental Design

3.1 Introduction

This chapter has two primary purposes. The first is to summarize the discussions of the previous chapter, and by doing so, focus in on a particular type of group to study, and develop a model for that group. The second purpose is to describe a pilot study, exploring the group model that was developed, that was carried out on software engineering project teams consisting of students in an undergraduate Computer Science class.

3.2 Selecting a Group Type to Model

The goal of this research is to develop a comprehensive group model to improve the efficacy of adaptive collaborative software tools. The first step in building a model of this type is to decide on what sort of group to work with. As discussed in previous sections, the number of groups in the real world combined with those made possible by collaborative technologies is vast, and their characteristics varied. As a result, any attempt to develop a general group model would have too broad a focus and be beyond the scope of this work. Rather, the type of group to focus on should have relatively well-defined and tractable goals as well as stable membership. These properties are desirable for two reasons: ease of modelling and an ability to test the validity of the group model by testing its predictions about group behaviour and performance against the actual performance of a group or groups with those properties.

For these reasons, and due to pragmatic issues surrounding the experimental resources available, modelling student software engineering project teams, both real and virtual, is an appropriate choice. Studying virtual communities is time-consuming with uncertain and arbitrary results, intensional networks require access to many different organizational structures and do not always have well-defined goals, and knots appear too simple a group type to do any interesting work with.

3.3 The Structure and Content of the Project Team Model

The three general factors determined to influence group performance, as identified in Chapter 2, are:

1. The functional knowledge and skills of the team members that relate to the group task.

2. The teamwork skills and social intelligence of the team members.

3. The task type.

In addition to those three factors, there is another, very general, category of influence on group performance:

4. The situational and contextual factors of the group, such as team member location and external time pressures.

The following subsections describe each factor and its relation to project team performance in more detail.

3.3.1 Functional Knowledge and Skills

The knowledge and skills needed by the individual members of the group are task dependent. The knowledge and skills for each task can be represented in a concept map of the problem domain once it is determined and appropriate tests administered to determine each individual's domain knowledge. Once the knowledge required to complete the task has been accurately determined, and the knowledge levels of the individuals in the selection set known, the knowledge component of the group model can be developed and instantiated.

The group selection parameterization rule (Section 2.6) states that the members of a group must collectively cover the knowledge requirements for the task put out for them in order to be effective. Unlike social intelligence and teamwork measures, which are generally static, knowledge measurements, especially of learning groups, can be expected to significantly change if the project team's task duration is sufficiently long. This introduces the issue of model updatability, as the project team model's knowledge component may require frequent updates to accurately reflect the level of knowledge within the team.

3.3.2 Teamwork Skills and Social Intelligence

Belbin's team role taxonomy will be used as the basis of the teamwork skills component in the team model [40] as it is a good specific measure of the various characteristics that determine an individual's teamwork skills, and not a more general measure of "social intelligence" or broad personality characteristics such as the Myers-Briggs test. As discussed in the previous chapter, three personality tests have generally been used in the literature to determine Belbin team roles: the BTRSPI, the OPQ and the 16PF. The BTRSPI has been criticized in the literature for its questionable validity, but it is also used widely in industry settings to determine Belbin team roles, unlike the 16PF and OPQ tests. The OPQ's advantage is that it has a measurable four factor structure. On the downside, the OPQ's questions are related to very specific workplace situations, which many undergraduate student pilot study participants may not have experience with. The 16PF is useful in that it has more generalized questions that should be answerable by the pilot study participants, and it also includes a critical thinking component, which will be a useful measure to take into account the learning that must be done by the project team members.

One objection, with some merit, that can be raised against the use of the Belbin team roles is that they were developed for groups of managers, not software engineering teams. However, the software engineering teams in the pilot study must deal with all the management issues surrounding the project, such as task assignment, resource investigation and team morale, that are accounted for in the Belbin team roles. The members of the teams will be more active in actual task completion and possess more technical responsibilities than most management groups, but this will hopefully not diminish the efficacy of team roles in determining the teamwork behaviour of the teams.

3.3.3 Task Type

The task and task type must be taken into account when determining the knowledge and skills required for a project team to function successfully, but it is also important in determining the proper predictive balance in the team model between the functional and teamwork skills held by the group.

As discussed in Section 2.4.4, project teams can be categorized according to how work flows through the team. The type of team, as categorized by the workflow, then determines how important the task and teamwork skills of each individual group members are in determining the success of the team. For example, project teams whose team interaction style is pooled interdependence will rely heavily on the task competencies of individual workers for group success, as the amount of interaction within the group is very small and relatively unimportant.

Software engineering teams seem to work in a reciprocally independent way. Software engineering team members often give themselves responsibilities for different parts of the project; for example, two members will often be responsible for programming the application logic, one will take charge of interface design and implementation, and one will be responsible for all design, diagramming and documentation. However, the team members are all working towards the higher goal of task completion, and the task of an individual team member is highly dependent on the work of his or her team members. For example, the application programmers and UML designer are reciprocally dependent on each other for making sure the project code matches the design and vice-versa. Also, software engineering teams display the typical reciprocal interdependent teamwork behaviours of task monitoring and backup. In order to complete the project for milestone submissions or the final deadline, team members will often all focus on whichever part of the project needs completion, whether it is programming or design. However, this initial analysis of software engineeering team workflow is speculative, and the exact nature of of software engineering team workflow will be one of the important issues explored in this research.

Assuming that software engineering teams work in a reciprocally interdependent way, the relative importance that must be placed on the task ability of individual group members versus their teamwork skills in the team model can be estimated. Members of reciprocally interdependent teams have enough independent responsibility that their task knowledge must be sufficient to perform their functional roles properly, but they also must be competent in teamwork behaviours such as coordination, communication, monitoring and backup to achieve the higher level goals of the group.

Telsuk et. al [49] discuss the positive effect of maximizing the contributions of individual team members by engineering the roles people play within a group. For example, a team member with excellent programming skills should take on the responsibility for developing the application logic for the project, while one with excellent teamwork skills should be given leadership responsibilities. However, roles within the project teams in the pilot study are self-selected, and as a pedagogical goal of the class the participants are in is to teach new skills to the students and have them practice a variety of different skills, it is likely that some group members in the study will take on roles for which they are not suited. Unfortunately, Telsuk gives no quantitative guides to the weighting of the relative influence of task vs. teamwork skills that should be used within the project team model.

3.3.4 Context and Situation

The final factor that will affect group performance is the situation or context surrounding the task and the users who will complete it. The items that will exist within this component of the model are those facts about the world that are not intrinsic to any of the other modelling areas previously discussed. For example, the schedule of users around the time a task must be completed is not an inseparable part of the user's knowledge or the task profile, but must nevertheless be accounted for when trying to model how a group will complete a project. Collaboration issues will be an important part of any contextual or situational model, as they will model the amount of information and cooperation that is able to take place between members of a physically or organizationally dispersed project team. A collection of users that may at first appear to be sub-optimal for a task may turn out to be the best option when collaborative factors are calculated. For example, situational knowledge surrounding the collaboration tools available to a group will have a direct impact on their ability to communicate and coordinate tasks. A group with high-bandwidth interaction and videoconferencing tools will better be able to function as a distributed group than one that relies on chat tools and group e-mails. A situational model component will have contextual elements dealing with both users and tasks. With users, aspects such as physical location and availability will be paramount in the situational model, while in relation to projects, deadlines and resources will be the important contextual factors.

3.3.5 The Complete Software Engineering Team Model

The table below summarizes the individual and task factors that affect team performance and are therefore included as components of the team model.

Table 3.1: Student Software Engineering Team Model

Model Component	**Method of Measurement**
Knowledge and Functional Skills	Tests on Task-specific Knowledge
Teamwork Behaviour	Belbin Team Roles
Task Type	Telsuk's Task Taxonomy
Contextual and Situational Factors	Not Taken into Account

This model is an overlay model, although its structure could be translated into a constraint or BBN-model given more empirical data. Instead of representing a subset of an expert's knowledge in a domain, this model is able to represent a subset of the composition of an 'expert team' in a given domain. An 'expert team' in a domain would have knowledge of all task elements in that domain, an optimal Belbin Team Role mix and the correct type of workflow to complete the task(s) at hand.

