support structures for remote international collaboration, encompassing strategies for communication, management and technology use" [2].

Since 1998, Runestone has involved over 225 students in collaborative software engineering. US students have been in their third or fourth year of university study, while Swedish students have been in their third year of university study.

For all of the Runestone instances reported here, the software development project has involved designing and implementing a distributed real-time system to navigate a steel ball through a board with a maze by tilting the surface of the board via positioning motors. This project, called Brio [2, 11], has evolved somewhat over the semesters.

For regular team meetings, students have used Internet Relay Chat (IRC). For other types of correspondence, students have used email (for example to communicate with instructors as well as with both local and remote team members). Web pages have been used initially for introductions within the teams and to share personal information. The web pages were later used to publish and make available project documents to the entire team.

Data from the Runestone Project has been collected in a variety of forms throughout the course. Sources of data have included all types of interaction between team members except for informal face-to-face meetings. The following data have been collected: background questionnaire, project logs, journals, student email and Internet Relay Chat (IRC) archives, web pages, and peer evaluations. Students were informed that the Runestone data would not be shared with the course instructors until after the course was completed and grades were assigned.

3. LEARNING FROM THE RUNESTONE PROJECT

In sections 3.1 through 3.4, we present problems encountered during each Runestone implementation and describe attempts to correct those problems in the next course offering. At the completion of each Runestone implementation, instructors, researchers, and staff have met to evaluate the just completed course and discuss changes in course structure for the following year.

3.1 Runestone 1998 Pilot Project

The first implementation of Runestone, a pilot project, occurred from early January through late March 1998 [2]. Eight students (four from Uppsala and four from GVSU) volunteered to participate. While no reliable conclusions could be drawn from a small pilot project involving only one team, the prototype did alert the researchers and instructors to problems with the project structure and team process.

The virtual aspect of the project was less frustrating for the students than the technical problems they encountered. These technical problems included: a discrepancy in programming language skills between the two local groups, writing software code for hardware that was physically located in only one place (Sweden), and testing when the target platform differed from the local platform.

The primary problem in team process during the pilot project was an "us vs. them" mentality that developed within the local sub-teams. The course instructor was located in Sweden (a GVSU professor on sabbatical at Uppsala), and the hardware was located in Sweden. The US students perceived themselves from the very beginning to be at a disadvantage. They felt that their technical skills were not equal to those of the Uppsala students and that they were "a step behind all the way." The Uppsala students in turn interpreted the Grand Valley students' behavior as a lack of motivation and commitment. A review of the IRC logs and email has not revealed any interactions that could be considered confrontational. Most discussions were on technical issues with very little social interaction. The student correspondence was polite with very little emotion conveyed by any team member. These sub-teams evidently experienced little or no explicit conflict.

3.2 Runestone 1999

The pilot version of the course turned up several technical problems. For example, the Swedish students were fluent in Java while the US students had more programming experience in C. To address this during the 1999 offering [11], the course coordinators re-designed the project to require a solution that used both Java and C. Another problem in the pilot was the fact that the Brio hardware was located in Sweden. During the 1999 offering, duplicate hardware was set up at both universities.

In 1999, two instructors (one from each university) supervised 42 students. Seven teams of six students (3 students from each institution) worked on the project from late January through the end of March 1999. Each team had one team leader and the remaining students on the team were assigned the role of developer. Half the teams had a US leader and half the teams had a Swedish leader. The US instructor had primary responsibility for all teams with a US leader and the Swedish instructor had primary responsibility for all teams with a Swedish leader.

The major technical or software-development problems that students described during the 1999 instance of the project included: no knowledge of UML, equipment not available, hardware incompatibility, communication protocols, and testing. The instructors did not have the hardware set up until a few weeks into the project, so some students had a hard time visualizing the final product. Unfortunately, when students started testing, they discovered that the code did not always work in exactly the same way at both locations. Because the project required different programming languages, students needed a standardized way of communicating between subprograms. When asked in the final journal for advice to future students, the majority of students said, "Start testing earlier!".

The major team-process problems that students had with the virtual team project included: poor communication, member non-participation, poor leadership, lack of technical skills, procrastination, and differences in motivation. Of the problems cited, many are common to all student teams. For example, Pournaghshband [15] reported that more than 50% of students in single-institution courses that include team projects reported problems with poor communication, member non-participation, poor leadership, and procrastination. From an analysis of the IRC logs and emails, it seems that problems with poor communication and differences in motivation are amplified when students are working in a virtual team.

