

Communication Tools for Distributed Software Development Teams

M. Rita Thissen
RTI International
PO 12194
RTP, NC 27709 USA
00-1-919-485-7728
rthissen@rti.org

Jean M. Page
RTI International
3516 Peninsula Circle
Melbourne, FL 32940
00-1-321-253-2725
jpage@rti.org

Madhavi C. Bharathi
RTI International
396, 43rd Cross, 8th Block
Jayanagar
Bangalore 560082, INDIA
00-1-919-386-0224 (US)
mbharathi@rti.org

Toyia L. Austin
RTI International, Suite 119
2951 Flowers Road South
Atlanta, GA 30341 USA
00-1-770-234-5023
taustin@rti.org

ABSTRACT

This paper examines communication tools for software development and data management teams who are distributed geographically. Using a case study approach, we describe collaboration processes used by RTI International's (RTI's) application programming teams for three projects. Techniques range from simple conference calls and email through webcasts, collaboration websites and remote server control. These tools allow teams to work together to produce software and data products in spite of differences in locations, time zones and network access. The qualitative findings are relevant to increased globalization, outsourcing and telecommuting in the software development workforce.

Categories and Subject Descriptors

D.2.9 [Software Engineering]: Management – *programming teams*.

General Terms

Management, Human Factors.

Keywords

Virtual Team, Global Workforce, Distributed Development, Communication.

1. INTRODUCTION

In the past, almost all software development was accomplished by co-located teams. For any particular development project, the team leader, designers, developers and testers worked together in close physical proximity. There was always an “expert in earshot.” [6] The concept of paired programming [7] did not exist, because it was so commonplace that people did not see it as a noteworthy behavior. For internal development projects, even the user community might be located nearby, handy for consultation on feature needs or demonstration of an application.

Now the business world has changed. Successful companies can be dependent on the use of these distributed teams for software development. [10] Distributed software development teams have become common, located at multiple sites and often in multiple parts of the world. Instead of a team working in the same building, there may be several locations in which the managers, developers and testers operate, potentially separated by hours, miles and cultures from each other and from the customers or end users of the product. [5] Often, a team is formed as no more than a community of strangers [8] given the task of producing a common software or data product. Exchange of knowledge among these remotely located software development teams must overcome new barriers in order to provide interactions and communication comparable to the original co-located operations. The ability to communicate may be directly related to project success. [12]

Distributed teams face several communication obstacles. Not only are there time zone differences, an inability to observe one's colleagues in person and increased difficulty exchanging goods,

Parts of the research upon which this report is based were supported by the U.S. Census Bureau, and we would like to acknowledge members of the Coverage Follow-Up team for their help. The views expressed are those of the authors and not necessarily those of the U.S. Census Bureau. This report is released to inform interested parties of ongoing research and to encourage discussion.

We would also like to acknowledge help from the team working on the Universities of Nairobi, Illinois and Manitoba (UNIM) Study. RTI is the data coordinating center (DCC) for UNIM, a randomized clinical trial of male circumcision to reduce HIV incidence in Kisumu, Kenya. The study is funded jointly by the U.S. National Institute for Allergy and Infectious Disease (NIAID) and the Canadian Institutes of Health Research. The DCC is funded through an NIAID grant to the Principal Investigator, Robert Bailey, Ph.D. at the University of Illinois at Chicago.

Parts of the research upon which this report is based were supported by the U.S. Centers for Disease Control and Prevention (CDC). We would like to acknowledge the PRAMS team (Pregnancy Risk Assessment and Monitoring Surveillance) for their help. The PRAMS project was directed and funded by CDC, National Center for Chronic Disease Prevention and Health Promotion, Division of Reproductive Health.

