



MINT~RC

MANAGEMENT OF INNOVATION  
AND NEW TECHNOLOGY  
RESEARCH CENTRE

**ORGANIZATIONAL LEARNING VIA GROUPWARE:  
A PATH TO DISCOVERY OR DISASTER?**

by

Neel Chauhan  
Nick Bontis

Management of Innovation and New Technology  
Research Centre

WORKING PAPER NO. 102  
January 2001

Innis

HD

45

.W657

no.102

McMaster  
University

MICHAEL G. DEGROOTE  
SCHOOL OF BUSINESS



# Organizational Learning Via Groupware: A path to Discovery or Disaster?

By

Neel Chauhan, B.Comm., M.Sc.  
Strategy, Finance and Economics Knowledge Team  
Arthur Andersen Business Consulting  
London, UK  
[neel.chauhan@uk.arthurandersen.com](mailto:neel.chauhan@uk.arthurandersen.com)

and

DR. NICK BONTIS  
ASSISTANT PROFESSOR, STRATEGIC MANAGEMENT  
DeGroote School of Business, McMaster University  
1280 Main Street West, MGD #207  
Hamilton, Ontario Canada L8S 4M4  
Tel: (905) 525-9140 x23918 Fax: (905) 521-8995  
[nbontis@mcmaster.ca](mailto:nbontis@mcmaster.ca)  
[www.Bontis.com](http://www.Bontis.com)

Copyright © 2001. Chauhan and Bontis. All rights reserved. Version 12-Jan-01.

This paper is open for comment.

No part may be reproduced without the permission of the authors.  
Please forward correspondence directly to Dr. Nick Bontis.

**ORGANIZATIONAL LEARNING VIA GROUPWARE:  
A PATH TO DISCOVERY OR DISASTER?**

by

Neel Chauhan  
Nick Bontis

Management of Innovation and New Technology  
Research Centre

WORKING PAPER NO. 102  
January 2001

The Working Paper series is intended as a means whereby a researcher may communicate his or her thoughts and findings to interested readers for their comments. The paper should be considered preliminary in nature and may require substantial revision. Accordingly, this Working Paper should not be quoted nor the data referred to without the written consent of the author. Your comments and suggestions are welcome and should be directed to the author.

McMASTER UNIVERSITY LIBRARY

## Organizational Learning Via Groupware: A path to Discovery or Disaster?

**ABSTRACT.** Due to changes ranging from the globalization of markets, to the accelerated improvements of technology, and from the tacit skills of knowledge workers, to the ubiquitous access we have to information, organizations must be able to learn efficiently and effectively in order to survive in today's increasingly fierce competitive environment. As a result of this need, the academic field of organizational learning has prospered (Crossan and Guatto, 1996). The study of organizational learning is important because it provides academics and professionals with more insight into the system, processes and behaviours that enable organizations to generate new knowledge and adapt to change (Crossan, Lane and White, 1999). Organizational learning also enables companies to build capabilities, and intellectual capital, which can provide the firm with the ability to manage intangible resources for a sustainable competitive advantage (Bontis, 1999; Bontis et al., 1999).

In order for organizations to keep up with the onslaught of challenges inherent in the new century they must continuously learn. From a normative perspective, groupware technologies facilitate organizational learning by providing a means of disseminating and codifying organizational knowledge. However, in practice the potential learning benefits of groupware are rarely achieved. This paper aims to contribute to the fields of organizational learning and information systems by examining possible reasons for groupware failures, which lead to lost potential learning benefits and thus unsustainable advantage. It is now common for computer applications to offer intangible, strategic benefits to organizations by supporting organizational learning behaviours (Venkatraman, 1991). From a normative perspective, groupware applications facilitate organizational learning by supporting collaboration amongst organizational group members. However, in practice, the full learning benefits of using groupware are rarely achieved because these applications are often resisted (Ciborra, 1996; Neilson, 1997; Orlikowski, 1996).

For this study, groupware was examined using the push-pull theory of implementation, drawn from the disciplines of engineering and marketing (Zmud, 1984). The main research proposition of this paper is that groupware applications that have been introduced as a result of a need (a pull strategy) are more successful than those that are forced onto organizational members by senior management (a push strategy). Research was conducted in the United Kingdom at a global information provider (InfoProv). Semi-structured interviews were conducted with a project coordinator, quality manager, user and project manager. The first two interviews focused on Teamlinks, which is an organization-wide groupware application similar to Lotus Notes. The latter two interviews focused on a Problem Management System (PMS), which is a knowledge database that is being rolled out on a global scale. Results indicate that there is a general lack of emphasis on pull strategies in groupware implementations, and that this may be contributing to their high failure rates. However, given the qualitative nature of this study, more empirical research is needed to confirm the results. Accordingly, guidelines for further research are suggested.

### Neel Chauhan Bio

Neel Chauhan is a senior analyst in the Strategy, Finance and Economics team of Arthur Andersen's Business Consulting practice in London, United Kingdom. He has worked on a variety of new economy projects ranging from eBusiness strategy to start-up business planning. Prior to joining Andersen, Neel was an independent consultant in the entertainment industry where he crafted and executed marketing plans with companies including Calvin Klein, Guess and Warner Bros. Neel has published a number of articles in Business Sense Magazine. He holds a Master of Science degree from the London School of Economics and a Bachelor of Commerce degree from Ryerson Polytechnic University.

### Dr. Nick Bontis Bio

Dr. Bontis is currently Assistant Professor of Strategic Management at the Michael G. DeGroote School of Business, McMaster University. He is also Director of the Institute for Intellectual Capital Research Inc., Chief Knowledge Officer of Knexa.com – the world's first knowledge and exchange auction and Associate Editor of the Journal of Intellectual Capital. Dr. Bontis completed his doctoral education at the Ivey School of Business, University of Western Ontario. Professor Bontis has won international acclaim for many of his academic research papers, book chapters and business management cases. FORTUNE magazine recently recognized him as "one of the world's real intellectual capital experts".

## Introduction

Due to changes ranging from globalization to improvements in technology, to the advent of the knowledge era, organizations must be able to learn efficiently and effectively in order to survive in today's increasingly fierce competitive environment (Bontis and Crossan, 2001, Crossan et al., 1999, Crossan *et al* 1995; Mumford 1992). Within this setting the field of organizational learning has thrived (Easterby-Smith Et al 1998). The study of organizational learning (OL) is important because it provides academics and professionals with more insight into the system that enables organizations to generate knowledge and adapt to change. Nonaka (1991) summarizes the essence and importance of knowledge generation, which is a critical component of OL:

When markets shift, technologies proliferate, competitors multiply, and products become obsolete almost overnight, successful companies are those that consistently create new knowledge, disseminate it widely throughout the organization, and quickly embody it in new technologies and products. These activities define the 'knowledge-creating' company whose sole business is continuous innovation (Nonaka 1991 p. 96).