Contextual factors present a challenge to the model, however, as they cover the 'everything else' that could affect a group. As contextual factors can be nearly infinite, and their methods of measurement varied, this team model does not explicitly record contextual factors nor measure them explicitly. As much as various contextual or situational factors can be discretized and measured, they should be incorporated in future iterations of the team model. Meanwhile their effect on the efficacy of the current team model will be limited as much as possible. It is quite possible that the effect of contextual factors can be mitigated in our study because they will be very similar in each group. Deadlines, time pressure, external activities and other situational concerns will be quite common across each group, due to each group consisting mainly of upper-year Computer Science students of a similar age.

3.4 Study Design

The proposed team model has a strong theoretical base, but the complexities of team functioning demand that empirical evaluation of the model also be carried out. As a beginning to that goal, this section describes the methodology, design and execution of a large-scale pilot study of the model that was conducted in the Winter

2002 academic semester at the University of Saskatchewan.

3.4.1 Study Issues

The purpose of the pilot study is to determine if the categories present in the team model (functional skills, teamwork, task type, contextual) are the right ones. Determining how to perform the study requires balancing two separate considerations that relate to the distinction between functional and teamwork behaviours.

The first consideration is the sustainability of the original measurements over the time it takes to complete the study task, especially concerning the evaluations of the team members' domain knowledge. If an evaluation of the functional skills of team members is performed after too long a time lapse since the task has begun, the initial measurements that were used to determine the team's knowledge level will no longer be accurate, which may also make the prediction of group performance inaccurate. However, if a measurement is done too early in relation to the task schedule, the actual skill level the group will apply to the task may not be represented, as the skills learned by the group members after the initial measurement will not be known. The second consideration is opposed to the first: the teamwork evaluation metric, which is the Belbin Team Role system, relies somewhat on the development of group-level effects, proper role placement and team chemistry. These team-level attributes will generally only emerge after a certain amount of interaction between the members of a team. One goal of the class that the project team members are in is to improve their teamwork skills, which has the potential to skew the teamwork measurements. However, similar to the contextual factors affecting group performance in the pilot study, the improved teamwork behaviours taught in the class should apply across the project teams more or less equally, canceling out any changes to the initial group behaviour measurements.

Another major issue affecting the study design is the difficulty of group performance measurement, as discussed earlier, especially the unit of analysis and process vs. outcome problems. The nature of the team model suggests a natural answer to the unit of analysis problem. As the team model is being constructed largely with the characteristics of its members, individual and task attributes will be modeled, but our team measurements will be made entirely at the group level. As for measuring team process versus measuring team outcomes, both types of team measurement will have to be performed in order to receive the maximum possible insight into the true effect of the various factors important in determining team functioning.

3.4.2 Study Participants

Selected as pilot study participants were the students enrolled in a third-year Software Engineering class at the University of Saskatchewan. The class consisted of seventy-two students divided up into groups consisting of eight to ten people each. The groups were responsible for a major project to be completed by the end of the semester. The project consisted of the design of an educational web site using current web development technologies including Java Servlets, XML and XSL. The teams were also responsible for presenting their designs for the project at various milestones to their instructor for marking and review. At the end of the semester, the entire project, including the design, documentation and finished product, were judged and an overall group mark was assigned.

This study population was chosen for several reasons. The first reason was that they were a pre-existing collection of subjects with a pre-formulated project task, thus removing the necessity for any further study around issues related to formulation of the team task and participant recruitment. The second reason was that the three-month length of the project allowed for a good balance between the valid measurement of functional skills and teamwork skills, as discussed above. A team working together for three months will have enough time to develop sophisticated team behaviours, while their individual functional skill sets will hopefully not have time to change dramatically. A concern specific to the selected teams is that their functional skill knowledge could see large improvements in the skills necessary to complete their assigned project because they are taking a class designed to teach them the skills necessary to complete the project. However, these effects should not be large in this specific class because the curriculum concentrates much more on teaching project management and teamwork skills than the functional skills such as software design and programming that are being measured. The third and final reason was that the timing of the class was convenient for purposes of the pilot study.

3.4.3 Study Design

The study consisted of giving each student in the class two series of questionnaires. The purpose of the first series of questionnaires was to get the data necessary to fill the group models, while the purpose of the second series was to gather precise evaluations of team performance from the team members themselves. The first series consisted of the 16PF personality test (from which the team members' preferred Belbin team roles were gathered), and a skills questionnaire. The skills questionnaire contained self-evaluation questions about the technical knowledge of the test-taker that was relevant to completion of the class project, such as Java programming skills and XML/XSL knowledge. These questionnaires were designed to evaluate the teamwork balance of the teams and the functional knowledge of the team members. The second questionnaire series, described further below, asked for the students' evaluations of their groups' performance, with particular emphasis on certain teamwork and functional skills. This questionnaire was administered to the students at the end of the academic semester, after their projects were completed.

For the evaluation of group performance, two types of questionnaire were used, allowing determination of team outcome and team process measures. The first, the outcome-based measure, was the project marks given to each team at the end of the semester. However, because these marks can be affected by a variety of outside factors unrelated to intrinsic team ability (e.g. high course load, sickness, etc.) the internal group evaluations gathered from the second questionnaire series done by the team members were used as a separate process measurement of group performance.

Given the minimally controlled nature of this experiment, there was no access to extensive and detailed logs of the actions and communications of the groups, details that would be necessary for the more involved process measurement methodologies such as shared mental models and the collaborative learning conversation model, as discussed in Section 2 [38]. Therefore, to measure and evaluate team process performance, behavioural summary scale tests were administered to each team member. The behavioural summary scale tests asked each team member to rank their team's performance on the seven key factors of teamwork [34], described previously, on an ordinal scale of one to five with narrative examples of teamwork behaviour used to illustrate behaviour associated with each score.

For ethical reasons, the students remained anonymous to the primary researchers, and the group numbers were randomized so that the primary researchers remained unaware of the academic performance of the students participating in the pilot study. An assistant with a background in clinical psychology was used to anonymize the study and to administer the 16PF personality questionnaire to the study participants.

Chapter 4

Pilot Study Results and Discussion

4.1 Pilot Study Results

4.1.1 Execution of the Pilot Study

A number of strategies to gather the necessary data for both questionnaire series from the students were considered. The initial idea for the first series, to gather the data directly, was impractical, as the time needed to complete the tests was longer than the lecture period for the class. The second idea was to hold 'open sessions' external to the class time where students could drop in at their leisure and participate in the study. Four of these open sessions were held in the middle of the semester but had disappointing turnouts. An alternative strategy was then developed that involved meeting with the groups directly during their self-scheduled meeting times and having the group members take the tests then. This strategy proved successful, as data was obtained from 61 students, out of the 72 who were enrolled in the class. Of the 9 project groups, 4 were covered completely and 4 had data from one student missing either because of absence from the group meetings or a refusal to participate. The ninth group was unable to arrange a time with the researches to participate in the study. The second series of tests, the behavioural scale summary tests to measure team process, were gathered from the students after completion of their final examination, with every student in the class completing the second questionnaire.

4.1.2 Evaluating Team Process and Outcome

To measure the success of the teams in the study, one outcome measure and five process measures were used, as discussed above. The outcome measure is the mark that the project teams received on their project, and the process measures are derived from taking the median score from a post-project ordinal group questionnaire given to each team member (that also included a section for comments on group functioning). The median score was used because the measures were ordinal, thus not allowing use of the average function. No other metrics, such as lowest rating in each category, will be analyzed in detail in this work, because preliminary analysis indicated that there were no promising results in those directions. The table below displays the relationship between the group mark and the median score each group gave itself on each of the behavioural summary scale questionnaires.