The 1999 implementation uncovered two interesting items with respect to group development. First, the instructors decided to require the students to use UML (Universal Modeling Language)

in the requirements specification. None of the students at either GVSU or Uppsala were familiar with UML. This common lack of knowledge seemed to foster more cohesion among group members. Because no one knew anything about the topic, they were all equals and could complain to each other. Second, we have found that teams that permitted themselves to have some healthy conflict with both local and remote teammates were rated by the instructors as producing better software and having a more cohesive team structure. Study B is currently using grounded theory to explore this finding.

3.3 Runestone 2000

For the year 2000 offering of the project, the instructors changed the course structure by setting up frequent deadlines with set deliverables. This was done to allow the instructors to recognize dysfunctional teams quickly and to enable students to move from conflict to cooperation earlier in the project. The set deliverables also added more clarity to what was required from the students, as each team was required to give a presentation and prepare a report at the end of each major milestone. Ninety-three (93) students participated in Runestone 2000.

Students evaluating the 1999 course offering recommended putting all students in the role of developer, with the leader role defined as an additional responsibility. This change was implemented in the 2000 offering.

An interesting twist on the interaction between the two universities came about during the 2000 academic year, when GVSU and Uppsala began a formal student exchange program. A few students in the Computer Science departments from each university took this opportunity and ended up as participants in the Runestone project from the "other" side of the Atlantic. In addition to the student exchange, at least one US student took his spring-break holiday in Sweden and several others made plans to visit their foreign team members in the future

The software development task, to design and implement a distributed real-time system to navigate a steel ball through a maze board, remained the same. The instructors made the code from the student projects for 1999 openly available via the project web site. The teams then had the choice of developing the project from scratch, enhancing code from the previous year's project (giving credit to the original authors), or using the past code as examples for guidance.

Students reported three major problems related to the hardware. These were:

- equipment breaking down "It was frustrating working with the particular equipment because it seemed to be broke more than fixed"
- equipment having different set-ups at the two locations: " it would be great if all the equipment was even at the different locations"
- not enough resources: "you need a lot more boxes so that 4 teams can work at a time"

As with earlier years, the teams encountered a few problems as well as some very positive experiences. One problem brought out during the early weeks of the project was that some teams had difficulty understanding the project's objectives. Some of the advice to the faculty was "get the project started earlier" and give "more information about the project at the beginning of the course."

In general, the teams were able to overcome these challenges and were able to see the positive advantages of the project. They saw these as "a good experience for the future and for the working life, it offer a rare experience at this level to work with others of a different culture" and "this is a real world situation. It prepares people to work together better."

Study A [3–6], which focused on the software development process, has investigated in detail the communication (email and Internet Relay Chat) from the 2000 instance. Among the several issues being studied are the software development process throughout the project timeline, specific team actions and their timings (identified through a category system), and team roles and their effects on performance. Because this is work in progress, there are no definite conclusions as yet; however, the work has already led to some interesting preliminary results. For example, the low performing teams had a higher percentage of emails vs. IRC, while the opposite was true for the high performing teams. The percentage differences in emails and IRC between the high and low performing teams was not significant. Further analysis will include significance testing and studying the email vs. IRC usage over time.

3.4 Runestone 2001

The 2001 instance of Runestone involved 86 students and introduced several changes in response to issues discovered during the year 2000 project. As in previous years, the teams worked on the Brio task. This time, rather than having the option to start from scratch, the teams were asked to enhance one of the projects from Runestone 2000. While the concept of deliverables remained as an important part of the process, the deliverable tasks were changed to accommodate the project enhancement scheme. The deliverables during the first three weeks of the project were the same for all teams. These deliverables involved team setup, analysing the existing software, and proposing extensions. The deliverables during the last part of the project varied from team to team, depending on the changes and enhancements they agreed upon with their teacher.

As in the Runestone 2000 instance, team milestone reports were required An added dimension to these reports during 2001 was to report team progress using a Gantt chart. The purpose of adding team progress to each milestone report was to encourage teams to work consistently and not fall behind. During the 2001 instance, teams received more guidelines and advice on communicating with team members, on dealing with problems, and on writing and presenting reports for each milestone. Web cameras gave students the opportunity to see each other and instructors worked hard to create a "home feeling" in the project room.