Organizational learning is important because, among other things, it enables companies to build capabilities and intellectual resources, which can provide the firm with a sustainable competitive advantage (Ciborra and Andreu 1998). It is now common for computer applications to offer much more intangible, strategic benefits such as supporting OL (Venkatraman 1991). A group of applications that provide such benefits are groupware. From a normative perspective, groupware applications facilitate organizational learning by supporting collaboration amongst organizational group members (Bikson 1996). However, in practice, the full learning benefits of using groupware are rarely achieved because these applications are often resisted (Bowers 1995; Ciborra 1996; Neilson 1998; Orlikowski 1996).

This study takes some initial steps to explore possible reasons for groupware implementation failures, which prevent organizations from gleaning full learning benefits from such applications. We draw on the push-pull theory of implementation, from the disciplines of engineering and marketing, in an attempt to gain insight into reasons for the lack of success with these collaborative applications. Specifically, we argue that groupware applications that have been introduced as a result of a need (a pull strategy) are more successful. We draw on findings from research conducted at InfoProv, a major global information provider.

The following two sections review the organizational learning and groupware literatures respectively.

## Organizational Learning Literature Review

Fiol and Lyles (1985 p. 804) define organizational learning as "...the process of improving actions through better knowledge and understanding." In the words of Garvin (1983 pp.80), "a learning organization is an organization skilled in creating, acquiring and transferring knowledge, and at modifying its behavior to reflect new knowledge and insights...". While some elements of the definitions of learning are similar, they tend to differ based on their assumptions of levels of analysis, cognition and the relation of learning and performance (Bontis and Crossan, 2001). Neilson (1997 pp.2) defines learning as "leveraging the collective intellectual material of an organization to produce a higher level organizational asset... ". Argyris (1977) defines OL as a process of detecting and correcting error. Another way to understand OL is as a level of learning that is something greater than group or individual levels of learning. This definition implies that OL is something more than the sum of the learning of the individuals that it is made up of (Crossan et al 1995). Most of these definitions imply that OL is a process, as opposed to a program, which involves some sort of change in behavior. Huber (1991) recognizes that learning does not necessarily result in changed behavior as some academics have suggested (Cangelosi and Dill 1965; Garvin 1993; Nevis et al 1995). This is evident in his definition of learning which states that "an entity learns if, through its processing of information, the range of its potential behaviors is changed (Huber 1991). For the purposes of this paper we adopt Fiol and Lyles' aforementioned classic definition, while recognizing the differences between organizational, group and individual levels of learning as per Bontis and Crossan (2001).

There are several key conceptualizations within the OL literature that serve as a platform for gaining a better understanding of how groupware applications can support learning.

### *Cognition*

A number of learning theorists have pointed out that there are two main views of learning, which are not necessarily mutually exclusive. These have been identified in the literature as cognitive and behavioral views (Fiol and Lyles 1985; Argyris 1977; Cangelosi and Dill 1965; Huber 1991; Crossan et al 1995). Crossan et al (1995 pp.348) make the distinction between these two views eloquently:

Cognitive theorists would assume learning has occurred if there is a change in thought processes (unobservable), even in the absence of adjusted behavior (observable). Behaviorists assume that learning has occurred if there is a noticeable change in behavior, even if not preceded by a change in thinking...

Beyond changes in behavior, it has been argued that organizational learning should result in some sort of improvement in performance (Garvin 1993; Fiol and Lyles 1985; Cangelosi and Dill 1965). However, there is no evidence to suggest that there is indeed a connection between learning and performance (Crossan et al 1995). One problem with making a distinction between cognitive and behavioral learning is that it assumes that it is possible to recognize or measure behavioral changes that result from learning.

### *Double Loop Learning Theory*

Argyris (1977) argues that organizational learning is a process of detecting and correcting errors. He identifies two types of learning, which he calls single loop and double loop. Single loop learning involves a simple change. It can be seen as a simple error detection and correction. Argyris (1977) gives the example of a thermostat turning on to adjust the room temperature to a predetermined heat. Double loop learning is a more involved than single loop learning, in that it can involve a change in the organizations methods or policies. Argyris (1977) gives the example of questioning the appropriateness of the current method of heating employed by an organization. A double loop approach to a heating “error” would be to question whether or not a heat pump could do a better job than the existing furnace. Concepts similar to single and double loop learning are present in the literature. For example, Fiol and Lyles (1985) refer to single loop learning as lower-level learning and Kim (1993) refers to it as operational learning. Argyris’ (1977) theory is lacking in a sense that it does not provide us with any insight into when to use each of the type of learning. Neilson (1997) also points out that Argyris’ theories do not provide practitioners with any guidelines to identify learning problems or errors.

### *Modes of Learning*

Miller (1996) identified and defined different modes of organizational learning. Learning takes place at different levels within these modes, which are analytical, synthetic, experimental, interactive, structural, and institutional. Analytical learning is characterized by informational gathering, rational analysis, selection of an “optimal” alternative, and evaluation of performance. Synthesis usually involves a creative individual looking at a situation, often by examining texts, in a different way. The result of this different view can be new goals, procedures or decisions. Experimental learning involves performing experiments and monitoring the results. Interactive learning involves learning by doing, but in a less systematic way than with experimental learning. Structural learning refers to learning from the routines of an organization. These structured routines are codified lessons from previous analytical learning. Finally, Miller (1996, pp.495) defines Institutional learning as “...an emergent inductive process by which organizations assimilate values, ideologies and practices...”. Miller (1996) points out that learning at this

level may involve indoctrination, socialization, and coercion and as a result does not allow much room for voluntarism.

### *Constructs of Learning*

Huber (1991) breaks the broad term of organizational learning down into four constructs, namely, knowledge acquisition, information distribution, information interpretation and organizational memory. Knowledge acquisition is defined as the process in which knowledge is acquired or obtained. Information distribution is the process by which information is shared. Information interpretation involves attempting to develop one or more common interpretations of distributed knowledge. Organizational memory is defined as knowledge that is stored for use.

### *Measurement*

A number of authors have stressed the importance of measuring learning (Bontis and Crossan, 2001; Crossan et al., 1999, Crossan et al., 1995; Garvin 1993; Bontis 1996). Garvin (1993) stressed that it is necessary to use a number of factors to measure learning benefits above and beyond the commonly used measures of price or cost improvements. Other factors that should be measured include quality, delivery and new product introductions (Garvin 1993). However, while measuring learning is important it is easier said than done.

### *Order from Chaos*

Nonaka (1988) argues that organizations that stay in a steady state (i.e. do not change) do not learn. He describes how organizations can generate knowledge by continually being in a state of non-equilibrium, which means identifying and considering a variety of options (never having a fixed course of action or way of organizing). This opens up a company's eyes to new points of view, which could be better, more profitable, ways of organizing (Nonaka 1988).

### *Bounded Rationality*

Herbert A. Simon (1991) recognizes the importance of understanding the intricacies of OL. In particular, he stresses understanding not just knowing or learning. Simon (1991) notes that knowledge is stored in two places, people's heads or in files (data banks). As a result, maintaining knowledge in organizations depends on codification and dissemination of knowledge of individuals. The knowledge that an organization has at any given time can be looked at like its memory and has been described by Bontis (1996) as stocks of knowledge.