Table 4.1: Process and Outcome Team Measures

Group	Mark	Comm.	Feedback	Leader.	Monitoring	Orient.
1	80	4	4	4	4	4
2	83	3	3	3	3	3
3	88	4	4	3.5	3	3
4	92	2	3	2.5	4	2.5
5	75	3	3	3	3	4
6	85	3.5	4	3.5	4.5	3.5
7	86	5	5	4	5	5

Immediately, some odd results are apparent. First, the group with the highest overall mark, group 4, has either the lowest or second lowest marks on all of the team process measurements. The team with the second highest mark, group 3, also does not have particularly high team process rankings, although they fare better in that respect than group 4. More encouragingly, the group with the lowest mark, group 5, has team process scores that are also very low, but the team with the third lowest mark, group 2, has very positive process scores. At first glance, there does not seem to be any general and consistent correlation between the outcome measure of group success (the marks) and the process measure (the average rating given by the group members to the effectiveness of their group's teamwork processes). The unintuitive process scores are not an artifact of the self-rating process, as these comments (as taken from the comments section of the team-rating questionnaires) from a member of the high-mark group 4 demonstrate:

> We tried to avoid creating conflict or an uncomfortable environment by not evaluating production. Looking back, this was a very bad policy.

Another member of the same group agrees:

> No one really said anything about the project during the meetings; what was said was unclear, but no one really questioned it. Mistakes were often made. [...] Leadership was our greatest problem - no direction, no goals and a lack of commitment. We finally put in leadership near the end, but group members resented it.

Looking deeper into the results, a more nuanced picture of the data appears. Calculating the correlations between the process ratings and the team marks, two are significant: Communication behaviour, as determined in the behavioural summary scale questionnaires, correlates positively at a .5 level ($p < .15$) and Leadership behaviour has a significant negative correlation at -.53 ($p < .15$).

As the measures for Feedback and Monitoring and Backup behaviour also had positive, but non-significant, correlations with the mark, it can be inferred that groups with effective means of co-ordination and cooperation had a distinct benefit when it came to project success. Good organization, as well as the ability to correctly give and receive evaluations of work being done on the project seem to be an indicator of project success. This hypothesis may have some connection with the highly distributed nature of the groups, a theme that will reoccur throughout this analysis. Comments by the students in their evaluations lend some support to this hypothesis: one group member of one of the poorest performing groups says in regards to the Monitoring and Backup behaviour of his group that:

> Tasks were left and [we] rushed to finish at [the] end. [...] We talked about lots of stuff except [for] what we needed to[o do].

Another member of the same group remarks on their Feedback abilities:

> Language was a major barrier. One team member had almost no English skills.

Finally, a last comment from a member of the ill-fated group regarding the reluctance of poor groups to give negative feedback:

> Poor evaluations were few but good, encouraging evaluations were abundant.

A member of the worst performing group commented regarding their Communication behaviour that

> [The] team tended to work on their own parts and did not care whether or not the other people were doing stuff correctly.

As for the view of a more successful group on the importance of Communication behaviour, there are comments from the members of one of the highest raking groups:

> Good feedback, but a little slow [...] Most people helped out others if they ran into trouble [...] We kept track of progress on the project and any time someone needed a hand all they had to do was ask.

Other members of successful project teams commented that:

> I believe that every member in the group missed at most two meetings and we all had each others' e-mail and phone numbers so communication was never a problem.

While this selection of quotes does not provide irrefutable proof that communicative, supportive and knowledgeable teammates are vital to the success of these student project teams, there certainly seems to be evidence that such teammates were a valuable asset to have present.

Counter-intuitively, the median scores for Leadership behaviour were negatively correlated with team outcome success. While it might be reasonable to see no correlation at all between good teamwork behaviours and group success, it is very surprising to see such behaviour portrayed negatively. Examining the qualitative statements regarding project leadership from poorly performing groups we see comments such as:

> Many times we did not not know what to do, as [a] group we would decide [...] There were no leaders, occasionally a few of us said that we had to get it in gear [...] Two members took charge because everyone was so disorganized.

These statements seem to indicate that a lack of leadership was a negative factor in terms of group success; however, statements from members of high performing teams have a very similar tone:

> We did not have a clearly defined leader as nobody really wanted the job. I believe this ended up hurting us as the deadline drew near [...] Two people emerged as leaders in the project. Unfortunately this seems to have divided the team into two waring factions now that contribution assessment has become an issue [...] This was our greatest problem - no direction, no goals and lack of commitment. We finally put in leadership near the end but group members resented it.

It appears that the negative correlation of leadership behaviour to project outcome success can be explained by a general association made by the students of leadership with unpopular, heavy-handed behaviour, especially as project deadlines were drawing near. It may also be that the leadership qualities rated highly by team members such as pleasantness and flexibility, especially by students with little real-world work experience, are not the qualities most conducive to team success. Also contributing to the negative correlation of leadership behaviour was the success of the top three teams despite generally mediocre leadership ratings; a finding which will be discussed further in section 4.3.

In general the teamwork behaviour measures were a useful tool to analyze in more detail the performance and makeup of the teams in the pilot study. The main drawback to the measurement instruments used in the pilot study was that they appear to have been asking the students for too much precision and insight into their team functioning. Examining the qualitative comments on the summary scale sheets, it was apparent that many students conflated the ratings of the Communication, Feedback and Monitoring behaviour, as well the the Leadership and Communication behaviour. Possible solutions to this problem for future studies include the presence of expert observers trained in observing and measuring team functioning, or the simplification of the teamwork evaluation instrument for future uncontrolled studies. The small sample size of the study also prevented many statistically significant conclusions regarding the results.

4.1.3 Evaluating Functional Knowledge and Team Outcome

Further analyzing the data, Table 4.2 and Table 4.3 display the self-assigned scores from the students regarding their knowledge of the skills needed to complete the project.

Table 4.2: Median Team Functional Knowledge

Group	**Mark**	**Java Med.**	**OO Med.**	**HTML Med.**	**XML/XSL Med.**
1	80	2	2	2	1
2	83	2	2.5	2	1
3	88	3	3	3	1
4	92	3	3	3	1
5	75	3	3	2.5	1
6	85	2	2	1.5	1
7	86	2	3	2	1

The correspondence between a team's self-rated functional knowledge and their project mark does not provide further insight into the alchemy of group success. The measures of median Java knowledge, median HTML knowledge and median

Table 4.3: High Team Functional Knowledge

Group	Mark	Java High	OO High	HTML High	XML/XSL High
1	80	3	3	3	2
2	83	3	3	3	2
3	88	3	3	3	1
4	92	3	3	3	1
5	75	3	3	3	2
6	85	3	3	3	1
7	86	3	3	3	2

OO knowledge all correlated with the project mark in the .3 range, but none of the correlations were statistically significant. Interestingly, Groups 3 and 4, those that had the highest marks, rated the highest in all of the median knowledge measures. Unfortunately, Group 5, the worst performing group, also had virtually the same ratings as those two groups and there are no apparent correlations between the other groups and their placement in terms of the project mark.

The possibility of functional roles being similar to Belbin team roles in the sense that it may only need one person in each group to fill each role was discussed above, but the comments made by many group members do not support that theory. Work in almost all of the groups looks to have been - or attempted to have been - evenly distributed amongst the team members, thus requiring that each group member be proficient in each technical area of the project. This means that the impact of one person who is a guru or expert in an area is generally limited to their small piece of the project, although having domain experts in a group should be expected to help team functioning in the area of feedback behaviour. Regardless, the predictive power of the highest individual functional role scores (e.g. the team member with the highest amount of knowledge on a particular topic) seems limited in this pilot study because the individual high scores are almost identical for each group.

4.1.4 Evaluating Belbin Team Role Scores and Team Outcome

Table 4.3 outlines the distribution of Belbin Team Roles amongst the project teams.