To date, only preliminary analysis has been done on the data from the 2001 instance. Student feedback from the end of the course suggests both negative and positive attitudes. Of the problems reported, some students mentioned that they would prefer "to work on a project from scratch, not to just improve some project from last year." Others suggested, "Change the project itself, but keep the idea of international collaboration." Problems with the hardware were again cited for the same reasons as the previous year.

The students' positive remarks were encouraging for the project as a whole. These comments included: "I feel that I have personally benefited from working with counterparts in a foreign country. I would highly recommend that this action continue" and "it helps teach the idea of software development and communication better."

Continued analysis on the data from the year 2001 project will allow further refinement and improvements to the project for future years. Course instructors currently are planning the next offering and will concentrate on process rather than product. The Brio project will not be used.

4. OUTCOMES

Over the years, the Runestone Project has evolved from being simply the idea of giving students international experience in a project setting to a reality where students can work, learn, share ideas, and socialize with peers in other countries. The project has run successfully for the past 4 years and has allowed us as researchers and educators to learn from the experiences of the more than 225 students who have participated in this project. Although there have been problems, we view these as opportunities for learning and for improving future courses. The technical, team, and cultural lessons we have learned are categorized and summarized in sections 4.1 through 4.3 below.

4.1 Technical

Technology is constantly changing. Through the students' experiences, we have learned that technology in teaching institutions also must improve. Study A, which was summarized briefly in section 3.3, has considered the use of different media for communication among remote team members. The teams involved in the Runestone project have provided the opportunity to investigate which, if any, media are best used for remote team communication.

4.2 Team

When grouped for the purpose of working together in a traditional face-to-face environment, people go through a predictable series of development phases during the project's lifecycle [1, 12–14, 17]. For example, the Tuckman model [17] identifies five stages of group development: forming, storming, norming, performing, and adjourning. In the Runestone project, we have had the opportunity to see many teams develop and work through these stages. Some teams have gone through the entire development cycle while others have gone through only some of the cycle. For the latter teams, the causes have centered on team disagreements, misunderstandings, and poor communication. These preliminary findings are consistent with a study done by Spargo and Kelsey [16] that showed some distributed student teams either bypass the storming stage or have that stage occur out of the normal sequence. Study B, which is discussed briefly in section 3.2, is studying the group development process in distributed teams and the reasons why some teams may not develop fully.

One discovery from Study A is that in some of the teams, the student who received the highest grade in a team was also the individual who had the greatest frequency of communication events [5, 6]. This can be summed up by the advice given by one student from year 2001: "You can never communicate too much, especially with the professors."

4.3 Cultural

The physical division of team members on two sides of the Atlantic does not seem to have caused problems with students in this study. The students appear to have worked through the time differences fairly quickly. Although there were cases when the timing of meetings was inconvenient for some team members ("too early" or too late - "I am not a night person"), in general team members tended to respect each other's schedules and time differences.

Humour seemed to be a binding factor in some teams. Because American television programs are available worldwide, students were able to discuss and understand humour from programs such as The Simpsons. Wherever humour was used that was not understood by the foreign counterpart, the team members took time to explain the joke. One year 2000 student said this about humour:

> *We from the very beginning of the project induced humor to the conversation/meetings/project. The humor made the conversation over IRC (chat) easy and swept away any feeling of nervosity.*

Because the Swedish students were all fluent in English, language did not become a problem. On the other hand, there were occasions when the Swedish students would used a shared forum to communicate with one another in Swedish; in these instances, they then tended to translate the meaning for their US counterparts.

The students in the Runestone project have shown us that international collaboration is not only possible but desired and necessary. Cultural boundaries are disappearing. The students involved throughout the Runestone project lifecycle have taught us that there is a yearning to learn and experience other cultures. Two comments from students about working with remote team members sum it up quite well:

> *I think that we were interested in the differences between us and the similarities between us and that helped open a good channel of communications.*

> *Even if you don't learn much by the project itself, you learn a lot about yourself, and the others in the group, and what problems you can encounter when trying to work with someone from another continent.*

5. CONCLUSION

From a student's point of view, there are many lessons that can be learned from the Runestone project. These include how to build better software, how to work together in groups, how to work remotely with other people, and how to work with people from different backgrounds and different cultures. There are many benefits to a student's involvement in an international project like Runestone. One year 2000 student stated,

> *When I interviewed with (a named company) in the DC [Washington, DC area] they were very impressed by the project and my involvement. They hired me!*

From a teacher and researcher's point of view there are also many lessons that can be learned from the Runestone project. These include: how to work collaboratively with other teachers and researchers to design and modify courses, how to teach software

development in a distributed environment, and how to accommodate students with different educational backgrounds.