The views expressed throughout the paper are those of the authors and not necessarily those of the funding organizations or project staff.

but there are behavioral barriers. An individual who will not think twice about being seen in person by a co-worker may balk at appearing on camera for a video conference. One who will readily speak to another person in the same room may be reluctant to call by telephone. Requesting help debugging a piece of software may require scheduling a mutually agreeable time, setting up remote access or organizing video services. Distributed software teams face these obstacles and others. Software development is carried out simultaneously by multiple team members and then combined into a single product. There is a high degree of interdependence in the tasks that is critical to the project's success. [2] A strong virtual team leader recognizes the stumbling blocks, promotes communication and encourages all to participate freely. [1]

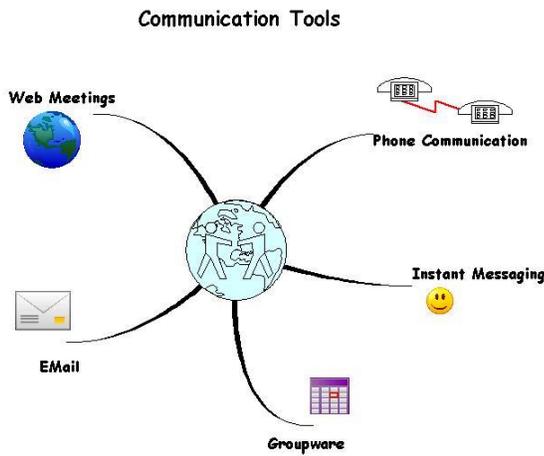


Figure 1. Communication tools for use by distributed teams

For successful teaming, the virtual group must provide tools which will allow communication to flow easily. The choice of tools may be important. It is widely believed by educators that people differ in preferred modes of communication, especially with respect to learning styles [3] [11]. For some, visual presentation proves most effective. Speaking, listening and discussing may work better for others. Some do best with tactile learning, in which they have an opportunity to explore the behavior of software through hands-on experience. Most people combine these approaches or use one or more selectively, depending on the circumstance. Accommodating the desire for multi-channel communication and increasing frequency of contact can lead to improved remote collegiality, depending upon the choice of tools and mechanisms for exchanging information⁹.

In this article, we take a look at three successful distributed software teams from RTI International, and at the tools they used, with the thought that the examples may provide guidance for future teams. RTI is a non-profit research organization that does contract work in a variety of scientific and engineering fields. It has a substantial group of software engineers and data managers, many but not all of whom are located at the home offices in North Carolina. When a new contract or project starts, a work team is assigned, based on the skills needed and availability of personnel. For the purpose of this project, we chose to study three teams, each of one spanning multiple geographic locations while sharing a common business purpose. Some members encountered each other for the first time as part of the team, some worked in physical proximity at times during the life of the projects, and

others never met each other at all. We examine the communication processes and tools which each team used to create a sense of virtual community, with commentary on the perceived effectiveness of each approach.

2. OVERVIEW OF TOOLS AND PROCESSES

Collaborative software and systems include instant messaging, email, shared calendars, remote control software, audio/video conferencing and any other tool that supports groups of people engaged in a common task. Sharing knowledge by voice, document, image and shared operational software supports individuals' needs for aural, visual and tactile communication. The immediacy or timeliness of the exchange varies, with some processes and systems serving better for specific needs. Synchronous systems act in unison, with participants at all locations receiving and sending information dynamically in "real time", perceived to be simultaneous. Asynchronous systems provide a mechanism for submission and retrieval, which allows information to be transferred over a period of time which is determined by the availability of the recipient.

One often-used synchronous tool is chat or, when between two individuals, instant messaging (IM). IM has become a business tool, falling somewhere between a phone call and email with respect to formality and immediacy. Once the service is set up, the user maintains a dynamic contact list which can indicate if any person on the list is online and/or available. If online, either party can send text or graphic messages to the other. When more than two people participate in IM at once, the technique is known as chat. Like telephone calls, chat can take place in real time. Like email, a message can be sent to another online user and persist on their monitor screen, even though the response may be delayed. Most IM and chat clients offer a logging option, for conversations which require a stored record.

Groupware is the name given to a general category of software which enables multiple users to use a single software resource. Tools like Lotus Notes, Microsoft Exchange and Novell Groupwise provide shared calendars, group meeting scheduling and shared contact lists. Each person maintains his or her own information in a private calendar, controlling which other users have access. When a meeting needs to be arranged, the program can suggest times when the participants are available and send a meeting invitation. Meetings can be arranged as in-person encounters, telephone conferences, web casts or any combination of these interactions. Other forms of groupware exist, such as whiteboards and document managers, but scheduling tools are among the most commonly used.