Many of the aforementioned conceptualizations have provided normative sequential models of organizational learning. It is unlikely that learning takes place in such a linear fashion (Nevis et al 1995). In fact most learning in organizations likely takes place in a more natural serendipitous fashion. This is part of a broader problem of oversimplifying the concept of organizational learning (Easterby-Smith et al 1998).

In both the learning literature and the groupware literature there have been legitimate complaints about a lack of prescriptions for applying normative academic theories. A further problem with organizational learning is that it is difficult to separate individual learning from its social context (Magalhaes 1998). Therefore deconstructing learning into different levels is problematic. Much of the literature implies that learning is invariably an intentional process (Argyris 1995; Fiol and Lyles 1985). Huber (1991) goes to the extent of claiming that organizational effectiveness must be improved in order for learning to occur. This is not realistic, as it neglects learning that takes place serendipitously which can be a vital source of learning and competitive advantage (Ciborra 1993; Mintzberg 1979; Nevis et al 1995; Nonaka 1991). The following section will build on OL theory by demonstrating how, from a normative perspective, groupware facilitates OL.

## **Groupware Literature Review**

Khoshafian and Buckiewicz (1995) define groupware as “any computer system which enables groups of workers to collaborate for some common purpose or task...[and] ...any system that helps collaborative and cooperative work...”. The Oxford Dictionary of Computing defines groupware as “software that supports the cooperation of a group of people, usually working in a distributed environment”. Groupware can also be defined as “ a set of tools to help groups work more effectively” (Neilson 1977 pp. 3). Groupware is a very general concept that encompasses many different applications and technologies (Khoshafian and Buckiewicz 1995). An example of a common groupware software package is Lotus Notes. Lotus Notes is a system for organizing and distributing unstructured information (Alsop 1997). Some common features on notes include bulletin boards, which enable users to interact and comment on issues in a structured format (Ciborra and Patriotta 1996; Grudin 1994); document sharing; news groups and electronic mail (Fallia 1996). Electronic mail (e-mail) is one of the most widely used groupware functions (Bowers 1995; Fallia 1996; Neilson 1997; Orlikowski 1996). This feature enables a number of useful collaborative functions such as the ability to communicate in real-time, ability to send notes to predetermined groups and the ability to file communication in a structured form (Fallia 1996).

Other well-known groupware applications, are teleconferencing, video conferencing and e-mail conferencing. The latter of which is essentially an e-mail distribution lists. Forums, another groupware function, allows for broadcasting of communications and dissemination of information at a group level (Failla 1996). It is usually some sort of shared file that allows access at different levels to different people. Some users may have permission to append and/or post information. The key benefit is that information can broadcast without duplication (Fallia 1996). Groupware functions enable one of the most important parts of organizational learning, which is the transfer of acquired knowledge from one part of an organization to others (Kock and McQueen 1998). According to Huber's (1991) model, groupware enables the second phase, among others, of learning, which is information distribution.

Groupware applications allow individuals to codify communication and make it visible to the organization. This reduces barriers to communication and facilitates sharing organizational knowledge (Ciborra and Patriotta 1996). Groupware technologies also facilitate the learning process by helping groups communicate and share work practices (Andreu and Ciborra 1998).

Groupware applications are relevant to learning in a sense that they are a means of connecting members of an organization. This includes members from different units and members within a given unit. Huber (1991) supports the use of groupware indirectly by pointing out that communication and knowledge sharing between units and within units is of paramount importance to organizational learning. In order for organizations to learn, there must be information to share and a means to share or disseminate information (Huber 1991).

Groupware applications enable a great deal of organizational knowledge to be codified. "If knowledge is power, then groupware empowers workers by providing an avenue for creativity, access to huge warehouses of information, and, most importantly, a means for collaboratively working with their peers" (Khoshafian and Buckiewicz 1995 p. 10). It has also been argued that communication and document routing functions in groupware can enable the same amount of work to be accomplished in less time, as measured by the number of meetings necessary to complete a task (Huber 1990). This can be seen as increasing learning efficiency, if it is assumed that knowledge is transferred in meetings. Huber (1990) also posits that electronic bulletin board functions effectively disseminate information about the organization's current problems, policies, and priorities. This permits managers to learn quickly, and use this knowledge to make better decisions. This in turn can lead to the desirable outcomes that many academics define as signs of learning (Huber 1990). While, these claims sound encouraging, the author does not provide any empirical evidence to support them.

Ciborra and Andeu (1998) provide an example of a collaborative system implemented at Mrs. Fields cookies. This system enables the head office to communicate routines and knowledge with stores. The learning benefit for the firm was the development and dissemination of core capabilities (Ciborra and Andeu 1998). Another key aspect of groupware is that people can work together who are located in different geographic regions. Groupware enables learning by facilitating shared work practices and communication within groups. Kock and McQueen (1998) conducted research on how the e-mail conferencing (EC) features on Novell Groupwise were able to support knowledge dissemination in organizations. They showed that an electronic conferencing function can facilitate dissemination of knowledge between functional departments. For a full normative review of the potential of groupware to support learning see Chauhan (1998).

### **Problems with Groupware: A behavioural perspective**

Groupware has not been a flawless organization learning solution. Even if the infrastructure to share knowledge is in place, in this case that being the groupware, there is no guarantee that people will use it. The “field of dreams” approach to groupware systems implementation – “if you build it, they will use it” – does not work unless there is a demonstrated need to intensively share information (Orlikowski 1992). It is up to management to convey this need to all users. This is known as a push strategy.

Empirical research has shown that in some circumstances, users of groupware use substitute applications such as fax machines, corporate e-mail, telephones, private databases and the internet (Ciborra 1996). This is attributed to many factors including substitutes being better known, user friendly and robust (Ciborra 1996). Ciborra (1996) also notes that it is only in rare situations where an application is the only reasonable alternative available to accomplish a task, that a system will be used without question (Ciborra 1996 pp.8). By examining a number of cases Ciborra (1996) found that it was rare for groupware to enable any sort of radical learning, such as double-loop learning (Argyris 1977). Users often lack the resources (time, training, expertise) and incentives to share knowledge, and consequently to use groupware (Ciborra 1996). Ciborra goes on to argue that users often do not perceive the application (in this case, Lotus Notes) to be anything more than an e-mail program or database, rather than a group learning tool. Such factors result in groupware failures, which prevent organizations from gleaning the learning benefits from such applications.

Some applications require a large percentage of people to use the system. Critical mass of users is often a requirement to glean the full benefits from groupware (Bowers 1995; Buchanan 1996; Grudin 1994).

Bowers (1995) provides an example of a scheduling function that required the majority of people to use it, in order for it to function successfully. This situation seems to be that of a chicken and egg, in that you need everyone to use the function for it to work, yet people will not use the function unless it works. Other reasons why this particular groupware application did not achieve its full potential were a lack of perceived benefits from switching to the new work practices, reluctance to share documents, creation of extra work, undesired visibility of document processes, and cultural incompatibility (Bowers 1995). Users were reluctant to try using new features, and with the exception of e-mail, the collaborative aspects of the system were a failure.