Table 4.4: Belbin Team Role Distribution

Group	Mark	No Able Member	Strongly Suited Member
1	80		
2	83		IM and TW
3	88		M-E and RI
4	92		M-E and RI
5	75	CF	
6	85		M-E and PL and CF
7	86		

This table illustrates how many team roles could be either well filled or not filled at all in each group, as determined by the results of the Belbin team role test. The Belbin scores are on a scale from one to ten, and if a person scores under four on a certain role they are considered unable to successfully fulfill that role. If a person scores eight or above on a role, they are considered exceptionally able to fill that role. If a role score is between four and eight, the person is considered competent but not ideal for that role. In the first column, it is shown that there was only one role in any of the groups that could not be potentially filled by one of their respective team members. Additionally, there are very few people in the entire study that are well suited to any role, and the most in any one group was three.

The most noticeable thing about the team role scores is how similar they are in each group. Most of the students scored within the average range for each team score, and almost all of the groups (with the sole exception of the Finisher role in group 5) had at least one person with an average or better score in each of the team roles. It is interesting that three out of the four best scoring groups had a team member able to excellently fill the Monitor-Evaluator role and the worst scoring group lacked someone to competently play the Finisher role. Generally, the team role scores provide very little in variation or insight into the performance of the teams in the study. While Belbin's ideal team role mix has one type of each team role present, there are more nuances present in his analysis. Belbin says that obviously bad teams are much easier to spot than obviously successful ones. Some particularly bad teams are those in which all the members have the same team role. None of the teams in the experiment match that description. Belbin also provides a list of particularly toxic team role combinations, the only one of which that could possibly apply is a Team Worker/Implementer/Finisher group with no Resource Investigator, Plant, Shaper or Chairman available, a combination which group number two somewhat fulfills. Belbin also comments that good team morale does not always lead to good team results, as a team that has no conflict issues will usually have a pleasant experience but not a successful one. This may explain some of the discrepancies between the team self-ratings and their observed results.

As for well-performing teams, Belbin believes that four factors are very important: i) the presence of a good chair, ii) the presence of a good Plant, iii) a mix of team roles and iv) the correct assignment of team roles within the group. In the experimental groups, there were no members with clear leadership tendencies (Chairman or Shaper), and only group 6 had a clear Plant member. Belbin also believes that teams need at least once creative/clever person to succeed. In Belbin's team role taxonomy, Plants, Resource Investigators and Monitor-Evaluators most often proved that creativity or cleverness. Groups 1, 2, 5 and 7 did not have anyone filling those roles (and were also three out of the bottom four finishing groups), but it is also possible to have high creativity and/or cleverness without a high score in those roles. With so few clear team roles being filled by the experimental participants, it is hard to gauge how important team role mix was to the performance of the groups in the study.

4.2 Methodological Issues with the Pilot Study

Complicating our ability to effectively analyze elements of the study are some validity concerns surrounding the application of the 16PF personality test. These concerns are as follows:

1. The norms used to score the 16PF test were not from a Canadian population sample, but from a British one. Scores in the 16PF are generated using population norms. The only norms available at the time of test acquisition were the British ones, which may have caused aberrant scoring if the norms have large differences.

2. Many students in the pilot study were foreign students. The16PF is normalized on a Western population sample, and is based on Western personality theories which bring into question its cross-cultural validity. English-language comprehension appeared to be a large issue with many of the foreign students, which was an issue for all of the questionnaires, but especially the 16PF, with its long-format questions and abstract concepts.

4.3 Overall Study Analysis

Examining the data from the pilot study, it is hard to make any definitive statements about the factors influencing student software engineering team success, and therefore about what the content of the team model should be. Good communication behaviour is positively correlated with team success and leadership behaviour, as understood by the students at least, is negatively correlated. Some of the best performing teams rate their functional knowledge very highly, but the worst performing team does as well. The Belbin team role data was disappointingly ambiguous and unhelpful as well. However, functional knowledge and teamwork behaviour are only two of the four factors that were identified as affecting team performance, leaving contextual factors and workflow.

The assumption made at the start of the study was that the task type of this project was "Reciprocally Interdependent", where team members have specialized work skills and defined roles within the team, but there is a back-and-forth movement of work between the team members and there is a lot of collaboration and coordination done by the team members in order to reach their higher-level goals. For a group facing this type of task, the teamwork and functional skills possessed by the individual members are of equal importance in determining the success of a group [49]. However, the workflow appears to have been of a different type in many of the groups, as evidenced by the following comments from members of different groups:

> Tasks were dealt with as one unit, and were generally limited to one person.
>
> Everybody went about their own tasks, communication was limited to assignment of responsibility.

The description of the workflow in these groups is described by Telsuk et. al as "Pooled Interdependence" in which the performance of a group can be described by adding the individual task contributions of each individual [49]. In a group with this type of workflow, each member works on one task until complete with very little communication or coordination with other members. The success of this type of group is generally determined by the task skills of each of the individual members as teamwork behaviour plays a very small part in the functioning of the group. This provides an initial explanation for the seeming importance of task skills in predicting the outcome success of some of the project teams. Unfortunately, there is not rich qualitative data describing the manner in which each group chose to tackle the assigned project, and it is quite possible that some groups worked in a "Reciprocally Interdependent" manner by having a horizontal division of tasks amongst the group members (e.g. each member is responsible for one particular skill area of the project) and relying more on coordination and other teamwork behaviours. It is even possible that some groups moved between the two types of team workflow as the project progressed which is supported by the positive correlation of communication behaviour to project outcome success, indicating that good teamwork behaviours were still a significant factor. Determining the type of workflow within the teams is very important, because it is a significant determiner of the relative importance of the other factors in the team model have to team success.

While there are some small correlations present between the task and teamwork abilities of the groups and their performance, the contextual and situational factors surrounding uncontrolled group activity, as discussed in section 3.2.4, appear to have been a significant factor. A member of group 3 states:

> Most of our group members worked hard on the project early on but many began to place it second to other work until at the end there were only four of us working on it at all.

Similar comments were made by members of group 5 and group 2 respectively:

> The last day of [the] project was really bad. Only four out of nine people showed up to finish.

> Lots of problem with English and people doing what they wanted to as opposed to what was agreed upon.

The contextual problems outlined by various group members took on two general forms. The first type of problem involved the scheduling difficulties of various group members. Especially during the critical last few weeks of the project, it appears that many of the students reduced their time commitment to this project in order to focus on their other courses. The second type of problem involved lack of communication within the team due to language issues. Many of the groups had a large number of foreign students for whom English was not a first language, and there were several comments by the students on the difficulty of maintaining positive communication in such a situation (although it should be noted there were two groups that consisted entirely of foreign students with a similar linguistic background).

In an uncontrolled study such as this, contextual factors are hard to measure and even harder to control, but can have a major impact on the performance of a group. A team with good task skills and a positive teamwork balance will still not succeed on a given task if they are unable to meet because of other deadlines or unable to communicate with each other due to language differences. Some of the external factors, such as deadline pressure, may be similar in each of the groups, but without more formal measurements, it is impossible to know for certain. The initial impressions given by the data indicate that teamwork and task skills had an impact in the performance of the teams in the pilot study, but the contribution of contextual factors should not be forgotten, even if they are not explicitly quantified in this experiment.

It was hoped that a Belbin team role analysis would provide further insight into the performance of the groups in this study. However, due to a combination of factors including methodological issues, possible intrinsic problems with the Belbin Team Role system (viz. Section 1.5.1) and a possible mismatch between the intended participants of the Team Role system and the group members in this study, the team role analysis is inconclusive. This result does not mean that the Belbin Team Role system is not a useful tool for analyzing software development or student teams, but the experience from this study indicates that it does not perform well in an uncontrolled study, especially one that contains many non-Western participants.

4.4 A Team Performance Prediction and Analysis System

One possible use of a team model is a team-outcome prediction system. This system would take in an instantiated team model and predict the team's likelihood of success at a particular task. Such a system could then be used in an educational or work environment to build optimal teams to tackle particular projects.