Meetings take on a new character when the team is a distributed virtual team. The internet provides a medium for "web conferencing" or electronic meetings using software or services from various organizations. We mention a couple here, but many more are available, either as a free download or for a fee. NetMeeting is a free tool available from Microsoft. It provides multi-point data conferencing, text chat, whiteboard, file transfer, audio and video. It is for use when all participants are logged in to a single network, even if that network spans great distances. Once connected, there are features that can be initiated such as web camera for video or sharing a dynamic image of the host's

desktop. Desktop sharing allows all participants in a meeting to see what appears on the shared computer desktop. It can be used for demonstrating, training or clarifying how applications work. There are companies, like WebEx, that provide meeting services for teams who do not belong to a common network, with a per-minute fee to the host for all who join the meeting. The participants only need a web browser and internet access to connect and use the service. Web conferencing and on-line meetings take place “live”, except for possible signal lag of a few seconds if the team members are extremely distant.

Email is still one of the most popular ways to pass information to people in different time zones or across the room. Its asynchronous nature allows the sender to transmit information regardless of whether or not the recipient is available at the moment. Storage of the message is at least temporarily on a central server, which maintains a persistent copy of the mail until it is retrieved or removed by the recipient. Email may include optional attachments of audio, video, document or other electronic content, depending on the specific software and protocol in use.

Table 1. Tools for distance communication and information exchange

Tool	Examples	Uses and Advantages	Bandwidth Needs, Immediacy	Sensory Modes
Instant Messaging and Chat	<ul style="list-style-type: none"> • Yahoo Messenger • MSN Messenger • AOL Instant Messenger • Internet Relay Chat 	<ul style="list-style-type: none"> • Instant interaction • Less intrusive than a phone call • View who is available • Low cost • Low setup effort 	<ul style="list-style-type: none"> • Low bandwidth, can use dial-up • Immediate • Synchronous or asynchronous 	<ul style="list-style-type: none"> • Visual • Text and limited graphics
Groupware / Shared Services	<ul style="list-style-type: none"> • Lotus Notes • Microsoft Exchange • Novell Groupwise 	<ul style="list-style-type: none"> • Calendars • Contact Lists • Arrange meetings • Cost and setup effort vary 	<ul style="list-style-type: none"> • High bandwidth is preferable • Low bandwidth may be enough • Asynchronous 	<ul style="list-style-type: none"> • Visual
Remote Access and Control	<ul style="list-style-type: none"> • NetMeeting • WebEx • Remote Desktop • pcAnywhere 	<ul style="list-style-type: none"> • User controls a PC without being onsite • Cost varies • Setup varies 	<ul style="list-style-type: none"> • High bandwidth • Immediate • Synchronous 	<ul style="list-style-type: none"> • Visual • Audio • Tactile
Web Conferencing	<ul style="list-style-type: none"> • NetMeeting • WebEx • Citrix GoToMeeting 	<ul style="list-style-type: none"> • Live audio • Dynamic video • Whiteboard • Application sharing • Moderate cost and setup effort 	<ul style="list-style-type: none"> • High bandwidth • Immediate • Synchronous 	<ul style="list-style-type: none"> • Visual • Unlimited graphics • Optional audio
File Transfer	<ul style="list-style-type: none"> • File Transfer Protocol (FTP) • Collaborative Websites • Intranets 	<ul style="list-style-type: none"> • Share files of any type • Cost varies • Moderate setup effort 	<ul style="list-style-type: none"> • Low bandwidth for small files • High bandwidth for large files • Asynchronous 	<ul style="list-style-type: none"> • Varies with file content
Email	Numerous vendors and free applications	<ul style="list-style-type: none"> • Send messages or files • Cost and setup effort vary 	<ul style="list-style-type: none"> • Low bandwidth, can use dial-up • Asynchronous 	<ul style="list-style-type: none"> • Visual • Audio in attached files
Telephone	<ul style="list-style-type: none"> • “Plain Old Telephone Service” (POTS) Voice Over Internet Protocol (VOIP) 	<ul style="list-style-type: none"> • Direct calls • Conference calls • Cost varies • Low setup effort 	<ul style="list-style-type: none"> • VOIP requires high bandwidth • Immediate • Synchronous • Asynchronous for voice mail 	<ul style="list-style-type: none"> • Audio

Websites may also be set up for collaborative work, especially for sharing documents which might otherwise exceed email capacity or violate email security, such as executable packages. Whether using secure FTP or a custom-built portal, a website offers access at any hour of the day, little maintenance effort and the convenience of accessibility from anywhere. Use of Secure Socket Layers and password-protected logins prevent unwanted intrusion by others who are not on the development team. For a

virtual team which does not share a common network, the web portal can replace file server shares as a confidential location for storing community files.