The IPM system at Unilever is a good example of how groupware applications, in this case Lotus Notes, do not effectively codify or externalize previously non-codified practices and knowledge. At Unilever, employees had problems searching for past product innovation information because finding it was contingent on development teams entering detailed reports, which were not always entered, and being able to identify the correct words to search (Ciborra and Patrioa 1996). In some cases employees may have legitimate reasons for not wanting knowledge to be codified and subsequently shared. This may be because they do not want to be evaluated based on their work-in-progress or because they feel their knowledge provides them with some sort of power. In such cases, employees have no reason to use the full functionality of groupware applications (Ciborra 1996b).

Some empirical work was conducted in a software development groupware at IBM, which found that there was a complete lack of awareness for the fact that a particular application even supported the work of groups (Fallia 1996). Fallia (1996) also found that some employees did not understand the full potential of what groupware applications could accomplish. In such cases management has not done its job to make employees aware of the work tools available to them.

It is possible that the reason why organizations can't glean the full benefits from various levels of learning is that the organization may not be capable of using new generated knowledge. Bontis (1999) provides the example of an individual undergoing training and building skills that cannot be applied in organizations. Conflict or frustration can exist because individuals, or groups, may desire to better themselves beyond what is necessary for achieving the organization's objectives. In the same way, much of the knowledge that is disseminated and generated using groupware may not be beneficial. This indicates that groupware applications may be failing not because they do not work, but because the knowledge they generate is not needed and as a result they are not used.

It has been established in a number of cases that groupware systems often impose ways of working on organizations (Ciborra and Patriotta 1996). Therefore, as opposed to these systems supporting existing processes, by facilitating learning, they end up imposing new processes. This brings about the question, of whether the potential improvements in learning come from the new system or the new work processes imposed by the system. A further problem with groupware is that the person who puts in additional work to use the tool or update it, does not always gain from taking on additional work. Such situations are conducive to resistance of applications (Fallia 1996; Grudin 1994; Orlikowski 1996).

In practice Lotus Notes, unless properly used, is merely a glorified e-mail program (Alsop 1997). The same holds true for other groupware applications in that the benefits often cannot be gleaned unless they are implemented and used correctly.

## **Research Question**

This study seeks to gain a better understanding of the affects of push/pull implementation strategies on successful groupware usage. This is relevant to learning because any improvements in groupware usage will result in improvements in learning benefits. Drawing on Zmud's (1984) argument, groupware applications that are introduced as a result of a need (pull) are more likely to succeed than those that are forced (push) onto the organization. Success in this study is defined as providing an organization with additional learning benefits (i.e., knowledge generation). The main hypotheses is reiterated as follows:

**Hypothesis: Groupware applications that are need-pulled are more likely to succeed (provide learning benefits) than applications that are only technology-pushed.**

## **Data Collection**

Semi-structured interviews were used to conduct the research. This method was selected in an attempt to gain additional richness of findings beyond what would be possible with a structured survey, or interview. While fixed-open-questions were set, the approach was flexible, leaving room for addition and omission of questions. The initial survey was pilot tested on several students to find areas where the clarity of the questions could be improved in order to alleviate problems with understanding the questions.

In addition to the interviews, manuals of the Teamlinks systems at InfoProv were reviewed. A number of the groupware functions on the systems including the scheduler, intranet, document manager and news

functions were tested. In addition, a demonstration of the Problem Management System (PMS) at InfoProv was reviewed.

The research was conducted in the United Kingdom at a global information provider referred to as InfoProv. Four semi-structured interviews were conducted. The first two focused on Teamlinks, which is an organization-wide groupware application similar to Lotus Notes. The latter two interviews focused on a Problem Management System (PMS), which is a knowledge database that is being rolled out on a global scale. The interviews ranged in length from forty minutes to one hour. The interviews were with a project coordinator, quality manager, user and project manager. The findings are arranged in two groups. The first includes information from the first two interviews, which dealt primarily with the Teamlinks applications. The second set of findings includes information from the latter two interviews, which dealt with the PMS.

### ***Findings: Teamlinks Application***

Teamlinks is InfoProv's e-mail and group collaboration application. Its features include a scheduler, directory and e-mail. Other groupware applications at the company users included video conferencing, teleconferencing, document flow functions, and a company intranet with collaborative features.

### ***Push Findings***

These applications are pushed through the company's technical policy. This policy dictates the software that will be installed on new PC's. This is a more subtle push, in that it only dictates that the application is made available. Training is provided to introduce new employees to the groupware applications. This was backed up with a manual for Teamlinks and Calendar Manager.

The basic e-mail functions were widely used. So much so that the second respondent, a quality manager, referred to them as "institutionalized". As a result of this, it was not necessary for management to push the e-mail function, because it was a necessary part of working for InfoProv. This shifted the duties of the technology push from the management level to lower levels. Employees knew that they needed to use e-mail in order to function in their teams.

Many of the functions on Teamlinks were not as widely used as e-mail. In the case of the scheduling function, calendar manager, management attempted to alleviate this problem by sending out electronic reminders to use functions that were not being used. The second respondent spoke of a number of cases he had experienced where his colleagues were not even aware of the basic functionality of the scheduling

function. In particular, some people were not aware that they could access other people's schedules and make appointments with them and that they could specify the room that they would use for a meeting. Despite management's efforts, it was indicated that the application was still not used by some. This created problems because without the critical mass of users these functions cannot be properly used. The calendar manager functions were not used despite the fact that it is "company standard" to use them.

### *Pull Findings*

The people interviewed did not have information on the perceived need for Teamlinks when it was first introduced, almost ten years ago. As one would expect, their knowledge was more in the area of current usage issues. This made it difficult to find out information on whether or not the application was implemented because of a need (pull) and/or because it was a new technology (push). This same problem was encountered when trying to examine the pull issues of several groupware cases studies.

There were a few methods that were available to examine employee's perceived need for groupware applications. One was whether or not they chose to adopt applications that were not absolutely necessary to use. An example was the choice not to use a scheduling function. However, it was not clear whether this lack of use was due to lack of awareness, know-how or need. In other cases, some groups of people opted to use substitute applications, such as Microsoft Outlook. This indicates that they saw a need for the generic function, a scheduler, but felt that another application could serve their purpose better.

### ***Findings: Problem Management System***

InfoProv is currently in the process of rolling out, on a global scale, a Problem Management System (PMS). This system tracks problems and solutions for company software. It enables the product support team to efficiently codify solutions so they can be looked up to avoid having to solve the same problem twice. It also offers a structured, efficient method of storing problem records. Many of the records kept on the database are technical in their nature. Records include code and methods to assist with problems developing software. The first interview was conducted with a user of system and the second with a project manager who was responsible for rolling out the system in the Numeric Database Technology Group (NDTC).

