A Primer for Applying Service Learning to Computer Science

Pete Sanderson and Ken Vollmar
Computer Science Department
Southwest Missouri State University
Springfield, MO 65804
{PeteSanderson, KenVollmar}@mail.smsu.edu

Abstract

Service learning is an educational philosophy that promotes active learning through community service. We have recently applied this approach in our computer science curriculum, specifically to our software engineering course. In order that other computer science departments can benefit from our experience, we have developed a primer one can follow to establish a program for service learning in the computer sciences. We also describe and assess our experience after one year of applying service learning to software engineering.

1 What is Service Learning?

Service learning is an educational philosophy based on the integration of classroom concepts and related community service experience. One key is reflection; Hatcher and Bringle [2] emphasize that students should "reflect on their (community) involvement in such a way as to gain further understanding of course content and of the discipline and its relationship to social needs and an enhanced sense of civic responsibility." These relationships are embodied in the Kolb Learning Cycle, a conceptual framework for experiential learning [3]. Each phase of the cycle represents learning activities: concrete experience → reflective observation → abstract conceptualization → active experimentation. Learning can begin at any phase but the entry point varies from individual to individual depending on learning styles and prior knowledge.

The most critical transition for long-term learning is from concrete experience to abstract conceptualization. Kolb relates them through reflective observation, defined as "the intentional consideration of an experience in light of particular learning objectives [2]." Students should have not only the opportunity to reflect on their experiences, but a framework to guide their reflection.

Service learning is related to active/passive learning theory. The amount you remember is directly proportional to your level of involvement in the learning process. This is often illustrated with the Cone of Learning [1]. Passive learning practices, such as reading or looking at pictures, result in retention rates of 10-30%. Active learning practices, such as giving a talk or doing an activity, increase retention to 70-90%. Service learning relates to a number of active learning strategies, especially collaborative learning and problem-based learning.

The above definition of service learning includes the phrase "related community service." Service learning is not simply academic credit for volunteer work and community service. Successful service learning also requires that the service experience be related to classroom concepts. The emphasis must on learning, not service[8]. Consider this a *principle* of service learning; it is your responsibility as instructor and discipline expert to assure that all persons involved understand it. Institutional support for service learning must come from the academic area (provosts, deans), rather than units such as student affairs or campus ministries [6].

2 Benefits of Service Learning

Service learning experiences benefit students in many ways besides the experiential learning benefits described above. The experiences are great resume builders, having many of the benefits of internships with smaller time commitments. Successful service learning experiences build self-confidence and allow students to "test drive" their career choices before graduation. They also provide valuable training in citizenship, by exposing students to larger societal issues they may have previously avoided or seen only as abstractions. Finally, they connect students to the community outside the university in a way that benefits all.

5 Experience with Software Engineering

Software Engineering, our companion course, has an advanced data structures (e.g. CS 7) prerequisite and is normally taken late in the junior or early in the senior year. It overviews software lifecycle models and activities at all phases of development and maintenance. About one half of the course grade is based on a semester-long team project in software development. Teams of 3-4 students are assigned a project and expected to conduct requirements analysis and specification, prototype development, design development, code implementation and testing, and to produce associated documentation.

Software engineering was selected as our first companion course because the team project naturally fits the service learning philosophy. Students who enroll in Service Learning form a team and develop a customized application for the community partner as their software engineering project. By organizing the service learning experience in this way, not only is its relationship to the companion course explicit, but the community partner also gets a higher quality product than would be expected from an equivalent one credit hour effort (such as an independent study). To date, three students have enrolled each semester (about 20% of all software engineering students).

5.1 Service Learning Course

Our service learning course is Service Learning in Computer Science. It is a one-hour upper division course that counts toward a computer science major's nine hours of computer science electives. The service learning component is graded separately from the software engineering course. It is desirable but not required that the same instructor teach the service learning component and companion course.

A student's grade in our Service Learning course is based on several components:

1) Each student must log at least 40 hours of service work. CASL policy dictates a failing grade if this requirement is not met. The student, the community partner, and a CASL administrator sign log sheets. These hours are to be in addition to those required for the companion course.
2) Each student is required to meet with a CASL administrator at the community partner site for a performance evaluation. This occurs about mid-semester.
3) The community partner is required to submit a student evaluation form provided by CASL.
4) Each student is required to submit a series of essays describing specific aspects of the project and the service learning experience.
5) The team is required to give at least one oral presentation about the project and service learning. This requirement has been met in the past by addressing a student ACM meeting.

5.2 Project Selection and Preparation

Because the project involves development of software applications for external clients, a number of issues should be resolved before project development begins. Many require the instructor's experience and should not be the students' responsibility. The issues addressed here apply to any project developed for an external client, but are particularly important for a service learning project. Community partners frequently are social service agencies that rely on scarce government funding and volunteer labor, and are unlikely to have computer science or data processing professionals on staff to provide support and assistance.