Projects like Runestone provide students with opportunities they would not normally have in a typical educational environment. For educators and researchers, Runestone provides the opportunity to participate in the evolution of learning and curriculum issues – creating something better for our students each year. What an exciting learning experience!

6. ACKNOWLEDGEMENTS

The authors are indebted to Dr. Vicki Almstrum, University of Texas at Austin for her editorial genius and encouragement. The Runestone project involves students and faculty at Uppsala University (Sweden), and Grand Valley State University (Michigan, USA), as well as researchers from the Open University (UK), the University of Kent (UK), St. Edward's University and the University of Texas at Austin (Texas, USA). The contributions of Bruce Klein, Carl Erickson, and Arnold Pears have been invaluable. A very special thank you to all the students who have participated in the Runestone project.

REFERENCES

[1] Bales, R.F., and Strodtbeck, F.L. 1951. Phases in Group Problem-Solving. *Journal of Abnormal and Social Psychology 46*, 485-495.

[2] Daniels, M., Petre, M., Almstrum, V., Asplund, L., Bjorkman, C., Erickson, C., Klein, B., Last, M. 1998. RUNESTONE, an International Student Collaboration Project. In *Proceedings of IEEE 1998 Frontiers in Education Conference,* Tempe, AZ, November 1998.

[3] Hause, M.L. "Communication in Remote Group Working." February 1999. Online. Internet. [31 October 2001]. Available WWW: http://www.cs.utexas.edu/users/csed/doc_consortium/DC99/hause-abstract.html New Orleans 1999.

[4] Hause, M.L. "Communication in Remote Group Working." March 2000. Online. Internet. [21 June 2001] Available WWW: http://www.cs.utexas.edu/users/csed/doc_consortium/DC00/#info.

[5] Hause, M.L., Last, M.Z., Almstrum, V.L., Woodroffe, M.R. 2001. Interaction Factors in Software Development Performance In Distributed Student Groups in Computer Science. *SIGCSE Bulletin 33(3)*, 69-72.

[6] Hause, M.L. & Woodroffe, M.R. 2001. Team Performance Factors in Distributed Collaborative Software Development. In *Proceedings of the 13th Psychology of Programming Interest Group*. Bournemouth, UK. April 2001, 71-82.

[7] Last, M. "Virtual Teams in Computing Education." February 1998. Online. Internet. [31 October 2001]. Available WWW: http://www.cs.utexas.edu/users/csed/doc_consortium/DC98/last.pdf.

[8] Last, M. "Virtual Teams in Computing Education." March 1999. Online. Internet. [31 October 2001]. Available WWW: http://www.cs.utexas.edu/users/csed/doc_consortium/DC99/last-abstract.html.

[9] Last, M, "Virtual Teams in Computing Education." March 2000. Online. Internet. [31 October 2001]. Available WWW: http://www.cs.stedwards.edu/~lastm/Doctoral_Consortium_2000.htm.

[10] Last, M. "Virtual Teams in Computing Education." February 2001. Online. Internet. [31 October 2001]. Available WWW: http://duke.csc.villanova.edu/docConsortium/DC01/participants/index.html.

[11] Last, M.Z., Almstrum, V.L., Daniels, M., Erickson, C., Klein, B. 2000. An International Student/Faculty Collaboration: The Runestone Project. *SIGCSE Bulletin 32(3)*, 128-131.

[12] Mills, T.M. 1967. *The Sociology of Small Groups*. Prentice-Hall, New Jersey:

[13] Poole, M.S. 1981. Decision Development in Small Groups I: A comparison of two models. *Communication Monographs*, *48,* 1-24.

[14] Poole, M.S. 1983. Decision Development in Small Groups III. A multiple sequence model of group decision-making. *Communications Monographs 50*, 321-344.

[15] Pournaghsband, H. 1990. The Students' Problems in Courses with Team Projects. *SIGCSE Bulletin 22(1)*, 44-47.

[16] Spargo, L., & Kelsey, B. "How Two Universities Crossed the Border." 1996. Online. Internet. [January 12, 1999]. Available WWW: http://www.uni-freiburg.de/rz/inet96/c8/c8_1.htm

[17] Tuckman, B. 1965. Developmental Sequence in Small Groups. *Psychological Bulletin, 63*, 384-399.