### *Push Findings*

PMS is being rolled out with predominantly a push strategy. In fact, once the application is rolled out in each department, people will not have any substitute applications. They will effectively be forced to use PMS, assuming they will be using a problem management database.

A project manager was in charge of rolling out the system. The user mentioned that "it [the project manager] was a key element of driving the project". Another major push element of the implementation strategy was training and manuals that were provided to new users.

PMS has the same problem as Teamlinks with critical mass. If all users are not on the same system, it creates more work in situations where the people not on the system need to be notified about a problem. Service Level Agreements, which dictate which applications should be used for communication, were set up to help support homogeneity in terms of application use.

The current implementation objective, in the NDTG, is to get the application installed on people's PC's. After this is accomplished the project manager will specify the precise methods and standards for using the application. He recognized that there would be some problems with getting people to use compatible configurations, but his present concern is getting the application rolled out. In the process of doing this, people are not sold on the full functionality of the new system, they are sold on the ease of switching to the new system from the old one.

#### *Pull Findings*

Prior to using PMS, people would use Teamlinks to send development related problems to those who were thought to have answers. Management thought that this was inefficient and accordingly it was suggested that some sort of a PMS be implemented. It was introduced in a number of pilot sights and then decided that it would be implemented on a global scale.

The project manager mentioned that the NDTG was in favour of using the PMS because they liked the benefits. However, it was found that another group, the Development Group, did not see the benefits of using PMS and would have preferred to use an alternative system called Observation Reporter (OR).

#### *Secondary Research*

Four cases from the groupware literature were also reviewed in an attempt to find evidence of push or pull strategies. Information was also collected on indications of the outcome of the systems. While some systems evidently had elements of a pull strategy, it was difficult to piece together the whole picture from cases that had not intentionally collected information on push-pull strategies. The findings are presented in the table below.

---

**Table 1: Case Study Analysis Findings**

<b>CASE / Application</b>	<b>Evidence of “Push” strategies</b>	<b>Evidence of “pull” strategies</b>	<b>Outcome</b>
<p>Zeta (Orlikowski 1996)</p> <p>Incident Tracking Support System</p>	<ul style="list-style-type: none"> <li>• training</li> <li>• ITSS was accessible to all members of the customer service department</li> <li>• users had to use the system to do their jobs</li> <li>• no substitutes</li> </ul>	<ul style="list-style-type: none"> <li>• installed system based on perceived need to replace the old system</li> <li>• system was approved after a pilot test to ensure it would meet users needs</li> </ul>	<ul style="list-style-type: none"> <li>• proactive collaboration amongst service representatives</li> <li>• learning benefits from within each division and within different divisions</li> </ul>
<p>World Bank (Bikson 1996)</p> <p>GroupSystem Technology (group decision support)</p>	<ul style="list-style-type: none"> <li>• system champion to ensure successful implementation</li> </ul>	<ul style="list-style-type: none"> <li>• assessment of users needs by doing research (primary and secondary)</li> </ul>	<ul style="list-style-type: none"> <li>• majority of users felt the system contribution to the learning and collaboration objectives.</li> <li>• the system was used by most participants</li> </ul>
<p>Unilever (Ciborra &amp; Patriotta 1996)</p> <p>Lotus Notes based Innovation process management system</p>	<ul style="list-style-type: none"> <li>• employees required to input all information on the development of a new products into the IPM system.</li> <li>• Desire to implement the system quickly and have fast visible results. Mainly for political. As a result the system was pushed on to people before they really understood it.</li> <li>• Technical training was provided; but no training was provided on the organizational implications.</li> </ul>	<ul style="list-style-type: none"> <li>• groupware in line with info culture will be pulled.</li> <li>• used user feedback about technical aspects of the system to improve technical functionality</li> </ul>	<ul style="list-style-type: none"> <li>• resistance: partially attributed to miss match between existing practices and those dictated by the structure of the application</li> <li>• people did not like having transparent WIP</li> <li>• duplication: employees had to do extra work to input material into the system (to meet the codification requirements).</li> </ul>
<p>IBM (Failla 1996)</p> <p>Notes style.. Video &amp; teleconferencing</p>	<ul style="list-style-type: none"> <li>• team achievement award (indirect)</li> <li>• other team-based performance compensation</li> </ul>	<ul style="list-style-type: none"> <li>• New matrix structure increased the need for coordination</li> <li>• amalgamation of products need for group coordination</li> </ul>	<ul style="list-style-type: none"> <li>• many applications used but problems emerged: 1) lack of perceived benefits 2) people not aware applications supposed to support teamwork 3) people not aware of potential</li> </ul>

## Analysis

This research assumes that if groupware applications were used successfully then learning would invariably occur. It should also be noted that even if groupware is implemented successfully learning does not always occur. Huber (1991) argued persuasively that more organizational learning occurs when more of an organization's members obtain knowledge and recognize it as useful. Groupware undoubtedly can facilitate efficient dissemination of knowledge; however, it does not necessarily aid people in recognizing the use of new knowledge (Chauhan 1998). More emphasis on pull strategies could help alleviate this problem. If members recognize the need for an application, and the information it will provide, then it is likely that they will be more likely to recognize the information as useful, and learn from it.

The problem of the lack of utilization of the full functionality of groupware also stems from people not recognizing such applications as being beneficial or needed (lack of a pull strategy). This highlights the fact that new strategies are needed to help users recognize the benefits (need) of using groupware to facilitate learning. Users also need to recognize the importance of the knowledge they are gaining through groupware so that they take advantage of it by applying it, in some way, to help the business. Further exploration of push/pull strategies (and combinations of both) may be useful in improving the current situation.

The findings indicate that push strategies are more prevalent than pull strategies. However, this is not to say that pull strategies are not as prevalent. This could be the case, but it appears that they are more difficult to research as the people who would be aware of pull strategies are those who initially implement a system. In the research push strategies were more prevalent both the empirical work and the review of cases (see table 1). While the results indicate that more emphasis on pull strategies could improve groupware success, they are in no way representative or conclusive.

### *Push Issues*

The results indicate that there is no shortage of push tactics in the implementation of groupware applications. A push technique that which often comes up in the literature, is that of providing users with incentives to use a system (Ackerman 1990; Orlikowski 1996; Ciborra 1996; Neilson 1998). An example of incentives facilitating the successful use of a system can be found in Ackerman's (1990) Answer Garden case, where contributors and users of the system were able to reduce their workload by using the system. It was found that applications and features that provided users with a perceived benefit were more successful. For example, e-mail was perceived to be easy to use, efficient and widely used, while the

scheduling function on teamlinks was perceived to be difficult to use and it was not widely accepted. E-mail was established to the point that it became “institutionalized”. As Crossan *et al* (1995) pointed out, organizations can institutionalize knowledge or practices that become of part of the organization, as opposed to only be embedded within individuals. This seems to be the case with the e-mail function on Teamlinks, at InfoProv. What is interesting is that it is difficult to separate the push from the pull. For example when the Teamlinks system was initially implemented there people must have had some good reasons for using (pulling) the e-mail function. Some of these would have likely been faster communication and efficient distribution of documents. However, it seems that after the e-mail function was “institutionalized”. Using the e-mail function necessary because of the large of percentage of people that depended on electronic communication. As a result, people came to depend on using e-mail. This was confirmed by Neilson (1998) who noted that peer pressure is a motivating factor for using groupware systems. This shows how a pull strategy or reason for use can easily turn into a push strategy.