Table 1 provides examples of each type of technology, its uses and values, how quickly the information is available, and which sensory modes come into play. The list is by no means comprehensive, but is intended to provide a conceptual guide to some of the common tools.

Each of the tools listed above has been used to some degree by the teams we chose to study for this article. There are other tools which remote colleagues might choose, such as video conferencing, which give more of a face-to-face feel in spite of the distances. We are limiting our review to those which were actually employed. An evaluation of their effectiveness and usefulness to the teams is given in the next section below. crosses continents. For comparison, the characteristics of

each team are given in Table 2. Although they differ in some ways, they share the general features of being spread across multiple locations, in multiple time zones, with a focus on computing products.

Table 2. Characteristics of three distributed software and data management teams

Team	Extent	Purpose	Duration	Composition
Team One	US, 30 +/- Sites	Data services	Ongoing for 15+ years	1-4 Software developers 3-4 data managers.
Team Two	1 location in each of three countries (USA, Kenya and India)	Development, deployment of a data entry system	Ongoing since 2002	3 Software developers 1 Tester 1 Project leader A coordination group consisting of 4-5 members End users
Team Three	US, 4 locations across 3 states	Software development	2 Years	17 Customers 5 Software developers 5 Testing and quality assurance staff 2 Support staff

3. SOFTWARE AND DATA MANAGEMENT TEAMS

3.1 Team One

Team One's purpose was to update a data entry system and re-engineer an automated data management and processing system. Software in use for data entry was designed to meet the specifications of 30 data collection sites located throughout the US, conducting a research survey of local individuals. No two surveys are alike across sites, although some questions are common to all. Each data collection site required the development of two separate data entry systems; one for mail responses and the other for telephone responses, as well as individualized data management programs. Updating the system was a critical task due to the fact that errors in the data entry software seriously impact the data processing system.

Development, deployment and maintenance of the data entry system and the data management and processing systems were the responsibility of a team of programmers and data managers located in Florida and Georgia, in the United States. There was no single lead person on the team, as during the update, responsibility is passing from some members of the group to others.

Skills required for this purpose were:

- Software project management
- Design and programming expertise
- Quality assurance experience
- Administrative and documentation skills
- Understanding the data processes and characteristics of the data

In this team, activities varied across position. Each team member brought the skills listed with an emphasis pertaining to job function.

All work was done within the United States. The data entry software development was performed in Georgia. The data processing systems have been developed by two programmers, one in Georgia and the other in Florida. The two programmers worked together as a virtual team to create programs to automate data processing. Communication was frequent and ranged from formal to informal. Team meetings were held on a weekly basis with the entire development team using standard conference calls. Other information was passed in email messages, and NetMeeting was used once to get a clearer view of some processes. Some of the team members used instant messaging to get quick answers.

With this team's management by peers, coordination of work and exchange of information was accomplished by consensus and mutually-accessible files. A shared drive was created as a testing environment and is used to maintain tasks which had been completed. It was seen to be most efficient for every team member to have access to the shared system where the data is processed. It was also imperative to have only one person responsible for each task instead of multiple persons due to the

Table 3. Communication tools used by Team One

Tool	Product	Extent of Use	Effectiveness of Use	Ease of Use
Remote Access	Net-Meeting	Rare	High. Used to demonstrate / identify necessary processes	Moderate
Email	Outlook	Daily to Weekly	High	Easy
Telephone	Nortel	Daily to Weekly	High	Easy
Instant Messages	Yahoo Instant Messages	Now and Then	High	Easy

increased chance of error and data being overwritten, and so team members agreed on a division of responsibility.

Table 3 lists the tools used by Team One. This team relied largely on file-sharing, telephone calls and email, with some use of other electronic tools as needed. The small size of the team allowed the members to work together easily without additional software.

3.2 Team Two

As the second case study, we examine the communication tools used by the software development team working to reduce HIV

incidence in Kisumu, Kenya. The software enabled the data entry staff located in the clinic in Kenya to key data written on paper forms into a database for further analysis.