One way to fashion a team prediction system is to use a classifier algorithm to build a decision tree containing rules about how to classify teams as successful or non-successful. An instantiated team model can be sent down the tree with various attributes of the model being tested at each node and then being routed down different branches of the tree depending on the results of the test. All instantiated models will eventually descend the tree until they stop at a leaf node that gives the instance a classification because no further branching is possible. Another benefit of using a classifier algorithm to analyze teams is that the classification rules are available and can be used to refine the team model itself.

A typical classification algorithm, such as C4.5, builds decision trees inductively by parsing pre-classified examples. Once the tree is constructed, it can then be used to predict the classification of future instances. For a team-prediction system using the team model framework discussed in this thesis, the instances consist of all the non-outcome information for each team, which includes the median functional skill scores for the entire group and the complete Belbin team role scores for each group member. The classification of each team is then either its functional outcome (group mark) or the process outcomes (Teamwork Behaviour Ranking, Feedback Behaviour Ranking, etc..) that are associated with the instance.

C4.5 itself works by examining each attribute contained in the instances and deciding whether or not to split the tree (e.g. add a decision node) on that attribute at that level in the tree and add branches from that node representing each possible value of the attribute. The decision is based on evaluating the estimated information needed to classify instances reaching that node in the tree. The information measure is expressed in 'bits' and is calculated for each attribute at each level of the tree. The splitting of the tree continues until the leaf nodes are pure (e.g. they contain only instances of one class) or any further splits will not add any benefit. The information measure used is called the entropy, and is calculated as follows:

entropy(p_0,p_1,...,p_n) = -logp_0 -logp_1 -logp_2 where each argument represents the fraction of each class represented in a daughter node compared to the total number of classes.

Generally, the data set for a machine learning algorithm is split into two: the training set and the test set. The training set is used to construct the decision tree and the test set is used to evaluate the error rate of the decision tree. The error rate is the number of mis-classifications that the decision tree makes on the test data set compared to the size of the test data set as a whole. The more complicated standard for machine learning evaluation, especially in situations where there is a small data sample, is stratified cross-validation. Using the stratified cross-evaluation technique, the data set is sampled randomly N times to construct a training set with two-thirds of the instances and a test set with the other third on the instances. Each sample is manually examined to make sure the the training set and the test set both have all of the classes represented in equal proportion to their presence in the full set (this is the stratification). The error rate is then calculated by averaging the error rate of all N decision trees, and is taken to be a good estimate of what the error rate would be on an independent test set.

The first step towards generating a classifier for use in group prediction is to actually classify each instance in the group performance data set (that is, each instantiated group model from the pilot study). The classifications in this case are the task outcomes of each group. There are six outcomes resulting from each group's project work, the first being the mark the group received and the other five being the median self-rated teamwork measures gathered from the Dickinson Behavioural Summary Scales. To generalize the classifiers' results, each measure is divided into thirds, with group scores in the first third of the group results being declared "Poor", scores in the middle third being declared "Average" and scores in the top third being declared "Good".

Using the J4.8 classification algorithm (a variant of C4.5) with 5-fold cross-validation, only the decision tree generated for predicting the project mark had a reasonable predicted success rate. The decision tree and error rates for that classification are:

Group Mark:

If Median Java Knowledge <= 2: Medium (3.0 instances)

Else If Median Java Knowledge > 2

    If Median HTML Knowledge <= 2.5: Poor (2.0 instances)

    If Median HTML Knowledge > 2.5: Well (2.0 instances)

Cross-validated success rate: 71.4

This indicates that the classification algorithm believes that only the median knowledge measures have predictive qualities in relation to student software engineering team performance, at least given the data in the pilot study. Median Java, OO and HTML knowledge were positively correlated with the group mark, although not with a high level of confidence, and the classifier confirms that weak correlation.

Chapter 5

Research Contribution and Future Directions

5.1 Research Contribution

At the start of this work, the nascent status of the work relating to team modelling in adaptive systems was noted. Most work done on the modelling of teams in the adaptive systems community focuses on the formation or analysis of short-term learning groups with explicit pedagogical goals within a highly structured pedagogical environment [38] [53]. The major conceptual and empirical contribution of this work is to extend the explicit modelling of student teams beyond short-term groups with very limited goals and to use research grounded in the organizational behaviour and psychology literature to provide a fully realized structure for an effective and complete team model.

Most of the projects discussed in Section 1.7 use individual user models as inputs to a team formation system. This work follows that approach by using a classifier algorithm to generate a decision tree that takes a set of individual user models as input and outputs a predicted outcome for a team composed of those users. The use of machine learning techniques to generate the group formation algorithm is relatively rare (but see [64]), and has never been used to predict the outcome of long-term groups.

In the introduction three specific and important questions regarding the details of group modelling were posed:

1. Should functional knowledge be measured at an individual level or a group level?

2. How big a role do social and teamwork skills play in group success?

3. How can social and teamwork skills be effectively measured?

The following subsections will look at the specific ways in which our approach of dealing with longer-duration teams and considering task, social and functional factors in team functioning has contributed to the knowledge regarding group modelling in the user modelling and adaptive systems communities.

5.1.1 How to Measure and Model Functional Knowledge

As discussed in Section 2.3.1, one of the fundamental issues with analyzing group performance is the 'Unit of Analysis' problem. Do you measure group attributes at an individual level or at the group level? This question is especially interesting when considering the functional abilities of the group members. Obviously, measurements of knowledge must be made at the individual level, so the Unit of Analysis problem is more accurately expressed in this context as the means by which the team members' individual task knowledge is translated into the ability of a team to solve problems involving the use of that knowledge.

The best guess on this subject currently found in the User Modeling literature is expressed by Hoppe [28] and his Problem Selection Criterion which explicitly assumes that a team which has members capable of performing each task necessary for the completion of a problem will successfully solve that problem. Variations on the Problem Selection Criterion were also used in work by Quignard et. Al [56], Supnithi et. Al [53] and Baker et. Al [57], all discussed in Section 2.6.1. In Section 4.1.3, the analysis of the pilot study results showed that this assumption was not warranted for the teams in the study. Qualitative responses by the team members showed that work was generally equally partitioned out in a vertical manner, so that the success of the team on the project depended on the level of functional skills of each of the team members. This result underscores the importance of understanding the nature of the flow of work through a team, and how it affects the relative importance of functional knowledge versus teamwork skills. This finding also indicates that a truly comprehensive team model would have to dynamically model the current assignment of tasks to individuals, and those individuals' task capabilities at that time, to have an accurate understanding of the actual functional capabilities of a team at any given point in time. Analysis of group workflow and its effects in the pilot study is discussed in Sections 2.4.4, 4.1.2 and 4.1.3.

To be fair to the research backing the Problem Selection Criterion and related work, they were not meant to function as formation rules for teams working on complicated tasks with many subgoals. Rather, they were meant to be used to form teams that were trying to achieve one very specific learning goal that would not last for a long-duration (groups closer to knots than teams). Even so, such linear individual-to-team transformation rules have showed some relevance in our work. Qualitative feedback from some of the most successful groups suggests that feedback and backup behaviour are very important in their success, indicating that team members skilled at various domain tasks are useful. Intuitively, the advantage of having domain experts on each task a team needs to complete seems obvious. Unfortunately, the self-rated functional knowledge tests were not detailed enough to identify true domain experts in the study group, so further exploration of this topic would be welcome. Also complicating the ability of the team model to accurately measure the impact of task expertise within a team is the wide availability of outside learning resources available to team members in the software engineering field.

Given the remarks considering the vertical division of tasks within a group, it should be expected that teams' self-rated median knowledge measures correlated positively with team outcomes. While there are positive correlations, and the top two performing groups have the best self-rated scores, the correlations are not significant, mostly due to a small sample size. However, this may also be due to noisy data or the coarseness of the self-rating measures.