A push technique that is commonly advocated in groupware literature is support from top management (Buchanan 1998). However, management support is a necessary, but not sufficient, condition for application success (Buchanan 1998). InfoProv’s project managers were perceived to play an important role in successfully implementing the PMS. However, the project manager does not always have the long term best interests of the organization in mind. The findings indicated that the project manager at InfoProv was making compromises in order to get the PMS rolled out. An example of this was the fact that the manager did not worry about how people would actually use the function – he was initially only concerned with getting the application rolled out. Again, this indicates a lack of balance between push-pull strategies. In this case there was too much focus on the push, which could be detrimental to the success of the application. In other words, the need to roll out the application quickly (technology push) should be balanced with the immediate and long-term needs of the users. It is possible that if these needs are not taken into consideration, the application will not be used, as was the case with the scheduling function that Bowers (1995) describes. In the case of PMS at InfoProv, the project manager is going to wait until after the implementation to find a way to coordinate people’s different tastes in terms of settings on the PMS. If no agreement is reached in terms of the settings, it could lead to the failure of the PMS. A little more emphasis on user needs (pull strategies) could alleviate such a problem.

It has been noted in the literature (Bikson 1996) that push techniques can increase groupware implementation success. A standard push technique that was found in most of the literature and cases was training. Bikson (1996 pp.161) noted that training was viewed by the implementation team at the World Bank as being “a critical factor in deploying the technology successfully”. Orlikowski, Grudin and

Neilson have also advocated the use of training to increase the success of groupware applications (Neilson 1998). Neilson goes on to explain how training leads to a perceived ease-of-use of collaborative applications, which contributes to user satisfaction.

Despite the strong presence of push strategies, groupware was found to be far from flawless. This was shown in the literature (see table 1) and with the Teamlinks systems at InfoProv. Despite the presence of extensive push strategies to implement the Teamlinks package, most of the functionality was not used. The fact that critical mass was not achieved with the calendar manager, the scheduling function, is evidence that the training, manuals, e-mail reminders and company standards, were not enough to ensure functions were used. The missing link in the case of the scheduling function could have been the lack of a pull strategy. It is possible that the lack of user support for the need to schedule electronically is a stronger force than all the push elements combined. There was a similar situation with the PMS. It had an extensive push strategy, but there was no evidence of a pull. The same push elements were used, training, project management, manuals etc. Yet the full functionality of PMS was not utilized. This indicates that a push is simply not enough to insure the success of an application. Zumd's (1989) theory states that technology implementation-success can be maximized by combining push and pull strategies. If this is true then InfoProv can improve its success by being more aware of pull strategies of implementation.

### *Pull Issues*

Pull strategies have been around for as long as information systems. In fact, the whole point of implementing an information system is often to meet a need. One of the techniques that is used to understand a social context, and therefore user needs, is ethnography (Bowers 1995). However, a need-pull is difficult to define, because management can recognize a need for a new groupware system, but if users do not recognize the need it makes the existence of a pull strategy difficult to confirm.

The fact that all the people interviewed at InfoProv did not have any idea about the perceived need for PMS when it was first introduced, may be an indication that there was not much emphasis put on pull strategies. Even if the application was introduced as a result of a need, it would still show a lack of pull tactics for current users not to have any idea of the reason (need) for the system to be implemented. This seemed to be the case with InfoProv.

There was one way in which to evaluate people's perceived need for software. This was by examining the use of particular functions. Although this is not the traditional way of examining pull strategies, it can provide some indication of functions and applications that users perceive a need for. In the case of

InfoProv, users perceived a need for a scheduler but did not always agree on which application should be used to fill this need (Teamlinks, Outlook etc.). This indicates that the generic need (or pull) may have been present at the time of deciding to introduce Teamlinks. This could explain the applications success in terms of user acceptance of e-mail the scheduling functions.

Evidence was found that supports the notion that pull strategies did exist in the implementation of the PMS. The fact that management decided to implement the PMS after deciding that using Teamlinks to perform the same function was inefficient, was an indication that the application was pulled. In addition, the fact that the Numeric Database Technology Group favored the application because of its benefits is also an indicator of this. The fact that the application has already been successful in the technical support department indicates that there was a perceived need for the technology.

#### *Additional Analysis*

All of the cases (see table 1) analyzed seemed to have push and pull elements present in their implementation strategy. The first two cases had strong push and pull strategies and were both considered successful in that the systems were used and learning benefits were perceived to be gained. The second two cases analyzed seemed to be marginally successful in that they were used, but the full functionality was not utilized in the latter case. It is interesting to note the first two cases had proper pull strategies, meaning the needs of the organization were properly assessed prior to implementing the system. In the second two cases needs were only assessed after the implementation to find ways of improving the system in the case of Unilever. This may indicate that systems with pull strategies, in addition to the existing push, are more likely to succeed in terms of user acceptance and perceived learning benefits.

Additional literature supports the argument that the presence of pull strategies can improve the success of groupware applications. Bontis (1999) found that information that can not be used does not benefit an organization. If a groupware system is properly assessed using a pull strategy (meaning it can meet user's needs) then it is more likely that the information that the system will disseminate will also meet users' needs. If this is the case then groupware systems that are pulled are more likely to disseminate information that can be used, and therefore provide learning benefits. Fiol and Lyles (1985) pointed out that information overload can be a hindrance to OL. Again, if a proper pull strategy was implemented which assessed the needs of the organization, it would rule out implementing any systems that would overload users, and hinder learning. Therefore, according to Fiol and Lyles, theory of information overload groupware implementation-success-rates could be improved by utilizing pull strategies.

Grudin (1994) has pointed out that one of the problems with groupware is that support has focused in the organizational and individual levels. In the cases analyzed and at InfoProv, it seemed that most of the need assessment took place at the individual level. Possibly, pull strategies could be improved if they looked at the needs of the group level, which would be the most appropriate for collaborative applications.

It should be noted that while this research did indicate that there is a lack of presence of pull strategies, it did not indicate why this is the case. It is not necessarily the case that pull strategies did not exist at InfoProv and in the cases examined. It is possible that pull strategies were present at the time of the conception of the idea of the system, but new managers that take on the responsibility of the system may not be aware of these. It is also possible that pull strategies never existed in the first place. It is believed that both these cases could hinder the potential success of a groupware system. In the former case, managers who implement systems that they do not understand the need for, may have problems convincing people of the relevance of such a system. This could lead to user resistance. In cases where pull strategies are not used, management is taking a gamble by implementing a system that may not be needed.