A larger project group consisted of two teams, the coordinating team and the software development team. The coordinating team interacted with the customer to design and provide specifications on

- System appearance and structure
- Reporting components
- Data validation tools

The software development team gathered the specifications from the coordinating team, and developed the software systems. This team worked on

- Software project management
- Design and programming in the selected development languages and databases
- Initial testing

Both teams worked jointly on these tasks:

- Software system testing
- Documentation
- Data validation tools

We focus here on the software development team. The software development team comprised the project leader, three developers and one tester. All the members of the software development team were located in the same building with the exception of one developer who telecommuted from India. Most of the members of the coordinating team worked out of another building a few miles away from the software development team. The end-users of the software systems were located in a clinic at Kenya.

The members on the development team had worked together on previous occasions. Only the team leader had personally met with members of the customer team in Kenya during the deployment phase. In spite of this lack of direct in-person contact, team members performed well as a unit by being open to communication and sensitive to differences in location and role.

The project leader interacted with the coordinating team by means of meetings, telephone conversations and email. The team members interacted with the end users by means of instant messaging and email due to the geographical distance. The development team in the US communicated with the developer in India using standard telephone, VOIP telephone calls, instant messaging and emails. Due to the expense involved in calling internationally by telephone, emails and instant messaging were the preferred communication tools for long distance communication.

Some of the challenges faced with respect to communication among the team members were varying geographic locations and time zones. The barriers were overcome mainly by careful scheduling of conference calls to take place in daytime for all parties and correspondence through email, which allowed asynchronous work.

Knowledge, data and software (both code and applications) were shared among the development team with the help of Visual

Source Safe and shared folders on the development servers. Applications and data were transferred to the end user machines

Table 4. Communication tools used by Team Two

Tool	Product	Extent of Use	Effectiveness of Use	Ease of Use
Group ware / Shared Services	Visual Source Safe	Used to modify all source code	High	Moderate, requires some training
Remote Access	Remote Administrator	Used regularly to transfer files and maintain the source code	High	Easy
File Transfer	Blast, Remote Administrator	Daily	High	Moderate
Email	Outlook	Daily	High	Easy
Telephone		Now and Then	High	Easy
Group ware	Yahoo Instant Messages	Now and Then	High	Easy

via the internet using a tool called Blast. The computers in Kenya were accessed from the development site using a remote connection tool called Remote Administrator. This tool was very effective during the deployment and maintenance phase as it provided the development team with a means of tracking the state of the application and database at real time. Team Two's tool use is summarized in Table 4, above.

3.3 Team Three

The third team was formed to create software from custom specifications for a particular client. Software was designed and produced according to a spiral model of development in two sequential year-long efforts. During the first year, one set of applications was developed from start to finish, with seven planned releases, each more closely approaching a final complete software package. During the second year, a similar process was followed for a second application package derived from the first.

Most of the software development and testing staff were located at three separate business parks two to three miles apart, except for one team member who telecommuted from approximately 1200 miles away. All team members were within the span of two time zones

Software developers on the team were selected by RTI for their skills, regardless of location. Skills included

- Software project management
- Design and programming expertise in the selected development languages
- Expertise in a specific automated testing system
- Quality assurance
- Administrative and documentation

Customer team members were located in several buildings within a single large campus, approximately 250 miles from the majority of the software developers and within the same time zone as all but the telecommuter.

Customer team members were selected by the sponsoring organization to provide

- Project management and oversight
- Requirements, specifications and acceptance criteria
- Change management
- User, system and integration testing

The team members ranged from total strangers to a few who had worked together in the past. Within the software developers, several had prior experience working with each other, but some had not met previously. Similarly, within the customer team, some were acquainted with one another, some had worked together in the past, and some were newly introduced on this project. Customers and developers had not met prior to team formation. In short, the group was constructed largely of people who had little or no experience working together.

Team members gradually became accustomed to working at a distance with only voice and data communication. At first, a need was felt to meet in person and during the first seven months four face-to-face meetings were held to review requirements and demonstrate progress. Not all team members attended each meeting. A fifth in-person meeting was held to present the product to a separate group. Throughout the effort, weekly conference calls were held, with written minutes distributed within 24 hours, to keep communication at a high level, and informal calls and email were encouraged. A web portal was created for exchange of large files and formal documents. As the development team became more comfortable working together, the need for in-person contact lessened, and only one face-to-face meeting was held in the remaining 17 months.