A final contribution of this work regarding the measurement and use of functional knowledge in teams is more of a cautionary note for future researchers than an advance: self-rating functional knowledge questionnaires given at the start of a fairly long-term study are not the best indicators of participant knowledge over the length of the study. This issue is a problem for study designers for both long-term and uncontrolled studies. To keep a team model updated with regards to the functional knowledge of the team members, especially in an educational setting, would probably require weekly updates of team member knowledge. Although self-rating questionnaires are not without use, more comprehensive questionnaires that use external evaluation of the participants' knowledge would be much more useful. However, giving detailed weekly quizzes to participants in an uncontrolled study may place an unsustainable burden on both the study team and the study participants. The issue of diagnosis of student competencies and skills is a long-standing one within the Intelligent Tutoring System community.

5.1.2 The Contribution of Teamwork Skills and Task Type to Team Success

A common sentiment expressed by colleagues during the performance of this work was that the most successful teams in the pilot study would be those "that had all of the smart people in them". Belbin [40] had a similar intuition regarding his management teams: he assumed that the groups filled with the high intelligence managers, which he called his "Apollo Teams", would generally do the best at the management games Belbin was testing the teams in his study with. Belbin was consistently surprised to see that the Apollo Teams tended to rank in the middle of the pack or lower when the management games were analyzed. Upon further analysis he saw that the Apollo Teams were full of similar personalities who would spend more time arguing for their own personal plans and tactics rather than doing the necessary work to make the team successful. Belbin therefore concluded that Apollo Teams were a team group type to be avoided, rather than welcomed.

There were no classic Apollo Teams in the pilot study, at least as determined by the Belbin Team Role system, but there is some evidence from the study that certain types of teamwork skills do have an effect on the outcome on the teams' performance on their project. As discussed in Section 4.1.2, Communication teamwork behaviour, as judged by the students, correlated positively with project outcome success, and leadership behaviour correlated negatively. There was also a lot of anecdotal evidence given by the students that teamwork behaviours such as task commitment, backup and monitoring contributed to the success of certain teams.

These observations regarding the utility of teamwork behaviour echo those of Telsuk [49] which were discussed in Section 2.5.4. Telsuk's group workflow taxonomy has proved to be a useful analytical tool in regards to the results of the pilot study, and the integration of the workflow taxonomy into the student software engineering team model is, to the best of our knowledge, a first in the field of team modelling. The impact of different workflow types on the student software engineering team model were discussed in further detail in Section 4.4.

Tempering the usefulness of analyzing the pilot study results is the large impact of contextual factors on the outcome of the team project. A contribution of this work is the identification of contextual factors such as external time constraints as an essential, if hard to measure, part of any team model in non-trivial situations. These contextual factors are also one of the main causes of the "measuring team process versus team outcome" issue in team behaviour research. Does one team perform better than another because of an intrinsic advantage or because the members are able to spend more time on their tasks?

5.1.3 How to Effectively Measure Teamwork and Social Skills

Unfortunately, this work has not succeeded in providing any definitive insights on how to seed a team model with effective individual measurements of teamwork behaviour. Belbin Team Role Scores were used as part of the pilot study, but methodological issues, both on the part of the study coordinators and with the Team Role Scores themselves, appeared to sabotage their efficacy. It was hoped that a Belbin team role analysis would provide further insight into the performance of the groups in this study. This result does necessarily mean that the Belbin Team

Role approach should not be explored further, but its predictive capabilities did not show up in our pilot study.

5.2 Future Research Directions

Future research directions arising from this work fall into two general categories: the further refinement of the student software engineering team model and the use of the team model in adaptive software applications. The following two subsections discuss both future research directions in greater detail.

5.2.1 Improving the Team Model

The four-part student software engineering team model developed in this work provides a solid template for team models in the domain of student software engineering and should be extensible to other domains as well. However, more work is necessary to refine each part of the team model. This work should primarily involve more studies being conducted that make one section of the team model the dependent variable so that the cause of changes in group performance can be more accurately determined. While one of the major contributions of this work is to extend team modelling over a longer time scale, the team model would benefit from studies relating to teams over both short-term and long-term time periods.

For improving very specific aspects of the model, studies should be carried out that are similar to those in the work of Soller et. al [38] and Feinman et. al [65], who observe group dynamics in tightly monitored virtual worlds and where teams do specific tasks within those worlds. In the Feinman work, a groupware program called VesselWorld was developed where small teams of users work together to clean a virtual harbor of toxic waste using tools provided by the virtual environment. All conversation between the team (using the VesselWorld chat program) is recorded as is use of any collaborative mechanisms available in the environment. In Soller's work a software engineering team is observed solving software engineering problems using a collaborative environment, although participants are videotaped as well. Neither work, however, concerns itself with explicit team modelling. Rather, both studies focused on analysis of the discourse produced by the teams in the study, and the identification of speech acts that indicate excessive cognitive load [65] or poor collaborative learning techniques [38]. The measurement of team performance factors such as cognitive load, cooperating and learning could then be correlated with the factors composing of the student software engineering team model developed in this work.

The advantages of this short-term study approach in refining the team model are several. The first is the controlled nature of such studies. By virtually eliminating contextual factors from the study, the simulated environment approach would allow each part of the team model, such as task competency, to be isolated and its effect on team performance measured properly. The second advantage is the extensive monitoring of the participants in the study, using both virtual and physical techniques. One of the drawbacks of the pilot study in this thesis work was the general lack of fine-grained observations about the actual behaviour and work processes of the teams. The Behavioural Summary Scales were used as a post-hoc measurement in this work, but expert analysis of all group interactions would be a far superior method of determining the exact nature of team behaviour. Using expert analysis of team interaction also allows the interaction to be measured more precisely, which gives the researchers the ability to rank groups on both team process and team outcome. The final advantage of this approach is the ease of study execution. There is a lot of overhead and uncertainty involved with conducting an uncontrolled study that can be eliminated by use of dedicated participants and a virtual or otherwise controlled environment.

The section of the team model that cannot be tested in this sort of study is the one concerning the contextual factors surrounding team performance. However, once the other factors are better quantified using controlled studies, the team model can be re-examined in more real world, uncontrolled situations, and changed accordingly. In future long-term studies, there should be a concerted effort to log and analyze as much as possible all of the interactions of the team. This sort of analysis would be more plausible with highly distributed teams that would likely use a large number of collaboration tools as a natural factor of their interaction.

Another area where the team model can be improved is through the use of an effective measurement of teamwork behaviour. The problems encountered in the pilot study with the Belbin Team Role system are documented above, and the correct measurement of individual social competencies is essential to the population of the team model.

The last suggested future research direction involving the refinement of the team model can follow the gathering of more empirical data from the controlled group studies, and that is the development of a Bayesian Belief Network team model. With the number of contextual factors involved in team performance and the imprecise nature of social skill measurement, a probabilistic approach to modelling team behaviour would be an advancement over the overlay method presented in this work.

5.2.2 Using the Team Model

The team performance prediction system has been presented in this work as a possible use of the student software engineering team model. Once the team model is further refined, the decision tree used in the system should become even more effective. An interesting extension of the team prediction system would be the optimal generation of effective teams in an education setting. It is a combinatorial problem to generate a set of effective teams from a given population of possible team members such that the average predicted effectiveness of each team is the highest possible. A system that could perform that task would have great potential usefulness in any education setting where teamwork is mandatory.

Collaborative systems would also benefit from the adoption of the team model in many ways. Given a certain team's competencies, a collaborative environment could attempt to pull in outside resources to provide help to the team, modify its interface in some way to better complement team functioning or merely update the team model for use in team reflection or external evaluation.