#### *Realization that Push and Pull Cannot be Separated*

From primary research and review of groupware implementation cases, it is believed that push and pull strategies are far from mutually exclusive. On the contrary, elements of both seem to be pervasive in most cases. It is unlikely that any system would succeed as a result of only one type of implementation tactic. Orlikowski's (1996) indirectly argues this by putting forward a theory of emergent change. Emergent changes come about over time, and are ongoing, which implies they rely on needs to be seen as changing (Orlikowski 1996). Therefore new applications will be needed to meet user's changing needs. Also, changes are not always instantaneous, as oppose to the theory of punctuated equilibrium, which implies that push tactics cannot insure the immediate success of a new application (Orlikowski 1996). Buchanan (1998) noted that it is not just top-down management support and other push techniques that lead to successful groupware applications. It is also necessary to have bottom up user support, which comes from having the correct pull tactics to support groupware implementation. This view is in line with Zmud's (1989) belief that success of technological innovation implementations can be improved by having both types of strategies. The analysis of the cases also indicates that this is true.

### *Prescriptions*

There has been some guidelines provided in the literature for improving the prospects of groupware implementations. An example is Munkvold (1998) who advises management to create incentives, provide guidelines for using the system and provide training for using the system. It should be noted that these guidelines include both push and pull strategies. The former two are pull strategies and the latter a push. The fact that Munkvold (1998) has emphasized pull over push strategies for improved implementation of groupware may indicate that the lack of success is a result of a lack of emphasis on pull strategies.

The intention of this analysis of push-pull strategies was not to nullify existing strategies of improving groupware implementations, such as those provided by Neilson (1997), on the contrary it aims to complement those strategies by providing an additional tool to aid in the implementation process.

Findings indicate that there is a lack of presence and awareness of pull strategies and that this may be causing groupware applications to fail. These findings need to be confirmed by further research, which will be outlined in the conclusion.

### **Conclusion**

This paper has aimed at contributing to the disciplines of organizational learning and information systems by examining a possible way to improve groupware implementations, which could lead organizations gleaning additional learning benefits.

Groupware's failure to provide full learning benefits is a result of a mismatch between the technology and the organizational context (Ciborra and Patriotta 1996). It may be possible to alleviate this problem by doing a better job of matching the technology to the existing context. This may be achieved by incorporating both push and pull tactics of implementation. In particular, pull tactics can be used to properly assess the needs and constraints of the existing organizational context, and by doing so provide a rich picture that can guide the selection or development of the according groupware application. By taking such steps, practitioners could improve the success rate of groupware application implementations.

As Markus and Robey (1988) point out, the organizational learning field is best researched via case study methods because it is necessary to look at multiple levels of analysis and gain a holistic perspective. Unfortunately, due to time constraints it was not possible to conduct a case study. Further research done

in this area should consider the case study approach as it would provide the depth and richness that is needed to better understand the social context in which groupware is implemented, and learning takes place.

Organizations have different needs in respect to groupware systems. However, the push-pull framework applied does not fully recognize the role of radical change. The idea of assessing the needs of an organization (a pull strategy) and implementing the appropriate technology assumes that the needs that are assessed will remain constant for a fixed period of time. On the contrary, in today's rapidly changing environment needs are likely to change constantly. Therefore the assumption of constant needs which the push-pull framework makes should be made explicit in any further research in this area. In line with the interpretivist approach, the results found at one organization cannot easily be generalized. In other words, the findings are not statistically significant and accordingly quantitative research, possibly in the form of a structured survey, should be conducted to test whether or not the findings apply to a statistically significant number of organizations.

By no means was the execution of the primary research flawless. Many difficulties were encountered along the way. For one, it was difficult to secure meetings with the correct respondents at InfoProv. In retrospect it would have been better to interview a project manager who was involved with the decision to initially implement PMS. This would have provided more insight into the initial pull strategy. In addition, it was difficult to glean the correct information from respondents, even after interviews were secured. The main problem was keeping interviewers on track.

After reviewing the organizational learning literature, it became evident that most of the literature is concerned with the acquisition of knowledge. Not enough attention has been paid to the dissemination of knowledge, which is a function which groupware supports (Nevis et al 1995; Neilson 1997). Accordingly, additional research is needed in this area to better understand groupware's role in this function and how it can be improved. As Ciborra (1996) has pointed out, looking at the learning processes around the application could provide some insight into the dynamics between groupware and the surrounding organization. This is an area that requires further exploration.

Successful implementation of groupware is not enough to guarantee full learning benefits. As Nonaka's theory of order from chaos implies (1988) it is necessary to ensure that the system is continually fed with new up to date information to disseminate throughout the organization. There seems to be a gap in the literature in respect to looking at the quality of information input into groupware systems. This area is

relevant for further research for the simple reason that if junk-information is put into groupware systems then junk-knowledge will be the output. As discussed in the analysis, this area can be researched by looking at how pull strategies can be used to alleviate this problem.

More research is needed to explore the effects of using push-pull strategies to improve groupware implementation projects' success. In particular, in-depth case study and quantitative research should be conducted to test whether or not more emphasis on pull strategies will help improve the success rate of groupware systems.

Within the area of pull implementation strategies it would be helpful for researchers to identify whether the problem is a lack of pull implementation strategies or a lack of awareness and communication of them. Once the problem has been established academics and practitioners alike can work on finding some prescriptions to deal with the problem of unsuccessful groupware implementation and the resulting unfulfilled learning potential. Further research in this area may help organizations learn more effectively, which will become increasingly important in today's rapidly changing competitive environment.