Much has been made of the need for frequent communication in establishing a sense of continuity and camaraderie among co-workers. This long-distance team achieved that sense of unity through the aid of almost all of the communication tools listed in Table 1. Table 5 lists the tools used, the frequency of their use, and their effectiveness.

Team meetings were held weekly by standard telephone conference calling, with attendance by those who were involved in the current stage of development. Between conference calls, individuals exchanged email and casual phone calls, sometimes as often as several times a week. Webcast technology was used to display and discuss implementation options, and a web portal provided a common site for sharing files.

The most effective tools for the team in general were telephone and email. Within each organization, calendaring and intranet shares facilitated teamwork, and a web portal was created for cross-organizational shared storage space. Although all documents and software versions were placed on the website and some team members depended on it heavily, most team members continued to rely on email for exchange of files. Web conferencing was used successfully on one occasion to demo and discuss the software, but usually application versions and

Table 5. Communication tools used by Team Three

Tool	Product	Extent of Use	Effectiveness of Use	Ease of Use
Group ware	Exchange and Lotus calendars. Visual Source Safe	Weekly	High. Used within each organization, but not unified across organizations.	Moderate. Coordination of schedules and notices required learning.
Remote Access	Remote Desktop, Net-Meeting	Now and then	High. Used for cross-location problem-solving among the development team.	Confusing for novice users.
Web Conferencing	AT&T	Now and then	High. Very beneficial for discussing software between releases	Somewhat confusing on first use
File Transfer	Private website with simple functions	Varied by person, from never to frequent.	High. Very effective for transferring documents, data and installation packages.	Easy access. Some effort required to create and maintain the system
Email	Outlook, Lotus	Weekly between organizations. Daily or more often within each organization.	High. Provides a persistent record with source and time-stamp. Less intrusive than phone calls.	Easy
Telephone	POTS	Several times per week within each organization. Weekly across organizations.	High since all were within adjacent time zones. All group calls had written agenda and follow-up notes.	Easy

documents were emailed to team members for installation and testing on local equipment.

Within the software developers' group, remote access software was combined with phone contact frequently between distant

sites for the effect of paired programming, particularly as an aid in debugging thorny problems. The immediacy, convenience and combination of audio and visual information, with the option of allowing either programmer to control the cursor, provided a highly effective way to share knowledge without traveling.

For this team, the variety of tools enhanced the ability to communicate and led to a sense of unity of purpose and familiarity among professionals who had not worked together beforehand.

4. DISCUSSION

There are several aspects of these teams which deserve thought. Which communication tools are best? How often should team members interact? Do these three case studies represent the virtual software engineering as a whole? Do the team members need to have regular in-person meetings? How important is a shared file repository? There are other concerns to address in managing any team, but they seem fundamental to working effectively in distributed groups.

Which tools are most effective seems to depend on the purpose and make-up of each team. In contrast to studies cited by Hiltz et al [4], in the three teams we describe, the choice of communication technology was left up to the team members and could vary over time. Each team selected different tools on different occasions, to meet their immediate needs most closely. Thus, synchronous media such as telephone, IM, remote access or webcast were used when immediate feedback or conversation was needed. Asynchronous media such as email or web-mediated file exchange were used when timing was not critical. This flexibility and control of tool choice may be a crucial factor in the effectiveness of the teams, as they tailored their toolsets to their goals.

The question of face-to-face meetings is an interesting one. Most participants in this study had expected to need more in-person time than was actually used. Team Three showed a decline in the desire and need for in-person meetings as time went on, demonstrating that communication through electronic tools could suffice to establish and maintain a sense of community.

No three case studies can adequately represent the diversity of global business teams. Specifically, all of these teams were small. Very different dynamics might be expected within teams of dozens or hundreds, whether they were collected in a single location or spread widely. The three teams we describe did, however, demonstrate some common barriers to communication which are typical of nationally and internationally distributed working groups:

- Communication across time zones
- Cost and effort of use as a factor in selecting tools (cost and technical learning curve)
- Lack of comfort or familiarity with remote teamwork for some team members at first (social learning curve)
- Setup time required for shared visuals, such as webcasts (ongoing effort)

None of the barriers proved prohibitive, and the effect of each declined as the team members worked together. Very short term

teaming arrangements might be expected to suffer greater difficulty as they might not have time to crest the learning curves.