1) Select or at least approve the project. Communicate with the community partner to assure that the project:
   a) Meets the main service learning goal of applying concepts taught in the course.
   b) Is of appropriate scale for the course project.
   c) Is of appropriate level of difficulty for the course and students involved.

2) Make sure that the community partner is prepared to devote resources. You need answers to these questions.
   a) What computer equipment and development software do they have or plan to acquire?
   b) What limitations are there on student access to their computer facilities?
   c) Who will be their project liaison, so students can have a single point of contact for decisions?
   d) Make sure they understand the liaison will have to devote time to the students and the project, especially during requirements analysis, specification and prototyping.

3) Make sure the community partner understands the limitations of a one-semester project. Negotiate the answers to these questions.
   a) How much maintenance (if any) will be provided? If corrective or perfective maintenance is desired, the students may have graduated and moved away.
   b) How much training will be provided?
   c) What documentation will be included?
   d) What if the project fails to be completed, or does not meet the requirements?

4) Write a license agreement for the community partner to sign. A sample agreement is available at http://www.cs.smsu.edu/~pete/csc300. It should cover the issues from the previous list, plus liability and other disclaimers. Privacy protection is especially important for social service agencies whose client databases contain sensitive and private individual

use has a list of questions for the student to answer concerning their experience so far, a list of questions for the agency to rate the student, and a comments section.

The CASL office has been quite pleased with the computer science students and projects. Both teams have been featured in the CASL newsletter and in the summer 1999 edition of the faculty/staff/alumni newsletter. We have also generated some publicity through a previous conference publication [7].

6.4 Faculty Assessment

Students completing the service learning component are assigned a letter grade based on the criteria outlined in Section 5.1. The CASL requirement of 40 hours of on-site service is difficult to meet since it is in addition to any requirements of the companion course. Students are able to meet this requirement through a number of activities: discussing requirements, iterating through several prototypes, installing software, training agency personnel and diligently documenting the system.

From our experience so far, it is obvious that service learning can be applied to other courses in our curriculum. Our new database course would be suitable of course, as would our networking course. The CASL upper-division requirement prevents us from emulating other efforts such as designing WWW pages for the community as a CS2 project [4]. We and our colleagues will explore other possibilities in the years ahead.

References

[1] Dale, E. *Audio-visual Methods in Teaching*, Holt, Rinehart and Winston

[2] Hatcher, J. and Bringle, R. Reflection: Bridging the gap between service and learning. *College Teaching 45:4* (1997).

[3] Kolb, D. *Experiential Learning: Experience as the source of learning and development*, Prentice-Hall, 1984.

[4] LeBlanc, M. Service learning in computing. *The Journal of Computing in Small Colleges 14:4* (1999), 173-181.

[5] Miller, S. *Science and Society: Redefining the Relationship*, published by Campus Compact (available from Campus Compact, Box 1975, Brown University, Providence, RI 02912), 1996.

[6] Morton, K. Issues related to integrating service-learning into the curriculum," in *Service-Learning in Higher Education*, Barbara Jacoby and Associates (Editors), Jossey-Bass Publishers, San Francisco, 1996.

[7] Sanderson, P. Service learning in computer science, *The Journal of Computing in Small Colleges 12:4* (1999), 341-344.

[8] Zlotkowski, E. Linking service-learning and the academy. *Change*, January/February 1996.

Big Dummy's Guide to Service Learning www.fiu.edu/~time4chg/Library/bigdummy.html
Campus Compact www.compact.org
Eastern Michigan University www.emich.edu/public/office_asl/home.html
Indiana University/Purdue University, Indianapolis www.psyunix.iupui.edu/service.htm
National Service-Learning Clearinghouse www.nicsl.coled.umn.edu
Rutgers University www.scils.rutgers.edu/case/case.html
Science, Engineering, Architecture, Math, CS (SEAMS) www.compact.org/awards/seams.html
Southwest Missouri State University CASL www.smsu.edu/casl
Stanford University haas.stanford.edu
University of California - Berkeley www-gse.berkeley.edu/research/slc/ServiceLearning.html

Table 1. Selected Service Learning WWW Resources. List current as of December 1999. Check www.cs.smsu.edu/~pete/csc300 for updates.

Question	*SL*	*EXT*	*INST*
Requirements analysis crucial	50	40	27
Prototyping crucial	33	67	18
Much more time on requirements vs. coding	67	17	73
Definitely satisfied with project choice	17	100	45
Definitely satisfied with project results	17	80	45
Definitely helped learn course material better	83	67	55
Frequently share experiences outside of class	67	20	55
Frequently share experiences in class	33	20	27
Definitely helped you participate in class	17	33	9
Definitely relevant to course material	50	17	36
Definitely resembles real-world experience	50	50	10
Definitely recommend to future students	83	33	9

Table 2. Percentage of students giving the most positive response to questions concerning the Software Engineering project. Groups are: Service Learning Students (SL), other students with external project clients (EXT), students with instructor as client (INST).