5.3 Final Conclusions

The overall goal of the pilot study was to confirm the structural validity of the four-part team model. While there was not conclusive proof overall that the four-part model is the ideal way to model teams, there was enough evidence gathered to support further work on the model. The first, and strongest, type of evidence to support the model are the qualitative comments from the students who participated in the study affirming the influence of the various factors captured in our team model. To quote:

> Many times we did not not know what to do, as [a] group we would decide [...] There were no leaders, occasionally a few of us said that we had to get it in gear [...] Two members took charge because everyone was so disorganized.
>
> Good feedback, but a little slow [...] Most people helped out others if they ran into trouble [...] We kept track of progress on the project and any time someone needed a hand all they had to do was ask.
>
> [The] team tended to work on their own parts and did not care whether or not the other people were doing stuff correctly.

These quotes, and many others like them, support the existence of the teamwork section of the team model, and, to a lesser extent, the importance of the modelling of functional skills.

There was also some quantitative evidence to support part of the team model, in this case the Communication behaviour measure correlating with team outcome performance at a .5 level ($p < .15$). The final bit of evidence supporting our use of a four-factor team model is the 71 percent score for the five-fold cross-validated evaluation in our J4.8 machine learning team prediction system, using the functional measures contained in the team model.

Taken together, these measures provide enough encouragement to support future work on the team model and to further explore its many potential uses.

References

[1] Steinfeld, Charles, Jang Chyng-Yang and Pfaff, Ben. Supporting Virtual Team Collaboration: the TeamSCOPE System. *Proceedings of the 1999 International ACM Conference on Supporting Group Work. 1999*

[2] Nardi, Bonnie A., Whittaker, Steve and Schwarz, Henrich. It's not What you Know, it's Who you Know: Work in the Information Age. *First Monday Magazine.*

[3] Salvador, Tony and Bly, Sara. Supporting the flow of Information Through Constellations of Interaction. *Proceedings of the Fifth European Conference on Computer-Supported Collaborative Work. 1996*

[4] McCalla, Gord. The Fragmentation of Culture, Learning and Technology: Implications for the Artificial Intelligence in Education Research Agenda in 2010. *International Journal of Artificial Intelligence in Education. V. 11. 2000.*

[5] Greer, J.E., McCalla, G.I., Cooke, J.E., Collins, J.A., Kumar, V.S., Bishop, A.S., and Vassileva, J.I. Integrating Cognitive Tools for Peer Help: the Intelligent intraNet Peer Help-desk Project. In *Computers as Cognitive Tools: the Next Generation.* 1998 S. Lajoie Ed.

[6] Engeström, Yrjo, Engestrom Ritva and Saarlema, Ritva. Computerized Medical Records, Production Pressure and Compartmentalization in the Work Activity of Health Center Physicians. *Proceedings of the ACM 1988 Conference on Computer-Supported Collaborative Work. 1998.*

[7] Rheingold, Howard. *The Virtual Community: Homesteading on the Electronic Frontier.* 1993

[8] Kim, Amy Jo. Killers Have More Fun. *Wired Magazine.* May 1998 V. 6.05

[9] Mynatt, D. Elizabeth, Adler, Annette, Ito, Mizuko and O'Day, Vicky L. *Design for Network Communities. Proceedings of the ACM 1997 Conference on Human-Computer Interaction. 1997*

[10] Mynatt, D. Elizabeth, Adler, Annette, Ito, Mizuko, Linde, Charlotte and O'Day, Vicky L. The Network Communities of SeniorNet. *Proceedings of the Sixth European Conference on Computer-Supported Collaborative Work. 1999*

[11] Turkle, Sherry. Life on the Screen: *Identity in the Age of the Internet: Touchstone Books. 1997*

[12] Rossney, Robert. Metaworlds. *Wired Magazine.* May 1996 V. 4.06

[13] Dunbar, R. *Grooming, Gossip and the Evolution of Language.* Harvard University Press, Cambridge, Mass. 1995

[14] Berger, C.R., and Knowlton, S.W. The Hierarchy Principle in Strategic Communication. *Communication Theory* v. 6, no. 2, pp. 111-142. 1996.

[15] Smith, M. Voices From the Well: The Logic of the Virtual Commons. *Sociology.* 1992.

[16] Latae, B. The Psychology of Social Impact. *American Psychologist.* V. 36 no. 4, pp. 357-360.

[17] Palme, J. *Electronic Mail.* 1995.

[18] Hiltz, S.R., and Turoff, M. *The Network Nation: Human Communication via Computer.* 1978

[19] Jones, Quentin and Rafaeli, Sheizaf. User Population and User Contributions to Virtual Publics: A Systems Model. *Proceedings of the ACM 1999 Conference on Groupware.*

[20] Korteum, G., Segall, Z. and Bauer M. Context Aware, Adaptive Wearable Computers as Remote Interfaces to Intelligent Environments. In *Second International Symposium on Wearable Computers.* 1998.

[21] Holt, Peter, Dubs, Shelli, Jones, Marlene and Greer, Jim. The State of Student Modelling. In *Student Modelling: the Key to Individualized Knowledge-based Instruction.* pp. 2-35: Springer-Verlag. 1992.

[22] Mitrovic, Antonija and Ohlsson, Stellan. Evaluation of a Constraint-Based Tutor for a Database Language. *International Journal of Artificial Intelligence in Education* V. 10, pp. 238-256. 1999.

[23] Ohlsson, S. System Hacking Meets Learning Theory: Reflections on the Goals and Standards of research in Artificial Intelligence and Education. *Journal of Artificial Intelligence in Education.* V. 3(4), pp. 429-447. 1991.

[24] Jameson, Anthony. Numerical Uncertainty Management in User and Student Modelling: An Overview of Systems and Issues. *User Modelling and User-Adapted Interaction.* V. 5(3), pp. 193-251. 1995.

[25] Russell, Stuart and Norvig, Peter. *Artificial Intelligence: A Modern Approach.* Prentice hall, Englewood Cliffs, New Jersey. 1995.

[26] Horvitz, Eric, Breese, J., Heckerman, D., Hovel, D. and Rommelese, K. The Lumire Project: Bayesian User Modelling for Inferring the Goals and Needs of Software Users. *Uncertainty in Artificial Intelligence: Proceedings of the Fourth Annual Conference.* 1998

[27] Billsus, D., and Pazzani, M.J. A Hybrid User Model for News Story Classification. /it UM99, User Modelling: Proceedings of the Seventh International Conference. 1999.

[28] Hoppe, Ulrich. The use of Multiple Student Modelling to Parameterize Group Learning. *Proceedings of the 7th World Conference on Artificial Intelligence in Education.* 1995.

[29] Brannick, Michael T., and Prince, Carolyn. An Overview of Team Performance Measurement. In *Team Performance Assessment and Measurement: Theory, Methods and Applications.* Michael T. Brannick, Eduarado Salas and Carolyn Prince Ed. 1995.

[30] Cannon-Bowers, Janice A. and Salas, Eduarado. A Framework for Developing Team Performance Measures in Training. In *Team Performance Assessment and Measurement: Theory, Methods and Applications.* Michael T. Brannick, Eduarado Salas and Carolyn Prince Ed. 1995.

[31] Bowers, Clint A., Braunt, Curt C., and Morgan Jr., Ben B.. Team Workload: Its Meaning and Measurement. In *Team Performance Assessment and Measurement: Theory, Methods and Applications.* Michael T. Brannick, Eduarado Salas and Carolyn Prince Ed. 1995.

[32] Hallam, Glen and Campbell, David. The Measurement of Team Performance with a Standardized Survey. In *Team Performance Assessment and Measurement: Theory, Methods and Applications.* Michael T. Brannick, Eduarado Salas and Carolyn Prince Ed. 1995.

[33] Senior, Barbara. Team Roles and Team Performance: Is there 'Really' a Link?. *Journal of Occupational and Organizational Psychology.* V. 70, pp. 241-258. 1997.

[34] Dickinson, Terry L., and McIntyre, Robert M.. A Conceptual Framework for Teamwork Measurement. In *Team Performance Assessment and Measurement: Theory, Methods and Applications.* Michael T. Brannick, Eduarado Salas and Carolyn Prince Ed. 1995.