For teams working on shared software or data files, access to communal storage was essential. Team One established a network share. Team Two employed groupware and remote access to a customer machine. Team Three created a web portal. The technology chosen was suited to the constraints faced by the teams; for those on a shared network, there was no need for internet or web-based technology. The degree of control of those shared resources varied as well. In Team One, an informal agreement among team members was sufficient, while Team Two chose the strongly structured approach of using Visual Source Safe. In Team Three, all participants had access to post or retrieve files, but only portal administrators had access to remove any.

Frequency of communication seemed to be a key to success. Regardless of communication channel or mechanism, each of these teams stayed in touch on a regular and frequent basis. The comfort level among team members results in part from routine contacts during which team members learn one another's styles, skills, personalities and areas of expertise.

5. CONCLUSIONS

We can draw some general conclusions from these studies to form guidelines for future teams.

- Allow teams to choose their own communication tools from a variety of options.
- Insist on frequent communication among all members, including some synchronous interaction (telephone, chat, webconferencing)
- Provide shared file storage to facilitate team interaction.

We can also reassure future teams that some perceived barriers are not insurmountable, given the right tools.

- There is no need for extensive travel, although early face-to-face contact is beneficial.
- Communication tools do not have steep learning curves; team members adapt quickly, especially if they have had a voice in the selection of tools.
- Paired programming can take place across long distance, through remote access or collaboration tools (virtual co-location).

The formation of data management and software teams which span distant locations will occur more often as business and industry adapt to a global economy. The growth of telecommuting, outsourcing and far-flung markets contribute to pressure for long-distance collaborative work. We believe that the lessons learned from these three case studies can inform managers who want to create their own successful and productive virtual teams.

6. REFERENCES

- [1] Blanchard, A.L. and Markus, M.L. The Experienced Sense of a Virtual Community: Characteristics and Processes. *The*

- DATA BASE for Advances in Information Systems*, 35, 1 (2004), 65-79.
- [2] Crowston, K., and Kammerer, E.E. Coordination and Collective Mind in Software Requirements Development. *IBM Systems Journal*, 37, 2 (1998), 227-245.
- [3] Giles, E., Pitre, S., and Womack, S. Multiple Intelligences and Learning Styles. In *M. Orey (Ed.), Emerging Perspectives on Learning, Teaching, and Technology*. eBook from <http://www.coe.uga.edu/epltt/mi-ls.htm>, 2003.
- [4] Hiltz, S.R., Fjermestad, J., Ocker, R.J., and Turoff, M. Asynchronous Virtual Teams: Can Software Tools and Structuring of Social Processes Enhance Performance?. In *Volume II: Human-Computer Interaction and Management Information Systems: Application*. Chapter 6, Galletta D. and Zhang P. (Eds.), M.E. Sharp, Armonk, NY, 2006.
- [5] Jarvenpaa, S.L., and Leidner, D.E. "Communication and Trust in Global Virtual Teams." *Organization Science*, 10, 6 (1999), 791-815.
- [6] Keuffel, W. Expert in Earshot. *Software Development*, 10, 12 (2002), 56.
- [7] McDowell, C., Werner, L., Bullock, H. E., and Fernald, J. Pair Programming Improves Student Retention, Confidence and Program Quality. *Communication of the ACM*, 49, 8 (2006), 90-95.
- [8] McElroy, J. *A Community of Strangers*. The New York Times Magazine, Oct 19, 1997.
- [9] McKinney, V.R. and Whiteside, M.M. Maintaining Distributed Relationships. *Communications of the ACM*, 49, 3 (2006), 82-86.
- [10] Powell, A., Piccoli, G., and Ives, B. "Virtual Teams: A Review of Current Literature and Directions for Future Research." *Data Base Advances*, 35, 1 (2004), 6-36.
- [11] Sarasin, L.C. *Learning Style Perspectives: Impact in the Classroom*. Atwood Publishing, 1998.
- [12] Werner, J.M., and Evanco, W.M. What Project Management Practices Lead to Success?. *IEEE Software*, 22, 1 (2005), 86-93.