## References

- Ackerman M S (1990) *Answer Garden: a tool for growing organizational memory*. Center for Information Systems Research MIT, Cambridge, Massachusetts.
- Alsop S (1997) Lotus Notes is a Software Fantasy. *Fortune* 135(No.4),pp.183-184.
- Andreu R and Ciborra C (1998) *Organizational Learning and Core Capabilities Development: The Role of Information Technology*, Ch 7,pp.87-109 in Galliers, RD. and Baets W R J (Eds) (1998) *Information Technology and Organizational Transformation*. John Wiley and Sons: Chichester.
- Argyris C (1990) *Overcoming Organizational Defenses: Facilitating organizational learning*. Boston: Allyn and Bacon.
- Argyris C (1995) *On Organizational Learning*. Cambridge MA, Blackwell Press.
- Argyris C and Schon D (1996) *Organizational Learning II: Theory, method and practice*. Reading, MA: Addison Wesley.
- Bontis, Nick. (1996) There's a price on your head: Managing intellectual capital strategically. *Business Quarterly*, Summer.
- Bontis, Nick, Dragonetti, N., Jacobsen, K. and G. Roos. (1999) The Knowledge Toolbox: A review of the tools available to measure and manage intangible resources, *European Management Journal*, 17(4), 391-402.
- Bontis, Nick. (1999). "Managing Organizational Knowledge by Diagnosing Intellectual Capital: Framing and Advancing the State of the Field", *International Journal of Technology Management*, 18(5/6/7/8), 433-462.
- Buchanan (1998) *Consequences of Groupware use: An organizational issue based perspective*. MSc, London School of Economics.
- Cangelosi V and Dill W (1965) Organizational Learning: observations toward a theory. *Administrative Sciences Quarterly*, September.
- Chauhan N (1998) *Organizational Learning and Groupware Technologies*. Term paper, LSE Department of Information Systems.
- Choo C (1998) *The Knowing Organization*. New York: Oxford University Press.
- Ciborra C (1996) *Groupware and Teamwork: Invisible Aid or Technical Hindrance?* John Wiley and Sons, Chichester.
- Ciborra C (1993) *Teams, Markets and Systems*. Cambridge University Press, United Kingdom.
- Ciborra C and Patriotta G (1996) *Groupware and Teamwork in New Product Development: The case of a Consumer Goods Multinational*, Ch5, pp. 121 in Ciborra C U (1996) *Groupware and Teamwork: Invisible Aid or Technical Hindrance?* John Wiley and Sons, Chichester.
- Cohen W M and Levinthal D A (1990) Absorptive Capacity: A New Perspective on Learning and Innovation. *Administrative Science Quarterly* 35(No.1), 128-152.
- Crossan M, Lane W, R White, Djurfedt L (1995) Organizational Learning: Dimensions for a Theory. *International Journal of Organizational Analysis*. 3(No.4).
- Crossan, M. & Guatto, T. (1996) 'Organizational Learning Research Profile'. *Journal of Organizational Change Management*, 9(1).
- Crossan, M., Lane, H. & White, R. (1999) 'An Organizational Learning Framework: From intuition to institution', *Academy of Management Review*, 24(3).
- Failla A (1996) *Technologies for Co-ordination in a Software Factory*, Ch. 4, pp. 61 in Ciborra C U (1996) *Groupware and Teamwork: Invisible Aid or Technical Hindrance?* John Wiley and Sons, Chichester.

- Fiol C and Lyles M (1985) Organizational Learning. *Academy of Management Review*, 10(No.4), pp.803-813.
- Garvin D A (1993) Building a Learning Organization. *Harvard Business Review* 71(No.4), pp.78-96.
- Grant R M (1996) Prospering in dynamically-competitive environments: Organizational capability as knowledge integration. *Organizational Sciences*, 7(No.4).
- Grudin (1994) Groupware and Social Dynamics: Eight Challenges for Developers. *Communications of the ACM*, 37(No.1), pp.92-105.
- Harris S (1998) *An Exploration into the relationship between organizational learning and intranet use in organizations*. MSc Dissertation, London School of Economics.
- Hirschheim R, Heinz K and Klein K (1989) Four Paradigms of Information Systems Development. *Communications of the ACM* 32(No.10), pp.1199-1220.
- Huber G P (1990) A Theory of the Effects of Advanced Information Technologies on Organizational Design, Intelligence, and Decision Making. *Academy Management Review* 15(No.1), pp.47-71.
- Huber G P (1991) Organizational Learning: the contributing processes and literatures. *Organizational Science*, 2(No.1), pp.88-115.
- Khoshafian S and Buckiewicz (1995) *Introduction to Groupware, Workflow, and Workgroup Computing*. John Wiley Publishing, Canada.
- Kim D (1993) The Link between Individual and organizational learning. *Sloan Management Review*, 35(No.1), 37-50.
- Kock N and Mcqueen RJ (1998) Groupware support as a moderator of interdepartmental knowledge communication in process improvement groups. An action orientated study. *Information Systems Journal* 8(No.3), pp.183-198.
- Learned E, Christensen C, Andrews K and Guth W (1969) *Business Policy: Text and cases*. Homewood, IL: Irwin.
- Magalhaes R (1998) *Organizational Knowledge and Learning in Knowing in Firms* (1998) in Krogh G, Roos J and Kleine D in *Knowing in Organizations*. Sage, London
- Malone T.W. & Crowston K (1994) The interdisciplinary study of coordination. *ACM Computing Surveys*, 26(No.1), pp.87-119.
- Miller D (1996) A Preliminary Typology of Organizational Learning: Synthesizing the Literature. *Journal of Management*, 22(No.3), pp.485-504.
- Mintzberg H, Ahlstrand B and Lampel J (1998) *Strategy Safari: A guided tour through the wilds of strategic management*. New York: The Free Press.
- Mumford A (1992) Individual and Organizational Learning: The pursuit of Change. *Management Decisions*, 30(No.6.).
- Munkvold BE (1998) *Challenges of IT Implementation for supporting collaboration in distributed organizations*. Department of Information System, Agder College, Norway, Unpublished Paper.
- Neilson R (1997) *Collaborative Technologies and Organizational Learning*. Idea Publishing Group, London.
- Nelson R and Winter S (1982) *An Evolutionary Theory of Economic Change*. Belknap Press, Cambridge, MA.
- Nevis E, Dibella A, Gould J (1995) Understanding Organizations as Learning Systems. *Sloan Management Review* (Winter), pp.73-85.
- Nonaka I & Takeuchi H (1995) *The Knowledge Creating Company*. New York: Oxford University Press.

- Nonaka I (1988) Creating Organizational Order out of Chaos. *California Management Review*. 30(Spring), pp. 57-73.
- Nonaka I (1991) The Knowledge-Creating Company. *Harvard Business Review* (November-December), pp.96-104.
- Orlikowski WJ (1996) Improvising Organizational Transformation Over Time: A Situated Change Perspective. *Information Systems Research* 7(No.1), pp.23-59.
- Porter M (1980) *Competitive Strategy*. New York; Free Press.
- Simon H A (1991) Bounded Rationality and Organizational Learning. *Organizational Science*, 2(No.1), pp.125-134.
- Stewart T (1997) *Intellectual Capital: The New Wealth of Organizations*. Doubleday/Currency: New York.
- Venkatraman N (1991) *IT-Induced Business Reconfiguration*. In: *the Corporation of the 1990's: IT and Organizational Transformation*. Edited by Scott Morton. London, John Wiley and Sons.
- Zmud R (1984) An examination of push-pull theory applied to process innovation in knowledge work. *Management Science* 30(No.6), pp.727-738.

Innis Ref.  
HD  
45  
.W657  
no. 102

The activities of the Management of Innovation and New Technology Research Centre are generously supported by:

- The DeGroot family
- DuPont Canada Inc
- Nortel
- Royal Bank
- The Society of Management Accountants of Ontario



## INFORMATION . . .

For information about the Management of Innovation and New Technology Research Centre or innovation research activities at the Michael G. DeGroot School of Business:



*Director; Dr. C. K. Bart, Ph.D., C.A.  
MINT - Research Centre  
Michael G. DeGroot School of Business  
McMaster University, 1280 Main St. W.  
Hamilton ON Canada L8S 4M4*

*Phone: 905-525-9140, Ext. 24105*

*Fax: 905-521-8995*

*email: [mint@mcmaster.ca](mailto:mint@mcmaster.ca)*

*www: <http://mint.mcmaster.ca/>*

\*Management of Innovation and New Technology Research Centre (MINT) is an official mark of the Michael G. DeGroot School of Business, McMaster University.