[35] Kragier, Ken and Wenzel, Lucy H. Conceptual Development and Empirical Evaluation of Measures of Shared Mental Models as Indicators of Team Effectiveness. In *Team Performance Assessment and Measurement: Theory, Methods and Applications.* Michael T. Brannick, Eduarado Salas and Carolyn Prince Ed. 1995.

[36] Rentsch, J.R., Heffner, T.S. and Duffy, L.T.. Teamwork Schema Represenations: The Role of Team Experience. *Proceedings of the Eight Annual Conference of the Society for Industrial and Organizational Behaviour.* 1993.

[37] Goldsmith, T.E. and Kraiger, K. Applications of Structural Knowledge Assessment in Training Evaluation. In *Improving Training Effectiveness in Organizations.* J.K. Ford ed. 1997.

[38] Soller, Amy L. Supporting Social Interaction in an Intelligent Collaborative Learning System. *International Journal of Artificial Intelligence in Education.* V. 12. 2000.

[39] McManus, M. and Aiken, R. Monitoring Computer-based Problem Solving. *Journal of Artificial Intelligence in Education.* V. 6(4), pp. 307-336. 1995.

[40] Belbin, R. Meredith. *Management Teams.* John Wiley and Sons, New York. 1981

[41] Belbin, R. Meredith. *Team Roles at Work.* Butterworth-Heinemann, Oxford. 1993.

[42] McCann, R. and Margerison, C. Managing High-performance Teams. *Training and Development Journal.* V. 13, pp. 53-60. 1995.

[43] Parker, G.M. *Team Players and Teamwork: The New Competitive Business Strategies.* Jossey-Basses, Oxford. 1990.

[44] Dulewicz, Victor. A Validation of Belbin's Team Roles from 16PF and OPQ Using Bosses' Ratings of Competence. *Journal of Occupational and Organization Psychology.* V. 68, pp. 81-99. 1995.

[45] Furnham, Adrian, Steele, Howard and Pendleton, David. A Psychometric Assessment of the Belbin Team-Role Self-Perception Inventory. *Journal of Occupational and Organizational Psychology.* V. 66, pp. 245-257. 1993.

[46] Cattell, R.B., Eber H.W., and Tatsuoka, M.M. *Handbook for the 16PF. Institute for Personality and Ability Testing;* Champagne, Illinois, USA. 1970.

[47] Saville, P., Holdsworth, R., Nyfield, G., Cramp, L. and Mabey, W. *Occupational Personality Questionnaire Manual.* Esher, SHL. 1992.

[48] Pfeiffer, J. William, Heslin, Richard and Jones, John E. *Instrumentation in Human Relations Training: Second Edition.* University Associates, La Jolla, California, USA. 1976.

[49] Telsuk, Paul, Mathieu, John E., Zaccaro, Stephen J. and Marks, Michelle. Task and Aggregation Issues in the Analysis and Assessment of Team Performance. In *Team Performance Assessment and Measurement: Theory, Methods and*

Applications. Michael T. Brannick, Eduarado Salas and Carolyn Prince Ed. 1995.

[50] Clancey, William J. A Tutorial on Situated Learning. *Proceedings of the International Conference on Computers in Education.* 1995.

[51] McCann, R. and Margerison, C. Managing High-performance Teams. Training and Development Journal. V. 13, pp. 53-60. 1989.

[52] Fisher, S.G., Hunter, T.A., Macrosson, W.D.K. The Structure of Belbin's Team Roles. *Journal of Occupational and Organizational Psychology.* V. 71, pp. 283-288. 1998.

[53] Supnithi, Thepchai, Inaba, Akiki, Ikeda, Mitsuru, Toyoda, Jun'ichi, Mizoguichi, Riichiro. Learning Goal Ontology Supported by Learning Theories for Opportunistic Group Formation. *Proceedings of the 9th International Conference on A.I. In Education.* pp. 67-74. 1999.

[54] Vygotsky, L.S. *Mind in Society: The Development of the Higher Psychological Processes.* Harvard University Press: Cambridge, MA. 1978.

[55] Soller, Amy, Linton, Frank, Goodman, Brad and Lesgold, Alan. *Toward Intelligent Analysis and Support of Collaborative Learning Interaction.* Proceedings of the 9th International Conference on A.I. In Education. pp. 75-82. 1999.

[56] Quignard, Matthieu and Baker, Michael. *Favouring Modellable Computer-mediated Argumentative Dialog in Collaborative Problem-solving Situations.* Proceedings of the 9th International Conference on A.I. In Education. pp. 129-136. 1999.

[57] Baker, Michael, de Vries, Erica and Lund, Kristine. *Designing Computer-mediated Epistemic Interactions.* Proceedings of the 9th International Conference on A.I. In Education. pp. 139-146. 1999.

[58] Inaba, Akiko, Supnithi, Thepchai, Ikeda Mitsuru, Mizoguichi, Riichiro, Toyoda Jun'ichi *How can we Form Effective Collaborative Learning Groups?* Proceedings of ITS 2000. 2000.

[59] Mühlenbruck, Martin, Hoppe, Ulrich. *Computer Supported Interaction Analysis of Group Problem Solving.* Proceedings of CSCL 1999. pp. 398-405. 1999.

[60] Singley, Mark K., Fairweather, Peter G., Swerling, Steven. *Team Tutoring Systems: Reifying Roles in Problem Solving.* Proceedings of CSCL 1999. pp. 538-548. 1999.

[61] Soller, Amy and Lesgold, Alan *A Computational Approach to Analyzing Online Knowledge Sharing Interaction.* Proceedings of AIEd 2003. pp. 253-260. 2003.

[62] Wang, Xueyi, Bu, Jiagun, Chen, Chun *Achieving Undo in Bitmap-based Collaborative Graphics Editing Systems.* Proceedings of CSCW 2002, pp. 68-76. 2002

[63] Blanchard, Anita L. and Markus, Lynne M. *The Experienced 'Sense' of a Virtual Community: Characteristics and Processes.* ACM SIGMIS Database. Vol. 35 Issue 1. pp 64-79. 2004.

[64] Serra Jr., G., Coutinho, L., Labidi, S. *Formation of Groups in Cooperative Learning: a Genetic Algorithm Approach.* Proceedings of 2001 Conference on Computers and Advanced Technology in Education.

[65] Feinman, Alexander and Alterman, Richard *Discourse Analysis Techniques for Modelling Group Interaction.* Proceedings of 2003 Conference on User Modelling. pp. 228-237

[66] Jermann, P., Soller, A., and Lesgold, A. *Computer software support for CSCL.*. In P. Dillenbourg (Series Ed.) and J. W. Strijbos, P. A. Kirschner and R. L. Martens (Vol. Eds.), Computer-supported collaborative learning: Vol 3. What we know about CSCL ... and implementing it in higher education pp. 141-166. Boston, MA: Kluwer Academic Publishers.

Appendix A

Functional Skills Questionnaire

Student Number:

The following section is for students who were enrolled in CMPT 371.

Circle one answer.

What was the level of your Java knowledge at the start of CMPT 371?

Poor Average Good

What was the level of your knowledge of UML design at the start of CMPT 371?

Poor Average Good

What was the level of your knowledge of XML at the start of CMPT 371?

Poor Average Good

What was the level of your knowledge of HTML at the start of CMPT 371

Poor Average Good

Appendix B

Example Behavioural Summary Scale (Feedback)

Feedback Behaviour

Definition: Feedback behaviour refers to the ability of the members of a team to give effective evaluations of each other's performance and to receive and act upon such evaluations gracefully and usefully.

How would you rank your CMPT 371 team's performance regarding feedback behaviour?

5 The team provided positive, helpful and timely evaluations of each other's work and the evaluations were received gracefully when given.

4

3 The team provided reasonably helpful and constructive evaluations to each other, and the members were generally receptive and open concerning evaluations of their own work.

2

1 The team provided either no comments or unhelpful comments to each other. The team members were not open to any evaluation of criticism of their performance, not matter how helpful

Comments: