INTERNATIONAL JOURNAL OF INSTRUCTIONAL TECHNOLOGY AND DISTANCE LEARNING

April 2007
Volume 4 Number 4

ISSN 1550-6908

PUBLISHER'S DECLARATION

Research and innovation in teaching and learning are prime topics for the *Journal of Instructional Technology and Distance Learning* (ISSN 1550-6908). The Journal was initiated in January 2004 to facilitate communication and collaboration among researchers, innovators, practitioners, and administrators of education and training involving innovative technologies and/or distance learning.

The Journal is monthly, refereed, and global. Intellectual property rights are retained by the author(s) and a Creative Commons Copyright permits replication of a small number of articles for education related purposes. Please request permission from the author and the Journal if extensive replication is planned or commercial gain is involved.

Publication is managed by DonEl Learning Inc. supported by a host of volunteer editors, referees and production staff that cross national boundaries.

IJITDL is committed to publish significant writings of high academic stature for worldwide distribution to stakeholders in distance learning and technology.

In 3 years the Journal has over 1.5 million page views and 330,000 downloads of monthly journals and eBooks.

International Journal of
Instructional Technology & Distance Learning

Vol. 4. No. 4.

ISSN 1550-6908

Table of Contents – April 2007

Editorial

Idea-Center or Cost-Center

Donald G. Perrin

Technology is a major driver for change and paradigm shifts in business, industry, government, and education. Each responds differently to change. Business is concerned with profits and market share; Industry on production cost and quality; Government on regulation and human services; and Education on ideas and social impact. Collaboration between these four areas (BIGE) facilitates change; philosophical differences are impediments to the change process.

An innovation that benefits one group may be detrimental to another. The development of civilization is largely due to "survival of the fittest" whether in health and physique, language, warfare, trade, productivity, systems of government, politics, economics, education, intellect, or some combination of these. Tensions caused by innovation can be destructive; they can also afford opportunities for alternative changes that bypass or replace the roadblock.

Academia is a crucible in which ideas are developed, tested, and propagated. Business and industry are more concerned with practical applications, marketing and sales. Sometimes there is collaboration with academia and sometimes they compete. Government's role is to stimulate collaboration when it is in the national interest and to arbitrate on infractions or differences. At the end of the day, all of these groups are dependent on each other.

Technology has made piracy a way of life for individuals and nations who otherwise could not afford or have access to the benefits of that technology, and also for the greedy. Traditional methods of control such as legal action or making the product "copy proof" have had little affect. In some instances technology and innovative business practices provided a solution. For example, mass marketing of movies on videotape reduced the cost so low that piracy was less attractive. Competition between video rental and cable television expanded markets and reduced cost. Extraordinarily low rental cost for DVDs, now facilitated by mail exchange, makes ownership or piracy even less attractive. Similarly, Internet delivery of films to theatre screens or home theatre further reduces cost, improves security, technical quality and service and stimulates consumption.

The Internet facilitated sharing of music at an unprecedented level through NAPSTER and similar services. A successful legal action against NAPSTER and individuals had little impact. It was eventually recognized that NAPSTER had pioneered a new business model. The Apple IPod emerged and was tremendously successful. It made music accessible and portable so people could listen to their favorites at any time anywhere. Again it was based on low cost and delivery via the internet. But the public wanted more – higher quality, and ability to copy to other devices. On April 2, 2007, Steve Jobs announced a change that will affect the entire music industry:

> *EMI Music's entire digital catalog of music will be available for purchase DRM-free (without digital rights management) from the iTunes® Store (www.itunes.com) worldwide in May. DRM-free tracks from EMI will be offered at higher quality 256 kbps AAC encoding . . . customers will have the ability to download tracks from their favorite EMI artists without any usage restrictions that limit the types of devices or number of computers. . . Purchased songs can (now) be played on … iPods, Mac® or Windows computers, Apple TVs and soon iPhones, as well as many other digital music players.*
>
> *The iTunes Store features the world's largest catalog with over five million songs, 350 television shows and over 400 movies. The iTunes Store has sold over two billion songs, 50 million TV shows and over 1.3 million movies, making it the world's most popular online music, TV and movie store.* http://www.apple.com/pr/library/2007/04/02itunes.html

The lead-article this month is about open source software. In my recent visit to Australia I was surprised to find computer software from American companies such as Microsoft and Adobe was notable for its absence in many institutions of higher learning. Compared to the United States, the cost of these products was greatly out of proportion to personal income and academic budgets. There was obvious resentment against these highly profitable companies and tremendous interest in development of open-source software.

Apple computer made major inroads into academic markets in its early days by providing extensive services and price breaks for academia, which was a major stimulant to customer loyalty and future business. Microsoft realized that academia and its graduates needed major incentives to adopt Microsoft products, and in 1997-98 initiated a program priced so low that it would ensure Microsoft was a dominant force in education and training in the United States.

As suggested earlier, the greatest benefits of technology result when BIGE - business, industry, government, and education – collaborate. Open source is not necessarily in competition with commercial development, as the following example illustrates:

> *Linux and IBM have partnered to drive the open standards revolution, moving business beyond the limits of proprietary software and hardware solutions. The entire IBM Systems product line is Linux enabled, making it easy for any size business to take advantage of the power of open standards. And because Linux is an open operating system it benefits from a broad developer base, making it one of the fastest growing operating systems in the world. To simplify your business even further, Linux supports greater and growing interoperability between diverse software and hardware technologies, which leads to a reduction in IT management costs.*
>
> http://www-03.ibm.com/systems/browse/linux/

Editor's Note: In academia, growth occurs through sharing and interaction; business growth is stimulated by competition and financial rewards. Knowledge belongs to the people, while business preserves its investment and profits with patents and copyrights. Business and academia depend on each other. For example, publishers rely on academia for authors and employees while academia relies on publishers to produce textbooks that become the course curriculum. Unfortunately, the high cost of textbooks is a problem for less affluent students. The same is true of commonly used computer software. This paper explores the purpose and historic roots of Free and Open Source Software (FOSS) in academia and in industry.

A Socio-Cultural Perspective on the Free and Open Source Software Movement

Guohua Pan, Curtis J. Bonk
Canada and United States

Abstract

This paper examines the history, leading figures, and sub-culture of the Free and Open Source Software (FOSS) movement from a socio-cultural perspective. It makes the argument that the evolution of FOSS movement is an interactional process between the socio-cultural environment and the ideas, negotiations, proposals, strategies, and overall leadership of leading figures of the FOSS movement. The openness and distributed nature of FOSS movement has already played a vital role in the current swell of knowledge sharing in both higher education and industry. As a social, political, and economic flattener of the 21st century, the FOSS movement may evolve into a powerful cultural artifact or tool that can help in efforts to break down walls of mistrust, resentment, or animosity among competing individuals or groups. It might also flatten the world of learning by providing freely available and widely accessed teaching and learning tools.

Keywords: open source, FOSS, sociocultural, constructivism.

Introduction

A special report on open-source business appearing in the March 18, 2006 issue of the Economist argued that the open source model of software development "has moved far beyond its origins" and that "open-business practices have emerged as a mainstream way for collaboration" (The Economist, 2006, p. 73). Research on "free and open source software" (FOSS) development is now flourishing and across disciplines. A Google search of the word 'open source' on August 7, 2006 returned 1,690,000,000 web page hits covering science, engineering, bio-technology, business management, and humanities. A salient feature of free and open source software (FOSS) is its openness and distributed development of the software.

This FOSS feature of openness, or 'gift' culture, and distributed development arose from the interplay of standing norms and institutions (the academy), leading figures, and technical/hacker traditions. Knowledge sharing has long been a tradition at the institutions of higher learning. Graduate students and faculty share their discoveries at seminars and conferences, and publish their research findings at conference proceedings and peer-reviewed journals. It was noted that the majority of participants in FOSS projects, that is, those who contributed freely to FOSS projects, were from universities and research labs (Kelty, 2001).

Accommodated in the sharing culture at various institutions of higher learning, sub-cultures such as hacker culture were seeded and sustained that advocate craftsmanship of computing and that value free availability of information. This hacker culture nurtured prominent FOSS figures like Richard Stallman, Eric Raymond, Linuz Torvalds, and Martin Dougiamas, helped them to initiate their FOSS projects with a new mode of software development, and cultivated them into FOSS leaders.

The gift culture and distributed development orientation of the communities of FOSS users have enabled the creative and broad participation of people, including people from educational institutions, in the development of FOSS. While FOSS helps higher education institutions and organizations to address the dual challenges of utilizing resources that are financially sustainable and developing innovative products and software programs that are geared to the needs of users, it remains a key question as what special implications the openness and distributed development model of FOSS may hold for teaching and learning within institutions of higher learning.

In seeking to address the question, a literature review on social constructivism and learning was conducted. From a socio-cultural perspective, this paper then examines the socio-cultural environment in which some of the prominent FOSS leaders studied and worked, the hacker culture that grew out of the socio-cultural environment, the tools, artifacts, and language used in FOSS projects, some representative FOSS figures, and the interplay between them. Doing so, this paper is intended to contribute to ongoing research into the FOSS movement and teaching and learning.

Socio-cultural views on learning

Constructivist theory is largely based on the work of Dewey and psychologists such as Vygotsky, Piaget, and Bruner (Bereiter & Scadamalia, 1996; Kearsley, 2004; Mahoney, 2004). Under a constructivist epistemology, knowledge is constructed or generated by individuals in their learning pursuits and explorations. Knowledge construction is not static nor does it take place inside an individual's head in isolation. Instead human thinking is distributed in society (Bonk & Cunningham, 1998); as Pea (1996) notes, the human mind does not work alone. Instead, all cognitive functions, according to Vygotsky (1978), originate in and must be explained as products of social interactions. From this perspective, knowledge construction is a process that takes place concurrently between an individual and the people around him. Long ago, John Dewey pointed out the importance of external influences on internal meaning constructions:

> *…we live from birth to death in a world of persons and things which is in large measure what it is because what has been done and transmitted from previous human activities. When this fact is ignored, experience is treated as if it were something which goes on exclusively inside an individual's body and mind. It ought not to be necessary to say that experience does not occur in a vacuum. There are sources outside an individual which give rise to experience. (Dewey, 1938, p. 39)*

From this perspective, knowledge is thus said to be both socially and individually constructed. Knowledge construction is a process by which individuals are integrated into a knowledge community, or a community of practice (Lave & Wenger, 1991). In effect, knowledge and reality are actively constructed by social relationships and interactions. Knowledge construction is a situated experience; it is a dynamic psychological process among individuals of their shared emotions, experiences, and representations when interacting with others and interpreting new ideas, perspectives, or situations on the basis of existing knowledge. It is contingent on convention, human perception, and social experiences. It must take into consideration the social, economic, legal, political, and cultural dimensions surrounding the learners and the learning process (Pfaffenberger, 1998).

This construction of new knowledge takes place within an enculturation process where one might be apprenticed by an expert, coach, mentor, or more knowledgeable other. Beyond the immediate social environment is the wider context of cultural influences. According to Vygotsky (1978), the mental functioning of an individual is inherently situated in social interactional, cultural, institutional, and historical contexts. The existing institutional settings and cultural artifacts, therefore, have significant bearings on meanings created and expressed (Bonk & Kim, 1998).

Socio-cultural settings shape the mind through the artifacts employed. While some in the psychological and educational literature prefer the term social artifact, for our purpose, we use term cultural artifact. A cultural artifact is an object that embodies the values and beliefs of its creators and users (e.g., a piece of software for concept mapping from the company Inspiration or an iPod from Apple Computer). In other words, what a creator (e.g., Apple Computer) thinks and believes is embodied in the cultural artifact (e.g., the iPod) that s/he creates while a user may accept or hold the values and beliefs of the cultural artifact through using it (e.g., to listen to music or record one's own music).The mind interacts with the artifacts in order to make 'sense' of everyday reality. As Lave and Wenger (1991, p. 53) point out, communities of practice "are part of broader systems of relations in which they have meaning. These systems of relations arise out of and are reproduced and developed within social communities."

From a socio-cultural point of view, knowledge is negotiated by two or more individuals through give-and-take, often in the format of discussion, debate, and interactive reasoning (Jonassen, 1999). Groups of individuals interact with each other and negotiate meaning out "a common ground of interest and understanding" in their work or study (Bonk & Cunningham, 1998, p. 31). This group or community of practice consists of individual members with shared interests, goals, and knowledge (Lave & Wenger, 1991). New meaning, ideas, and values generated or promoted must be accepted by the members of the community of practice before they are considered valid. Although each member has her or his own personal preferences, beliefs, and values, there is a set of important understandings that members of the community share in common.

These shared understandings consist of the norms, values, attitudes, beliefs, and paradigms of that community (Bonk, Wisher, & Nigrelli, 2004). Such principles underlie the culture of this community of practice. The culture, albeit implicit, is grounded in the same experience that the members of this community of practice share and a concomitant knowledge as how they should be able to perform (Cole, 1996; Fisher, 2005; Mahoney, 2004). The more that the members of the community of practice think the same way and understand the reality from a common basis, the more they will trust members to perform in the same way that they would (Cole, 1996; Fisher, 2005; Viseu, 1999).

Admittedly, one of the most troubling concerns about constructivism is its lack of empirical findings: "[constructivist] theory and conjecture far outstrip empirical findings" (Driscoll, 2000, p. 395). For those operating from a socio-cultural or social constructivist orientation or perspective, a social group, or a community of practice, becomes relevant when the artifact at hand has some meaning to that community (David 2003). "A 'problem' is a 'problem' only inasmuch as there is a social group [community of practice] that perceives it as such" (Pinch & Bijker, 1987, p. 30).

In the next section we review the environment in which free and open source movement was born and grew, including unique and insightful perspectives on this new and still emerging field from the actual leaders of the free and open source movement. In addition, we attempt to link this movement in the field of computer science to the recent heightened interest in and growing acceptance of socio-cultural theory and social constructivism within the fields of education and psychology.

Social-cultural environment and leaders of free and open source movement

The free software movement preceded open source software movement. The creation of Free Software Foundation (FSF) in 1985 marked the beginning of free software movement. In addition, the foundation of the Open Source Initiative (OSI) in 1998 proclaimed the beginning of open source software movement. A spirit of sharing of both source code and software development was the central thesis in the free software movement and is manifested in the electronic artifacts of FSF. At the Free Software Foundation Web site, for example, FSF defines the connotation of the word “free” in “free software” as the freedom for software users “to run, copy, distribute, study, change and improve the software” (Free Software Foundation, 2006). Of those six verbs, the first three, that is, ‘to run, copy, and distribute’ encourage individuals to use (freely), or in other words to use the product (free software) and share it with others. The other three verbs, that is, ‘to study, change, and improve’ may be interpreted as actually inviting users to share their ideas and perhaps their skills as well of improving the software.

This belief in knowledge sharing and distributed development not only permits but also encourages interested software users to become involved with the continuing development of the free and open source software (FOSS). Such involvement includes disseminating the inner workings of computing and cultivating a community of FOSS users who may become knowledge disseminators. This spirit of sharing is further witnessed in FSF’s (1991) clarification of GNU/GPL that “…the GNU General Public License is intended to guarantee your freedom to share and exchange free software.

Likewise, the spirit of sharing is advocated in the open source movement. Recently, OSI (2006) announced that the basic idea behind open source is that “programmers can read, and modify the source code for a piece of software” and that the programmers and users are entitled to “improve, adapt, and fix bugs” of the software and “redistribute” part or all of it. This belief in freedom of sharing within communities of the FOSS movement can be traced to the environment under which leaders of the FOSS communities studied and worked; in effect, traced to the institutions of higher learning and everyday environments where most of the FOSS movement leaders studied and worked when they initiated their FOSS projects.

Though not always explicitly stated or enthusiastically supported, sharing knowledge and culture is a deep-rooted tradition among the institutions of higher learning (Rhodes, 1999). Scholars are “stewards entrusted” by the public “with resources to feed and nurture the world of ideas and innovation” through researching, teaching, and publishing those ideas (Hilton, 2005, p. 73). Faculty share their discoveries, new theories, research results, and new methods in academic publications such as peer-reviewed journals, professional conference proceedings, and, increasingly, electronic journals, personal blogs, and personal homepages and Websites. Sharing enables a member of a community to let others know what she is doing and whether what she is doing is worthwhile, and vice versa.

The Johns Hopkins University, the first research university in the United States, was a leader in promoting the academic freedom of sharing and the norms of ‘open science’ (Feldman & Desrochers, 2003; Merton, 1979). When Hopkins’ scientists discovered restriction enzymes, one of the bases of the biotechnology industry, the university put the discovery in the public domain for sharing, which might have actually sped up science and the leading role American industry played in this field (Brody, 1999). Without a doubt, when sharing information, including new discoveries made on the basis of existing work, those in academia can more promptly discuss and debate different inventions, ideas, and trends. At the same time, students are encouraged to participate in the discussions and contribute their perceptions of the subjects under study. This sharing culture helps to nurture the curious and innovative minds of the millions studying and working at academic institutions, including Richard Stallman, Linuz Torvalds, and Martin Dougiamas, each of whom is discussed below.

Richard Stallman, the leader of free software movement and the founder of FSF, had worked at MIT’s Artificial Intelligence Lab (AI Lab) for over 10 years when he resigned from the AI Lab in 1984. The sharing culture and freedom of inquiry at MIT, and at the AI Lab in particular, cultivated his firm belief in free software, leading to his founding of the FSF in 1985, his development of GNU (“Gnu’s Not Unix”), and his tireless promotion of free software. Stallman “recognized that his personality was unyielding to the give-and-take of common human interaction" (Levy, 1984, p. 422). Stallman's staunch belief in sharing is evidenced in an early manifesto:

> *The Free Software Foundation is dedicated to eliminating restrictions on copying, redistribution, understanding and modification of software. …*
>
> *When it is released, everyone will be permitted to copy it and distribute it to others; in addition, it will be distributed with source code, so you will be able to learn about operating systems by reading it, to port it to your own machine, to improve it, and to exchange the changes with others." (GNU Bulletin, 1987)*

The academic freedom of sharing at MIT not only helped to shape Stallman's belief in free software but also allowed him to put such beliefs in action. When he worked at the AI Lab, Stallman wrote his best known FOSS work "the first extensible EMACS text editor – the extensible, customizable, self-documenting real-time display editor." In addition, he helped develop a few other programs for the later GNU/GPL operating system such as the first free software operating system, although he may have been doing something not in his job description (Stallman, 2006). As cited in Levy (1984, p. 422), Stallman's free distribution of EMACS brought to life a vision of sharing which he expected many others to follow as seen in the following quote: "that they give back all extensions they made, so as to help EMACS improve. … As I shared, it was their duty to share; to work with each other rather than against each other."

Linuz Torvalds, the developer of initial Linux kernel and owner of the Linux trade mark, is another leader of FOSS movement. A kernel consists of the core code of the operating system which is responsible for providing secure access to the machine's hardware and to various computer processes. Torvalds himself is a beneficiary of the open and sharing culture of institutions of higher learning. For example, Torvalds acknowledges that while working at the University of Helsinki, the university knew and supported his work on Linux although "doing Linux wasn't part of his job description" (FM, 1998). And it was in this sharing culture and academic environment that the well-known and sometimes inflammable Usenet discussion between Torvalds and Dr. Andrew Tanenbaum took place (see quote below). Dr. Tanenbaum was and still is a respected computer science professor at Vrije University in the Netherlands and author of Minix:

> *MINIX is a microkernel-based system. The file system and memory management are separate processes, running outside the kernel. The I/O drivers are also separate processes (in the kernel, but only because the brain-dead nature of the Intel CPUs makes that difficult to do otherwise). LINUX is a monolithic style system. This is a giant step back into the 1970s. That is like taking an existing, working C program and rewriting it in BASIC. To me, writing a monolithic system in 1991 is a truly poor idea. (Tanenbaum, 1992)*
>
> *MINIX is a microkernel-based system. [deleted, but Not so that you miss the point] LINUX is a monolithic style system. If this was the only criterion for the "goodness" of a kernel, you'd be right. What you don't mention is that minix doesn't do the micro-kernel thing very well, and has problems with real multitasking (in the kernel). If I had made an OS that had problems with a multithreading filesystem, I wouldn't be so fast to condemn others: in fact, I'd do my damndest to make others forget about the fiasco. (Torvalds, 1992)*

Minix is one of the earliest free Unix-like operating systems. Torvalds was a computer science student at the University of Helsinki when the fairly public debate took place. Although they still maintain their respective viewpoints to this day, Tanenbaum's account of their interchanges during an interview with Ken Brown (Tanenbaum, 2004) shows that there is no animosity between Torvalds and Tanenbaum.

A third leader in this field is Martin Dougiamas, who is the creator and original developer of Moodle. Moodle is the first open source learning management system that claims to use a social constructivist orientation. Dougiamas created Moodle while working as a WebCT administrator at Curtin University of Technology in Australia. Dougiamas (1998) acknowledges the influence of socio-cultural factors on people's thinking and believes that cultural factors affect, in various forms of activity, what and how people think. The academic environment at Curtin University and his disenchantment with the closed system of WebCT certainly helped to shape his social constructivist orientation for the Moodle system as well as his open source learning management system initiative.

From AI Lab at MIT, to University of Helsinki in Finland, to Curtin University in Australia, Stallman, Torvalds, and Dougiamas either studied or worked at their respective institutions; naturally, such institutions were part of a wider community with their cultural practice and social norms. The practice and norms were embodied in the actions and activities that the members of the communities engaged in. As members of their respective communities participating in various activities and utilizing many resources and tools, Torvalds and Dougiamas interacted with their peers, accepted the values of the community, observed the norms, and were able to grow from peripheral members of their communities into experts and initiate their FOSS projects in the process. They were apprenticed into this community undoubtedly through countless e-mail exchanges by experts within that community who answered their initial questions and concerns (e.g., the e-mail exchanges noted earlier between Torvalds and Tanenbaum). As members of the FOSS movement, they also benefited from code enhancements, advice, and suggestions from more knowledgeable peers who continue to refine Linux and Moodle. Unlike a flesh and blood community, they can be influenced by anyone on the planet who has a brilliant idea for refining their code and access to the Internet. As Wertsch (1991) notes, mental functioning and the tools and artifacts humans create are situated in existing social, cultural, and institutional contents. Unlike Torvalds and Dougiamas, Stallman, on the other hand, was an expert at the center of the community when he began the FOSS movement (i.e., he was not unknown). Still, he had many years at the AI Lab at MIT where he likely was influenced by many other people.

In the next section, we analyze hacker culture and its impact on FOSS.

Hacker culture and its impact on FOSS movement

To a large extent, hacker culture has the greatest influence on free software movement and subsequent open source movement. Hacker culture is a loose term covering a networked collection of subcultures (Raymond, 1992). The hacker culture in this case is a voluntary subculture that originated from MIT's computer culture in late 1950's. The original hacker culture was one of craftsmanship. Logically, computer professionals (hackers) wanted to be responsible for a project from start to finish, beginning with identifying the problem with the client, to writing the code, to operating the machine (computer). The core value of hacker culture is the belief of knowledge sharing – the freedom of exploring computer programming secrets and sharing them with others, free "access to computers – and anything which might teach you something about the way (the) world works – should be unlimited and total" (Levy, 1984, p. 27). Hacker culture values the free availability of information rather than exclusive right and that "the possession by one of information need not deprive another." (Wark, 2004, p. 15).

It was the fertile soil of sharing culture at institutions of higher learning that bred hacker culture. MIT Artificial Intelligence Laboratory (AI Lab), together with Carnegie Mellon University, Stanford University, and the University of California, Berkeley, were the well-known hotbeds of early hacker culture (for more information, see David, 2003; Johnson, 2001; Levy, 1984; Pfaffenberger, 2000; Steel, 1992; Weber, 2004). As a major center for computer software development, the AI Lab at MIT had a culture that valued "openness, sharing, and collaboration." Additionally, the advanced concepts such as networking, file sharing between machines and terminal-independent I/O were pioneered there, while hackers at AI Lab were developing Incompatible Timesharing System (ITS) for the lab's DEC PDP-6 and PDP-10 computers (Hannemyr, 1999; Weber, 2004, p. 46).

Richard Stallman (RMS) was one of the hackers working at AI Lab. For RMS, software was a "manifestation of human creativity and expression… and represented a key artifact of a community … to solve problems together for the common good." (Weber, 2004, p. 47) It was during his work at the AI Lab when RMS and his hacker colleagues expressed their desire for and actually exercised "open systems, integrated solutions and distributed resources" (Hannemyr, 1999). They preferred working in an intimate and interactive way such that they could share what they were working on or caring about, including reviewing and re-using each other's source code (Hannemyr, 1999). However, the introduction of a scientific management method to computer software development changed the way hackers worked and the way computer software was developed and used.

A scientific management method of software development was introduced to the computing community by corporate management in mid-1960s to improve computer programming efficiency through standardization and specialization of work, with the hope of replicating the success of Ford automobile manufacturing and other industrial activities. Broken away from the traditional craftsman way of computing, the hackers were now stratified into a hierarchical one based on their status and seniority - analyst, programmers, coders, testers, maintainers, computer console operators, computer room technicians, key punch operators, tape jockeys, and stock room attendants (Hannemyr, 1999; Levy, 1984). Separated from each other and from computers, the hackers experienced greatly reduced social interaction with their peers thereby reducing the open and free exchange of ideas, and, ironically, limiting or curtailing their access to computers.

Up until the introduction of this scientific management method within the computing field, the intersubjectivity or temporarily shared collective reality (Bonk & Kim, 1998) that RMS and his hacker colleagues likely experienced depended on a craftsman or craftsman-apprentice way of computing. It was their shared social space (Schrage, 1990) for inventing new ideas and testing them against other great minds. Constructing social meanings, including the procedures and norms of computing, therefore, involves intersubjectivity or shared meanings being exchanged among members of that space or community setting. Any personal meanings shaped through these experiences are affected by the intersubjectivity of the community to which the people belong (Rogoff, 1990). When the scientific management method was introduced and implemented, it naturally ignited resistance from hackers since the scientific management method was not built on the intersubjectivity that the hackers had experienced in terms of software development; it was not part of their community norms or expectations. It was at about this time that the ARPANET arrived to help rejuvenate the hacker culture.

ARPANET, the experimental beginning of the Internet, was a small research experiment funded by Advanced Research Project Agency (ARPA) under the US Department of Defense in 1968. ARPANET was initially built to connect computers at ARPA's major research centers and scientific laboratories for researchers to share resources, including computer programs, research results, and electronic mail to specific individuals as well as to massive mailing lists which could also be used for socialization. As a completely new communication architecture, ARPANET was a network of many local networks through which messages were created at any local network and routed through the main network, then left to find their own way to their destination (Hafner & Lyton, 1996; Hannemyr, 1999; Weber, 2004). The ARPANET automatically routed messages around in case a local network or some local networks broke down or were blocked for one reason or another. Importantly, anyone from a local network could add new services to ARPANET and make the new services available to all those connected to the ARPANET. The network was thus distributed (i.e., no network was at the center or no computer or no place at the center); consequently, there was no authority that could control all actions or create or maintain any type of lock-step hierachical factory production model. Almost concurrently with the evolution of ARPANET, the Unix operating system emerged that helped to advance the hacker culture.

Unix is a computer operating system built in 1969 by Ken Thompson, Dennis Ritchie, and Ossanna McIlroy, then three Bell Labs researchers. Written in C language, Unix is a portable, multi-tasking, and multi-user in a time-sharing configuration. Unix allows computer professionals, including the hackers working at MIT, to move chunks of working code from machine to machine and project to project. It is analogous to a toolbox of small and simple modules that can be combined or recombined to create more useful or complex functions (Weber, 2004). Since its inception, various versions of Unix have been built for different purposes or applications. As the quote below indicates, the philosophy behind Unix was the “nuclei of communities” and “fellowship” that would “resonate all through Unix’s subsequent history” (Raymond, 2004, p. 31):

> *What we wanted to preserve was not just a good environment in which to do programming, but a system around which a fellowship could form.*
>
> *We knew from experience that the essence of communal computing, as supplied by remote-access, time-shared machines, is not just to type programs into a terminal instead of a keypunch, but to encourage close communication.*
> *(Ritchie, 1984, p.1577)*

At the time when Unix was built, most users of operating systems were still computer operators/hackers working at research centers. The distributed architecture of the ARPANET, and the access to computer power and creativity through computing with Unix helped to nurture the unorthodox hacker culture of decentralization, sharing, and open standards. When Transfer Control Protocol/Internet Protocol (TCP/IP) networking was integrated into Unix in 1983, computer connectivity was substantially increased, which, in turn, expanded the user base of the ARPANET. It was believed that the integration of TCP/IP networking into Unix lay the foundation of the Internet today. TCP/IP is a set of rules for communication among computers on the Internet.

As noted earlier, mental functions are shaped by the tools and artifacts in our socio-cultural settings (Cole, 1996, p. 333, Wertsch, 1991). Here such tools included Unix, TCP/IP, and common programming languages and procedures. The meanings that computer professionals, and, in particular, hackers, conveyed was linked to the resources within their socio-cultural environment. From socio-cultural perspective, we see that there were specific tools (i.e., computers, pens, pencils, etc., used for communication among hackers), artifacts (i.e., ITS, ARPANET, Unix, etc., created by the hacker culture) and a social environment (i.e., Bell Labs, Carnegie Mellon, MIT, Stanford, UC Berkeley, etc. where most hackers worked) that nurtured the hacker culture and belief in free software. In addition, there was also a language component or shared lingo that was essential to the evolution of the hacker culture. Like the rest of Americans, hackers at American institutions and organizations used English for communication and exchange of ideas. Moreover, the hackers also communicated with each other using some special vocabulary that would identify them as the members of the hacker community. As hackers became increasingly conscious of their culture, they started to collect the frequently used slang terms and put them into printing, first the ‘Jargon File’ (also known as ‘AIWORD.RF[UP,DOC]’) by Raphael Finkel from Stanford University in 1975, then ‘The Hacker’s Dictionary’ by Guy Steele of Carnegie Mellon University in 1983, and ‘The New Hacker’s Dictionary’ by Eric Raymond in 1992.

From socio-cultural perspective, Jargon File, The Hacker’s Dictionary, and The New Hacker’s Dictionary, like ARPANET, Unix operating system, played the role of mediating between the hackers personal meanings of the activities and words and culturally established hacker meanings of the activities and words of the wider society, i.e., the hacker’s society (Cobb, 1994). It was a unique language and vocabulary which mediated their actions; those who did not comprehend this language or were unable to use it, were less equipped to become full members of the hacker community. In effect, these dictionaries and jargon files provide a sense of scaffolding for novices to participate in such a community and help apprentice them into it.

Thus far, we have discussed the effects of tools, artifacts, computer languages, and social environments, among other factors, on the evolution of the hacker culture. In the next section, we discuss the values and beliefs of FOSS movement.

FOSS movement

“Even after other attributes are accounted for, cultural traditions powerfully influence performance and the ability to embrace new initiatives.” (Feldman & Desrochers, 2003, p. 108) In effect, it was the fertile soil of the sharing culture at institutions of higher learning that seeded and nurtured the hacker culture. And it was the evolution of hacker culture that in turn bred and helped blossom the FOSS movement. In particular, the beliefs and values of hacker culture are embodied in the artifacts of FOSS movement. One such artifact is an open letter titled ‘free-Unix’ that RMS published on the Usenet newsgroup net.unix-wizards on September 27, 1983. That single event could be interpreted as the start of the FOSS movement:

> *Starting this Thanksgiving I am going to write a complete Unix-compatible software system called GNU (for Gnu's Not Unix), and give it away free to everyone who can use it. Contributions of time, money, programs and equipment are greatly needed. (Stallman, 1983)*

In this electronic artifact, RMS appealed to the members of his hacker community for sharing their resource in developing the GNU and explained that he was going to share, or in his words “give it away,” the program with other people who like it (Stallman, 1983). RMS (1985) further explained in the GNU Manifesto why he must write GNU:

> *I consider that the golden rule requires that if I like a program I must share it with other people who like it. Software sellers want to divide the users and conquer them, making each user agree not to share with others. I refuse to break solidarity with other users in this way. I cannot in good conscience sign a nondisclosure agreement or a software license agreement. For years I worked within the Artificial Intelligence Lab to resist such tendencies and other inhospitalities, but eventually they had gone too far:*
>
> *I could not remain in an institution where such things are done for me against my will. So that I can continue to use computers without dishonor, I have decided to put together a sufficient body of free software so that I will be able to get along without any software that is not free. I have resigned from the AI lab to deny MIT any legal excuse to prevent me from giving GNU away.*

In 1985, RMS drafted some of his hacker colleagues and created the Free Software Foundation (FSF), “The first software-sharing community”. The purpose of FSF was to develop and distribute software under the General Public License (GPL), or “copyleft,” that protected the right to share using Copyright. Stallman later refined and elaborated on his vision of GNU/GPL software as consisting of four essential freedoms:

1. The freedom to run the program as you wish.
2. The freedom to study the source code and change it to do what you wish.
3. The freedom to make copies and distribute them to others.
4. The freedom to publish modified versions. (Biancuzzi, 2005)

As is manifested in the four essential freedoms, the culture of sharing dominated RMS's vision of GNU/GPL software. He granted users the right to share the free software at their own volition. Furthermore, he also granted the users the right to share the free software with other people by giving them "the freedom to make copies and distribute them to others." While giving them the freedom to share the free software, Stallman also viewed the users as potential developers and invited them to share their knowledge, skills and expertise of the software with him and with people at large by asking them to modify the source code and share the modified versions with others.

The formation of FSF and the publication of GNU Manifesto was a turning point in the history of FOSS movement in that it unified hackers under a new community – the FOSS community of practice with a uniting principle of sharing, thereby transitioning the relatively insular hacker culture into a mass movement. The hackers suddenly obtained a collective identity and a set of values and normative and principled beliefs that constituted the FOSS culture to guide their everyday practices. The FOSS culture thus provided a value-based rationale for contributing to GNU/GPL project. It was with this FOSS community and its culture that GNU/GPL projects such as a compiler and EMACS were started and completed. As has been witnessed, the FOSS community and FOSS culture were able to gather thousands of people of similar values and beliefs across the world to work on GNU/GPL projects such as Linux.

As is popularly known, Linux is a FOSS project initiated by Linus Torvalds, then a hacker studying at University of Helsinki. As mentioned earlier, Torvalds' cultural orientation toward knowledge and code sharing was evidenced in his Usenet discussion with Tanenbaum in 1992. Before that discussion, Torvalds had manifested his immersion in the sharing culture when he posted a message on the Usenet of Minix newsgroup, reaching out his community members for information sharing:

> From:torvalds@klaava.Helsinki.FI (Linus Benedict Torvalds)
> Newsgroup: comp.os.minix
> Subject: What would you like to see most in minix?
>
> Summary: small poll for my new operating system
>
> Message-ID: 1991Aug25, 20578.9541@klaava.Helsinki.FI
> Date: 25 Aug 91 20:57:08 GMT
>
> Organization: University of Helsinki.
>
> Hello everybody out there using minix- I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu) for 386(486) AT clones. This has been brewing since april, and is starting to get ready. I'd like any feedback on things people like/dislike in minix; as my OS resembles it somewhat (same physical layout of the file-system due to practical reasons) among other things. I've currently ported bash (1.08) and gcc (1.40), and things seem to work. This implies that i'll get something practical within a few months, and I'd like to know what features most people want. Any suggestions are welcome, but I won't promise I'll implement them :-) Linus Torvalds torvalds@kruuna.helsinki.fi
>
> (Torvalds Posting to comp.os.minix. Quoted in Moody, 2001, p. 42)

Torvalds's call to the community ("I'd like to know what features most people want") showed that he, like most other members of this newsgroup community, took for granted a hacker culture of cooperation and knowledge sharing. This culture of cooperation and knowledge sharing is exercised throughout the development process of Linux, with hackers from all over the world sharing their skills and knowledge to make Linux function at the highest level possible. It was his belief in knowledge sharing and freedom of software that led to Torvalds' decision to release Linux under GNU/GPL, a decision that ensured the incorporation of future modifications into the main development branch (Moody, 2001). Torvalds felt it a natural decision for him to release Linux system under the GNU/GPL as it had been a common practice within the community he wanted to be part of. This community is "a reasonably strong academic and open" community within the USENET community, and, in fact, in the general UNIX world (FM, 1998).

The Linux kernel was believed to fit Stallman's GNU/GPL operating system like a hand to a glove (Moody, 2001). The development process and the release of the Linux system under GNU/GPL has proven crucial to the later proliferation of the Linux system, and more significantly, it signaled the birth of a new software development model, the model of open source. As noted below, Eric Raymond is closely associated with this open source development model.

Eric Raymond was involved in hacker culture as early as 1976 (Eric Raymond, personal communication, January 31, 2006). Raymond was fascinated by the success of GNU/GPL and published an essay entitled "The Cathedral and the Bazaar" to anatomize "a successful open-source project, fetchmail, that was run as a deliberate test of the surprising theories about software engineering suggested by the history of Linux" (Raymond, 1997). The Cathedral and the Bazaar is commonly regarded as the de facto manifesto of open source movement. In this essay, Raymond postulated a new model of software development that he called the "Bazaar model" to distinguish it from the traditional "Cathedral model" of software development. In the cathedral model, the software development process is centralized in a system where there is relatively strong control over who can submit patches to the code and how such patches are integrated, as well as a rigorous plan for code releases. Under this model, the source code is available with each software release, but code developed between releases is restricted to an exclusive group of developers (Abel, 2005; Johnson, 2005; Pan & Bonk, 2006; Robels, 2004; Weber, 2005; Wheeler, 2004).

In contrast to the cathedral model, the development of software under the Bazaar model is that of a gift culture and distributed development. Raymond (1997) notes that, in effect, the source code of the prototype software is open and freely available to users who are potential co-developers, even though the source code may have limited functionalities. Open source code to users is like a free gift to people. However, this gift culture is not a one way function. Rather, it functions in a give-and-take fashion that creates an obligation for people to give back when a gift is given; of course, the values and beliefs of the giver may also be passed on to the recipients. It thus binds people together. As Weber (2004, p.149) notes, "the artifact being gifted is not just a functional widget but carries with it some of the giver." In this approach, the software is released whenever significant changes are made such that the product evolves in an incremental way, enabling users to modify and debug it. The rationale is that "Given enough eyeballs all bugs are shallow" (Raymond, 1997). This gift culture and distributed development is further manifested in the Open Source Initiative's (2006) introduction to the idea of open source: "When programmers can read, redistribute, and modify the source code for a piece of software, the software evolves. People improve it, people adapt it, people fix bugs."

Such social interaction and knowledge sharing actions are highly linked to the socio-cultural movement. As noted earlier, human development is a function of action within social settings whose values embody the settings' cultural histories (Cole, 1996; Smagorinsky, 2001; Wertsch, 1991). “Every human being has her or his subjectivity and mental life altered through the process of seizing meanings and resources from some socio-cultural environment and using them” (Cole, 1996, p. 117). The evolution of FOSS movement, of the FOSS culture in particular, shows that the beliefs and values of FOSS movement leaders were inherently situated in a social interactional, cultural, institutional, and historical context. As such, the values and beliefs of these leaders impacted the artifacts created as well as the tools and language(s) used (Bonk & Kim, 1998; Cole & Wertsch, 1996; Vygotsky, 1978). As Raymond acknowledges, the “theoretical work on open source came directly from my long-term observation of the hacker culture in action” and “the open-source movement and the hacker culture are no longer distinguishable …” (Eric Raymond, personal communication, January 31, 2006).

Table 1 below is a sequential display of significant tools, artifacts, languages, and social environments in the evolution of FOSS movement. It was developed based on our reading of the literature as well as interviews and discussions with major figures in the field.

A major worry of the FOSS development model relates to the quality of a FOSS project because the project is open to abuse since anyone can download source code, work on it, and redistribute the “finished” product. This quality issue became obvious when people created entries freely in Wikipedia that were biased and inaccurate (The Economist, 2006, p. 74). Equally problematic, the open source product is often left unattended or forgotten once the original developer decides to no longer fund it and offloads this product as open source for the world community (e.g., the e-education course management system (CMS) from Jones Knowledge, Inc.; see Kraan, 2002). For example, on March 28, 2006, there were 116,682 registered open source projects on SourceForge.net, an online hub for open-source software projects. Only six of them, however, were listed as highly active during the previous week, while three out of those six were asking for donations. Clearly, there is continued uncertainty as to whether an open source project will ultimately lead to success or failure (Bezroukov, 1999; Robles, 2004; Room 17, 1999).

Table 1

Evolution of tools, artifacts, languages, and social environments in FOSS

Tool(s)	Artifact(s)	Language(s)	Environment(s)	Representative Figure(s)
Scientific experimental facilities, pen, pencil, journals	Restriction enzyme	Open science, sharing	John Hopkins University, universities, research centers	Hopkins scientists
IBM computers	User-programmer written software	Computer professionals as craftsmen	Research centers, universities	Computer hackers
Computer	Compatible Time-Sharing System (CTSS), Multiplexed Information and Computing Service (Multics)	Computer professionals as craftsmen	Carnegie Mellon, MIT, Stanford, UC Berkeley, Bell Lab	Computer hackers
Computer	ARPANET	Network, network communication	Research institutions under US DoD	Computer hackers
Computer, ARPANET	Unix	Nuclei of communities, fellowship, time sharing	Bell Labs, Carnegie Mellon, MIT, Stanford, UC Berkeley	Thompson, Ritchie, McIlroy
Computer, book, ARPANET	Dragon File The Hacker's Dictionary	Collection of frequently used slang terms	Hacker community at Carnegie Mellon, MIT, Stanford, UC Berkeley	Raphael Finkel, Guy Steele
Computer, ARPANET, Usenet	GNU Manifesto, GNU/GPL	Sharing, free as freedom, not free beer; open source code, free distribution and redistribution	Community of Free Software Foundation, computer software users, hacker community	Richard Stallman
Computer, ARPANET, Usenet,	EMACS	Free distribution and redistribution of source code, software	Community of Free Software Foundation, computer software users, hacker community	Richard Stallman
Computer, Usenet, Internet	Linux	Open source, distributed development, distribution and redistribution of source code	Community of Free and open Software movement, free and open source software users, hacker community	Linus Torvalds
Computer, Usenet, Internet, book	New Hacker's Dictionary	Collection of frequently used slang terms	Community of free and open course software, free and open source software users, hacker community	Eric Raymond
Computer, Usenet, Internet, journal	The cathedral and bazaar	Free and open source code, gift culture/economy. distributed development	Community of free and open software movement, free and open source software users, hacker community	Eric Raymond
Computer, Usenet, Internet	Open Source Initiative	Open source software, business friendly, less ideological	Open Source Initiative community, free and open source software users, hacker community	Bruce Perens, Eric Raymond
Computer, Internet	Moodle Sakai	Social constructivism, community source development	Free and open source community, free and open source software users, hackers, community of core schools	Martin, Dougiamas Indiana University, Michigan University MIT, Stanford

Conclusion

This paper reviewed the evolution of free and open source (FOSS) movement. From a socio-cultural viewpoint, knowledge construction is situated in an interactional, cultural, institutional, and historical context. The evolution of FOSS movement is an interactional process between the socio-cultural environment and the ideas, negotiations, proposals, strategies, and overall leadership of leading figures of the FOSS movement. This socio-cultural environment is first and foremost the institutions involved in it since inception, namely, universities and research centers and the accompanying culture where the leading figures of the FOSS movement studied and worked. It was out of this interaction between the early hackers and the academic culture of open science and knowledge sharing at learning and research institutions that the hacker culture emerged and grew, with a strong belief in the openness and distributed nature of software development.

Not surprisingly, the FOSS movement inherited the values and beliefs of the prevailing hacker culture. The resulting FOSS heritage is manifested in the artifacts created, the tools used for artifact creation, and the language used for communication under the FOSS movement. As the field continues to evolve in the coming decades, socio-cultural perspectives and methodologies might play a significant role in understanding the underpinnings of its evolution as well as the factors that shape communities of knowledge and idea sharing.

As detailed in this paper, the FOSS movement has already played a vital role in the current swell of knowledge sharing in higher education and industry. Perhaps it will evolve into a powerful cultural artifact or tool that can help in efforts to break down walls or pockets of mistrust as well as resistance to communicate with those one has never met. It might also help eliminate the pervasive politics of secrecy within higher education and associated organizational chart hierarchies or chains of command in the name of power since the power is now in anyone's hands with access to the Internet, the willingness to try free and open software, and a creative idea on how to use it. Perhaps it already has.

FOSS is a social, political, economic flattener of the 21st century (Friedman, 2005), but, more importantly, it flattens the world of learning by providing teaching and learning tools that previously were out of reach of the majority of the citizens of this planet. As this becomes increasingly obvious, each of us will likely be involved in it in one way or another. It might be necessary for us to consider the kind of meaningful contributions we can each make to the FOSS movement and intellectually begin to touch other human beings on this planet with our ideas, wisdom, and social supports as well as with our free software developments.

ACKNOWLEDGEMENTS

The authors would like to specifically thank Dr. David Mappin and Dr. Gabriella Coleman for their comments and criticisms during the writing of this paper, and also thank those who were interviewed for this paper.

References

Bennis, W., & Biederman, P. W. (1997) Organizing genius: The secrets of creative collaboration, Perseus Books, Cambridge, MA.

Bereiter, C. & Scardamalia, M. (1996). Adaptation and understanding: A case for new cultures of schooling. In S. Vosniadou, E. De Corte, R. Glaser, & H. Mandl (Eds.), International perspectives on the design of technology-supported learning environments (pp. 149–163). Mahwah, NJ: Lawrence Erlbaum Associates.

Bezroukov, N. (1999). Open source software development as a special type of academic research (Crique of Vulgar Raymond). FirstMonday. Retrieved March 28, 2006, from http://www.firstmonday.dk/issues/issue4_10/bezroukov/index.html

Bonk, C. J., & Cunningham, D. J. (1998). Searching for learner-centered, constructivist, and sociocultural components of collaborative educational learning tools. In C. J. Bonk & K. S. Kim (Eds.), Electronic collaborators: Learner-centered technologies for literacy, apprenticeship, and discourse (pp. 25-50). New Jersey: Erlbaum.

Bonk, C. J., & Kim, K. A. (1998). Extending sociocultural theory to adult learning. In M. C. Smith & T. Pourchot (Ed.), Adult learning and development: Perspectives from educational psychology (pp. 67-88). Lawrence Erlbaum Associates.

Bonk, C. J., Wisher, R. A., & Nigrelli, M. L. (2004). Learning communities, communities of practice: Principles, technologies, and examples. In K. Littleton, D. Miell, & D. Faulkner (Eds.), Learning to collaborate, collaborating to learn (pp. 199-219). NOVA Science.

Brody, W. (1999). From minds to minefields: Negotiating the demilitarised zone between industry and academia, Remarks to the biomedical engineering lectures series, Johns Hopkins University, April 6, 1999.

Cobb, P. (1994). Where is the mind? Constructivist and socio-cultural perspectives on mathematical development. Educational Researcher, 23(7), 13-20.

Cobb, P. (1998). Analyzing the mathematical learning of the classroom community: The case of statistical data analysis. In A. Olivier & K. Newstead (Eds.), Proceedings of the Twenty-Second International Conference for the Psychology of Mathematics Education (Vol. 1, pp. 33 - 48). Stellenbosch, South African: Program Committee of the 22nd PME Conference.

Cole, M. (1996). Cultural psychology: A once and future discipline. Cambridge: Harvard University Press.

Cole, M., & Wertsch, J. V. (1996). Beyond the individual-social antimony in discussion of Piaget and Vygotsky. Human Development, 39, pp 250-256.

Crane, D. (1972). Invisible colleges: Diffusion of knowledge in scientific communities. Chicago, IL: University of Chicago Press.

David, S. (2003). On the uncertainty principle and social constructivism: The case of free and open source software. Unpublished essay. Retrieved February 19, 2006, from http://www.shaydavid.info/indexpapers.html (free software and open source papers)

Dewey, J. (1938). Experience and education. New York: Macmillan.

Dougiamas, M. (1998). A journey into constructivism. Retrieved February 12, 2006, from http://dougiamas.com/writing/constructivism.html

Driscoll, M. P. (2000). Psychology of learning for instruction (2nd ed). Needham Heights, MA: Allyn & Bacon.

The Economist. (2006). Open, but not as usual. The Economist, 378(8469), 73-75.

Feldman, M. P., & Desrochers, P. (2004). Truth for its own sake: Academic culture and technology transfer at Johns Hopkins University. Minerva 42(2), 105 - 126.

Fisher, G., Giaccardo, E., Eden, H., Sugimoto, M., & Ye, Y. (2005). Beyond binary choices: Integrating individual and social creativity. International Journal of Human-Computer Studies, Special issue on creativity, 63(4), 482-512.

Fluid Signal Group. (1992). Linus vs. Tanenbaum. Retrieved February 23, 2006, from http://people.fluidsignal.com/~luferbu/misc/Linus_vs_Tanenbaum.html

FM (1998). FM Interviews with Linus Torvalds: What motivates free software developers? First Monday, Retrieved March 9, 2006, from http://www.firstmonday.dk/issues/issue3_3/torvalds/

Free Software Foundation. (2006). The free software definition. Retrieved February 22, 2006, from http://www.fsf.org/licensing/essays/free-sw.html

Friedman, T.(2005). The world is flat: A brief history of the 21st century. New York: Farrar, Straus and Giroux.

Gilman, D. C. (1998). University problems in the United States. New York: Century.

GNU Bulletin. (1987). What is Free Software Foundation? GNU Bulletin 1(3). Retrieved February 22, 2006, from http://www.gnu.org/bulletins/bull3.html#SEC1.

Greeno, J. G. (1997). On claims that answer the wrong questions. Educational Researcher. 26(1), 5-17.

Hannemyr, G. (1999). Technology and pleasure: Hacking considered constructive. FirstMonday, 4(2). Retrieved February 27, 2006, from http://www.firstmonday.org/issues/issue4_2/gisle/index.html

Haythornthwaite, C., & Wellman, B. (1998). Work, friendship, and media use for information exchange in a networked organization. Journal of the American Society for Information Science 49(12), 1101-1114.

Hilton, J. L. (2005). In praise of sharing. EDUCAUSE Review, 40(3), 72-73. Also available at: http://www.educause.edu/apps/er/erm05/erm053.asp

John-Steiner, V. (2000) Creative collaboration. Oxford University Press, Oxford, UK.

Jonassen, D. H. (1994). Thinking technology: Toward a constructivist design model. Educational Technology, 34(4), 34-37.

Jonassen, D. (1999). Designing construictivist learning environment. In C, M. Reigeluth (Ed.), Instructional theories and models, 2nd. Ed. Mahwah, NJ: Lawrence Erlbaum, pp. 215-240.

Kearsley, G. (2004). Social development theory. Retrieved February 12, 2006, from http://tip.psychology.org/vygotsky.html

Kelty, C. (2001). Free software/Free science. First Monday. 6 (12). http://www.firstmonday.org/issues/issue6_12/kelty/

Kraane, W. (2002, August 29). Jones e-education course management platform open sourced. Retrieved August 5, 2006, from http://www.cetis.ac.uk/content/20020830142138

Krauthammer, C. (2006). Don't believe the hype. U.S. is still no. 1. Time (Candian Ed.), 167(7), 25.

Lave, J., & Wenger, E. (1991). Situated learning: Legitimate peripheral participation. Cambridge, UK: Cambridge University Press.

Lee, C. D., & Smagorinsky, P. (2000). Introduction: Constructing meaning through collaborative inquiry. In C. D. Lee and P. Smagorinsky (Eds.), Vygotskian Perspectives on literacy research (pp. 1-15). NY: Cambridge University Press

Levy, S. (1984). Hackers: Heroes of the computer revolution. Garden City, NY: Anchor Press/Doubleday.

Mahoney, M. J. (2004). Constructivism. Retrieved February 12, 2006, from http://www.constructivism123.com

Menand, L. (2001). The metaphysical club. New York: Farrar, Straus and Giroux.

Moody, G. (2001). Rebel code: Inside Linux and open source revolution. New York: Perseus Publishing.

Open Source Initiative. (2006). The basic idea behind open source. Retrieved March 6, 2006, from http://www.opensource.org/

Pan, G. & Bonk, C. (2007). The emergence of open-source software in China. International Review of Research in Open and Distance Learning, 8(1). Retrieved April 6, 2007, from http://www.irrodl.org/index.php/irrodl/article/view/331/777

Pea, R. D. (1996). Practices of distributed intelligence and designs for education. In G. Salomon (Ed.), Distributed cognitions: Psychological and educational considerations (pp. 47-87). New York: Cambridge University Press.

Pfaffenberger, B. (1996). If I want it, it's OK': Usenet and the (outer) limits of free speech. Information Society 12, 365-388.

Pfaffenberger, B. (1998). Mining communities, châiines opératoire, and sociotechnical systems. In A. B Knapp, V. C. Piggot, & E. W. Herbert (Eds.), Social approaches to an industrial past: The anthropology and archaeology of mining (pp.291-300). London: Routledge and Kegan Paul.

Pfaffenberger, B. (2000). The rhetoric of dread: Fear, uncertainty, and doubt (FUD) in information technology marketing. Knowledge, Technology, and Policy 13(3): 15

Pinch, T., & Bijker, W. (1987). The social construction of facts and artifacts: Or how the sociology of science and the sociology of technology might benefit each other. In W. Bijker, T. Hughes, & T. Pinch (Eds.), The social construction of technological systems: New directions in the sociology and history of technology (pp. 17-50). Cambridge, MA: MIT Press.

Raymond, E. S. (1992). The new hacker's dictionary. In E. S. Raymond (Ed.), The new hacker's dictionary

Raymond, E. S. (1992). The art of UNIX programming. Boston, MA: Addison-Wesley.

Raymond, E. S. (1997). The cathedral and the Bazaar. Retrieved March 6, 2006, from http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/index.html#catbmain

Rhodes, F. H. T. (1999). The university at the millennium. Retrieved February 21, 2006, from http://www.glion.org/

Ritchie, D. M. (1984). The Evolution of the Unix Time-sharing System. AT&T Bell Laboratories Technical Journal 63(6), part 2, 1577-1593. Retrieved March 1, 2006, from http://cm.bell-labs.com/cm/cs/who/dmr/hist.pdf

Robels, G. (2004). A software engineering approach to Libre software. Retrieved March 29, 2006, from http://www.opensourcejahrbuch.de/2004/pdfs/III-3-Robles.pdf

Robert K., & Merton, R. M. (1979). The sociology of science: An episodic memoir. Carbondale: Southern Illinois University Press.

Rogoff, B. (1990). Apprenticeship in thinking: cognitive development in social context. New York, NY: Oxford University Press.

Room 17, Enterprises (1999). Bazaar source. Retrieved December 28, 2005, at http://www.room17.com/ramblings/bazaar_src.shtml.

Russel, S. (1986). The social construction of artifacts: A response to Pinch and Bijker. Social Studies of Science, 16(2). 331-346.

Salomon (Ed.), Distributed cognitions: Psychological and educational considerations (pp. 47–87). New York: Cambridge University Press.

Scardamalia, M., & Bereiter, C. (1996). Adaptation and understanding: A case for new cultures of schooling. In S. Vosniadou, E. De Corte, R. Glaser, & H. Mandl (Eds.), International perspectives on the design of technology-supported learning environments (pp. 149–163). Mahwah, NJ: Lawrence Erlbaum Associates.

Schrage, M. (1990). Shared minds: The technologies of collaboration. New York: Random House.

Smagorinsky, P. (2001). If meaning is constructed, what is it made from? Toward a cultural theory of reading. Review of Educational Research, 71, 133-169.

Stallman, R. (1983). Initial announcement. Retrieved March 2, 2006, from http://www.gnu.org/gnu/initial-announcement.html

Stallman, R. (1985). The GNU project. Retrieved March 3, 2006, from http://www.gnu.org/gnu/thegnuproject.html

Stallman, R. (2006). Serious bio. Retrieved February 23, 2006, from http://www.stallman.org/#serious

Stoan, S. K. (1991). Research and information retrieval among academic researchers: Implication for library instruction. Library Trends 39(3), 238-257.

Tanenbaum, A. (1992). Linus vs. Tanenbaum. Retrieved February 26, 2006, from http://people.fluidsignal.com/~luferbu/misc/Linus_vs_Tanenbaum.html

Tanenbaum, A. (2004). Some notes on the “Who wrote Linux” Kerfuffle, Release 1.5. Retrieved February 25, 2006, from http://www.cs.vu.nl/~ast/brown/

Teel, G. L. Jr. (1992). Confessions of a happy hacker. In E. S. Raymon (Ed.), The new hacker’s dictionary (pp. ix – xv). Cambridge, MA: The MIT Press.

Tharp, R. G. & Gillimore, R. (1988). Rousing minds to life, teaching, learning, and schooling in social context. New York: Cambridge University Press.

Torvalds, L. (1992).). Linus vs. Tanenbaum. Retrieved February 26, 2006, from http://people.fluidsignal.com/~luferbu/misc/Linus_vs_Tanenbaum.html

Vosniadou, S. (1996). Towards a revised cognitive psychology for new advances in learning and instruction. Learning and Instruction 6, 95-109.

Vygotsky, L. (1978). Mind in society: The development of higher psychological process (M. Cole, V. John-Steiner, and E. Souberman [Eds.]). Cambridge, MA: Harvard University Press.

Wark, M. (2004). A hacker manifesto. Cambridge, MA: Harvard University Press (p. 15).

Weber, S. (2004). The success of open source. Cambridge, MA: Harvard University Press.

Wertsch, J. V. (1991). Voices of the mind: A sociocultural approach to mediated action. Cambridge, MA: Harvard University Press.

About the Authors

Guohua Pan

Guohua Pan is an instructional designer with Instructional Media and Design, Grant MacEwan College. Prior to his employment with MacEwan College, Guohua was an instructional designer at Lakeland College in Vermilion, Alberta, for two-and-a-half years. While completing his PhD studies at the University of Alberta, Guohua was a teaching assistant and a research assistant. Originally from Chengdu, China, Guohua taught at a college there before coming to Canada in 1995 to pursue graduate studies. He earned his MEd from the University of Lethbridge and his PhD from the University of Alberta. Guohua is interested in learning theories, technology-based instructional environment, multimedia development, and open-source learning management systems.

Email: pang@macewan.ca

Curtis Bonk

Curtis Bonk is Professor of Instructional Systems Technology in the School of Education at Indiana University and adjunct in the School of Informatics. Dr. Bonk is also a Senior Research Fellow with the DOD's Advanced Distributed Learning Lab. He has received the CyberStar Award from the Indiana Information Technology Association, Most Outstanding Achievement Award from the U.S. Distance Learning Association, and Most Innovative Teaching in a Distance Education Program from the State of Indiana. Dr. Bonk is in high demand as a conference keynote speaker and workshop presenter. He is President of CourseShare and SurveyShare (see http://php.indiana.edu/~cjbonk/;

email: cjbonk@indiana.edu).

Editor's Note: Feedback "makes the communication perfect." How perfect depends on many factors. Immediate knowledge-of-results is better than delayed confirmation. This research goes into three types of feedback and the student's locus of control. It determines that there are significant differences, and that individual learning styles play a role.

A Study on Different Types of Feedback in a Language Multimedia Courseware

Yeap Lay Hwa, Fong Soon Fook, Hanafi Atan, Omar Majid, Wong Su Luan

Malaysia

Abstract

This article reports on a study that examined the effects of different types of feedback in a language multimedia courseware. Three versions of the language multimedia courseware entitled *Subject-Verb Agreement* which incorporated three types of feedback, namely, Knowledge of Results (KOR), Knowledge of Correct Response (KCR) and Elaborated Feedback (EF), were developed. The moderator variable for this study was the students' locus of control, which comprised two levels, namely, the internal and external locus of control for a *Subject-Verb Agreement* test. A total of 134 Semester-Two students enrolled in the Certificate in Mechanical Engineering was randomly divided into three groups with each group being assigned to one of the treatments. A 3 x 2 quasi-experimental factorial design was used in this study and an analysis of variance (ANOVA) was performed to examine the effects of the types of feedback and the students' locus of control on their performance in the *Subject-Verb Agreement* test. This study concluded that the EF, as compared to the KCR and KOR feedback, was the most useful to students with different loci of control in learning *Subject-Verb Agreement*. This study also revealed that students with the internal locus of control achieve better scores than students with the external locus of control. To the "external" students, EF was the most effective type of feedback in learning *Subject-Verb Agreement*. Thus, due consideration has to be accorded to learner characteristics while selecting the type of feedback in the design and development of instructional materials.

Keywords: Feedbacks, multimedia courseware, instructional design, locus of control, computer-based instruction, language courseware, student performance, learner characteristics

Introduction

Feedback is an important input for students in language learning. Immediate and informative feedback will keep students continuously aware of the result of their use of language. According to Vigil and Ollers (1976), feedback determines the degree of internalisation of grammatical rules. Findings from several studies (Tomasello and Herron, 1989; Lightbrown and Spada 1990; White, 1991; Caroll and Swain, 1993; McDonald et al., 2005; Jaehning & Miller, 2007) show that the provision of feedback immediately following an error is important to inform the learners of the discrepancies between faulty and correct forms in the target language. In other words, the information provided in the feedback guides the learners to abandon the wrong form and acquire the correct ones in their language learning.

In the current situation in Malaysia, besides the possible lack of emphasis on providing feedback in the communicative methodology used, the high student-teacher ratio in the classroom may deter teachers from giving enough and quality feedback needed by the students. Nor Hasimah (1999) found that due to this high ratio, teachers have been unable to provide individual attention to the students. Providing feedback and comments on students' work was also a burden to the teachers. As a result, students did not receive enough attention and immediate feedback on their work from the teachers. According to Ross and Morrison (1991), a student-teacher ratio of 25:1 or higher, limits the frequency and quality of the feedback the student will receive. Hence, using the computer in the instructional process through the use of multimedia courseware may have great potential to improve the quality of feedback provided to a learner as this allows the 1:1 ratio of interaction between the computer and the learner. Studies by Waldrop et al., (1986), Clarina et al., (1991), Fong (1996), Myint (1995), Nor Hayati (2002), Rosa and Leow (2004) and Chong (2005) found that feedback provided through computer-based instruction enhances performance in biology, mathematics, and behavioural management courses as well as in text comprehension.

According to Dempsey et al. (1991), there are three common types of feedback in computer-based instruction. These are:

1. **Knowledge of Results** (KOR) – this feedback informs the learner of a correct or incorrect response.
2. **Knowledge of Correct Response** (KCR) – this feedback informs the learner what the correct response to a question should be.
3. **Elaborated Feedback (EF)** – this feedback provides an explanation as to why the learner's response is correct or incorrect.

Several studies have shown that the KCR feedback is superior to the KOR feedback while the latter feedback is better than the absence of feedback (Noonan 1984, Fong 1996, Nor Hayati, 2002). However, evidence also suggests that the EF does not show any significant improvement over the KOR or KCR feedback (Merrill 1987, Spock 1987, Myint, 1995). These findings replicated the results in Schimmel's (1983, 1986) studies. Schimmel (1983, 1986) concluded that the hierarchy of feedback types has not been well established. Hence, this study was undertaken to investigate the relative effectiveness of the KCR and KOR feedback and the EF in the learning of *Subject-Verb Agreement*, one of the topics in the English language grammar course.

Besides the inconclusive evidence on the hierarchy of feedback types, Clarina et. al. (1991) pointed out that there is still a lack of understanding on the situations in which different types of feedback tend to operate most effectively. They attributed this lacking to the failure to account adequately for the influences of factors such as learner characteristics. Several studies have shown that learner characteristics, such as the anxiety level (Nor Hayati, 2002; Chong, 2005) and the cognitive style (Myint, 1995; Fong, 1996; Fong and Ng, 2002), do influence the effectiveness of feedback. Apart from these two factors, the locus of control is another important learner characteristic.

The locus of control was first conceptualised by Rotter (1966) in his theory of social learning. It relates to the aspect of personality which is characterised by a sense of control over reward and reinforcement. There are two types of locus of control namely the internal locus of control and the external locus of control. Student with the internal locus of control exercise control over their life by the choices they make. The external locus of control in the other hand are characterised by beliefs that their destiny is controlled by powerful others, forces outside their control, luck and other unpredictable forces. To add to Rotter's definition of internality and externality, Jonassen and Grabowkli (1993) describe the internals as individuals who tend to attribute causes of success and failure to themselves. They believe that their success is the results of their effort, ability, or competence, and their failure is resulted from their lack of those. Conversely, the externals tend to attribute their successes and failures to external forces that control individual performance such as ability or likelihood to perform acceptably. Rotter (1996) believes that internal-external control is a prominent determinant of an individual alertness to information, which is potentially helpful in guiding the person's future behaviour. Thus he hypothesised that internals would engage themselves more in achievement related activities than externals that tend to feel that they have little control over their rewards and punishment.

The study of the locus of control and types of feedback in determining learning has not been extensively studied. The few studied reported included those of Nishikawa (1988) and Lonky & Reihman (1980). Nishikawa (1988) investigated the effects of locus of control and varied feedback strategies on learner performance during computer-assisted instruction. They found that learners who attribute their successes and failures to internal events (internals) performed significantly better than those who attribute successes and failures to external events (externals) on the delayed feedback on the test for recall. Nishikawa (1988) however did not find any significant differences or interactions between feedback and locus of control. Lonky & Reihman (1980) looked at the effect of individual differences in locus of control school and positive verbal feedback towards student's intrinsic motivation. For individuals lower in locus of control, verbal praise appeared to decrease motivation. For individuals with an external locus of control orientation, it appears that verbal praise given in support of individual performance on intrinsically motivated tasks may actually reduce intrinsic motivation when that praise is no longer forthcoming.

In this study, it was the contention that the locus of control may have significant effects on language learning, especially on the learning of grammar. Several studies (Kasin and Reber 1979, Bar-Tal et al. 1980, Maqsud 1983) have shown the significant effects of the locus of control on language learning. In Kasin and Reber's (1979) study, students who tended to be more internal were able to extract more invariances and thus learned more about the underlying grammatical structure compared to those who tended to be external. Bar-Tal et. al. (1980) found that students who tended to be internal had higher academic achievements in mathematics, English, reading comprehension and Bible studies. Similarly, in an investigation on the effects of the locus of control on academic achievement in mathematics (Chong, 2005) and English, Maqsud (1983) found the levels of academic achievement in the two subjects for students who tended to be internal to be significantly higher than those for the students who tends to be external.

Given the relatively little research on the student's locus of control and different types of feedback types, it was the objective of this study to investigate the effectiveness of three types of feedback, namely, the KCR and KOR feedback and the EF in relation to the students' locus of control in the learning of *Subject-Verb Agreement*. In undertaking this study, the following research questions were asked:

- Do students who received EF feedback, perform significantly better than students who received KCR feedback, who in turn, perform significantly better than students who received KOR feedback in a *Subject-Verb Agreement Test*?
- Across the treatment group, do internals perform significantly better than externals in a *Subject-Verb Agreement Test*?
- Do externals, who received EF feedback, perform significantly better than externals who received KCR feedback, who in turn, perform significantly better than externals who received KOR feedbacks in a *Subject-Verb Agreement Test?*

Methodology

Subjects and the Research Design

The study involved polytechnic students in four intact Semester Two Certificate classes in Mechanical Engineering. The subjects (a total of 134) were Form Five school leavers, aged between 19-20 years old, who had undergone the same English language curriculum for primary and secondary schools in Malaysia. They had also completed the English for Technical Purposes One course, which is compulsory for all technical students in Malaysian polytechnics.

This 3 x 2 factorial quasi-experimental study deployed a pre-test-post-test control group design. The independent variable was feedback, which comprised three levels – Knowledge of Results (KOR), Knowledge of Correct Response (KCR) and Elaborated Feedback (EF). The second factor was the locus of control, a moderator variable, which comprised two levels – the internal and external locus of control. The dependent variable was the students' performance in a *Subject-Verb Agreement* test (measured by the gain score). The gain score was the difference between the post-test and pre-test scores.

Materials and Procedures

Three versions of a "Subject-verb Agreement" courseware, which were identical except for the types of feedback embedded in the practices were developed and used in this study. Figure 1 shows the three different types of feedback incorporated into the courseware. An equal number of each version of the courseware was installed in the computer laboratory. In addition, three sets of instruments – the pre-test paper, the post-test paper and the Intellectual Academic Responsibility (IAR) questionnaire (Crandall, Katskosky, & Crandall, 1965) were used in data collection. This questionnaire contained 34 questions and was administered to determine the students' locus of control. In this study, students who scored above the mean score were identified as students with an internal locus of control while those who scored below the mean score were categorised as being those with an external locus of control. The pre-test/post-test contained 20 multiple-choice questions, which gauged students' understanding and ability to apply the three subject-verb agreement rules learned from the courseware. The two tests were similar in content except for the sequence of items.

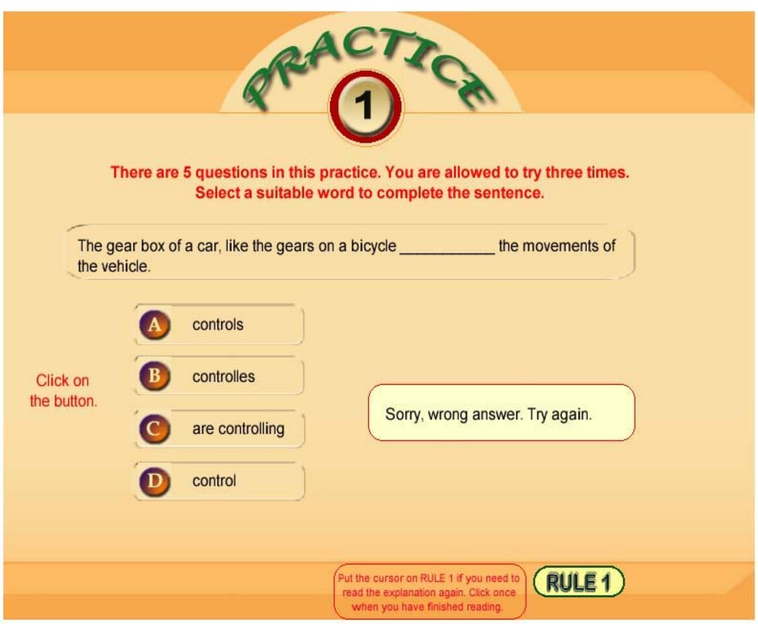

Figure 1. KOR type of feedback

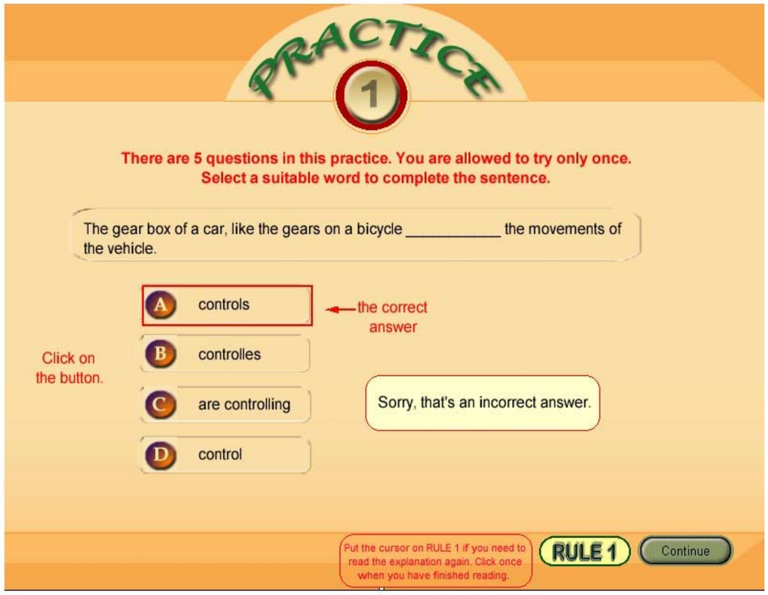

Figure2. KCR type of feedback

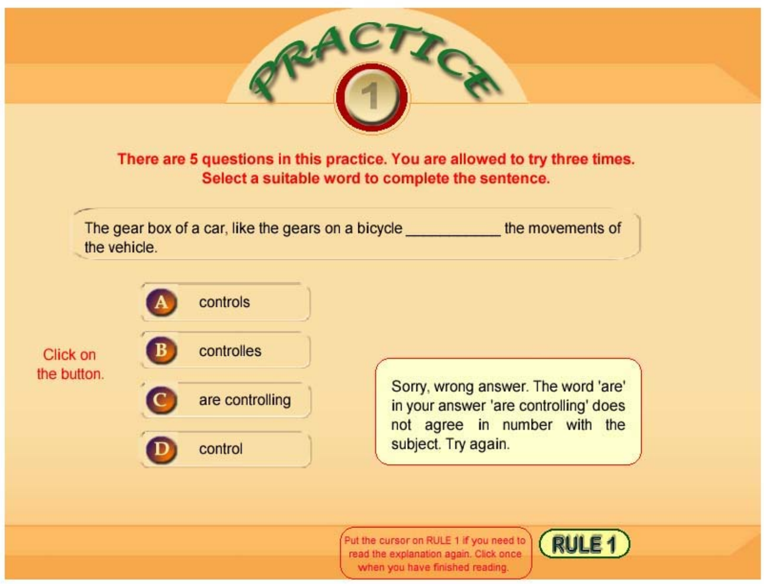

Figure 3. EF type of feedback

The students were randomly assigned to three treatment groups. Upon receiving the treatment condition, the students sat for a pre-test and answered the IAR questionnaire. Two weeks after the pre-test, the students received instruction using the *Subject-verb Agreement* courseware. A pos-test was administered immediately after the treatment. A series of analysis of variance (ANOVA) was performed to examine the effects of the types of feedback and the students' locus of control on their performance in the subject-verb agreement test.

Results and Discussion

Regardless of the type of feedback received and the category of locus of control, all the students in the study were essentially the same in the measure of the dependent variable at the beginning of the study. There was homogeneity in their performance prior to the treatment. The low pre-test mean scores of 31.68 at the beginning of the study indicated non-mastery of the subject-verb agreement grammar area before the treatment. Following the feedback treatment, the mean score was improved by 29.40 points in the post-test, which illustrated the efficacy and the role played by the feedback in enhancing the students' performance.

Performance of Students by Feedback Groups

The first research question examined whether there was a significant difference in the performance of students who received different type of feedback. The results from the analysis of variance and post hoc multiple comparison tests indicated significant difference in the performance of students in the three feedback groups (see Tables 1 and 2). It was found that there was a significant difference in the performance of students who received EF, KCR and KOR feedback. Students who received the EF performed significantly better than students who received the KCR feedback as well as those who received the KOR feedback. Students in the KCR feedback group, in turn, outperformed students in the KOR feedback group significantly.

Table 1

ANOVA of mean gain scores by feedback groups

Source of Variation	Sum of Squares	Df	Mean Square	F	Sig.
Main Effects	7278.181	2	3639.091	14.293	.000
Feedback	7278.181	2	3639.091	14.293	.000
Explained	7278.181	2	3639.091	14.293	.000
Residual	33353.162	131	254.604		
Total	40631.343	133			

Table 2

Post hoc tests: multiple comparisons of mean gain scores by feedback groups

Feedback Group (I)	Feedback Group (J)	Mean Difference (I - J)	Std. Error	Sig.
KOR Feedback	KCR Feedback	-8.67*	3.365	.030
	EF	-18.18*	3.402	.000
KCR Feedback	KOR Feedback	8.67*	3.365	.030
	EF	-9.52*	3.365	.015
EF	KOR Feedback	18.18*	3.402	.000
	KCR Feedback	9.52*	3.365	.015

The mean difference is significant at the .05 level.

The results of this study are supportive of the positive value of feedback to enhance the performance of students. Figure 4 illustrates the hierarchy of feedback types in terms of their effects on students' performance as found in this study.

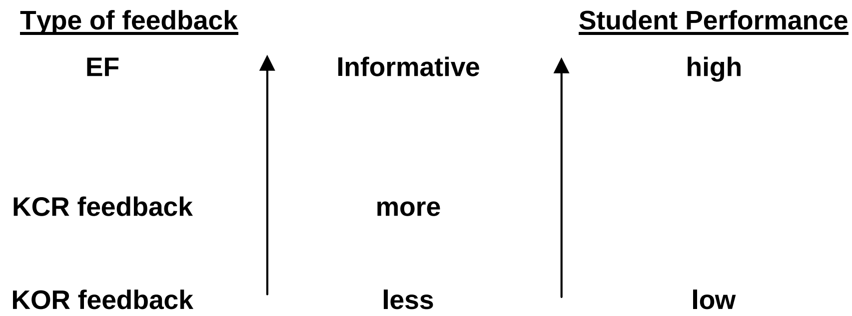

Figure 4: Hierarchy of types of feedback and effects on student performance

The results are consistent with the findings from previous studies by Nor Hayati (2002), Fong (1996), Myint (1995), Waldrop et. al. (1986) and Noonan (1984). The EF, as explained by Fong (1996), is superior to the KCR and KOR feedback because, apart from providing the correct answer, the EF provides an explanation to an answer. For an incorrect response, the EF assists students to arrive at the correct answer by providing corrective information. The KOR and KCR types of feedback, on the contrary, provide only verification information. The KOR feedback informs students on the correctness of their responses while the KCR feedback provides the correct answer in response to a student's incorrect answer. Van der Linden (1993) as well as Caroll and Swain (1993) suggest that feedback with corrective information, as compared to verification feedback, enhances language learning because students are guided to abandon their faulty form and acquire the correct form of the target language. Kulhavy and Stock (1989) provided another explanation to the relative effectiveness of EF over the KCR and KOR feedback. Kulhavy and Stock (1989) found that if students are confident that their answers are correct but later discover that they are incorrect, they would spend more time to study the feedback. It is possible that during this type of response interaction, the EF assists students to correct their errors and this subsequently improves their performance.

Performance of Students by Locus of Control

The study second research questions explore the possible influence of locus of control on student performance. The finding of this study also revealed that students with the internal locus of control achieve better scores than students with the external locus of control (see Table 3). The students with an internal locus of control gained significantly higher scores than the students with an external locus of control in the three feedback groups (see Table 4). This finding is in line with the results of other studies such as those by Fong (2000), Khor (1999), Yeoh (1999), Park and Kim (1998), Ferguson (1988), Maqsud (1983) and Bar-Tal et.al.(1980). According to Rotter (1966), students with internal and external loci of control differ in their achievement due to their different alertness to information. Students with an internal locus of control are more attentive to information and they will engage more in learning activities compared with students with an external locus of control.

Table 3

ANOVA of gain scores by locus of control

Source of Variation	Sum of Squares	df	Mean Square	F	Sig.
Main Effects	1765.792	1	1765.792	5.997	.016
LOCUS OF CONTROL	1765.792	1	1765.792	5.997	.016
Explained	1765.792	1	1765.792	5.997	.016
Residual	38865.551	132	294.436		
Total	40631.343	133			

Table 4

Comparisons of mean and standard deviation by locus of control

Feedback Group	Locus of control	N	Mean	Std. Deviation
KOR Feedback	Internals	23	24.57	14.374
	Externals	21	16.43	9.765
	Total	44	20.68	12.922
KCR Feedback	Internals	29	32.59	20.117
	Externals	17	23.82	15.261
	Total	46	29.35	18.786
EF	Internals	21	42.62	13.381
	Externals	23	35.43	16.714
	Total	44	38.86	15.472
Total	Internals	73	32.95	17.889
	Externals	61	25.66	16.240
	Total	134	29.63	17.479

Student Performance with External Locus of Control by Feedback Groups

The third research question in this study investigated whether the different types of feedback have any effect on performance of students with external locus of control. The ANOVA analysis on the effect of different types of feedback on the performance of student with external locus of control yielded a value of F=9.957 (2,58) which was significant at p=0.00 (see Table 5) indicating a significant difference in the performance of student with external locus of control in the three feedback groups. A further analysis on the mean gain scores of the three groups in the Post Hoc Multiple Comparison analysis (Table 6) shows that of the three types of feedback, the EF feedback was found to benefit the students with an external locus of control the most. These students in the EF group performed significantly better than the students with the external locus control in the KOR as well as the KCR group. It was also found that the KCR feedback group attained scores which were marginally higher than the KOR feedback group. However, there was no significant difference in the performance of these students in the KOR and KCR groups.

Table 5

ANOVA of gain scores of students with an external locus of control in various treatment groups

Source of Variation	Sum of Squares	df	Mean Square	F	Sig.
Main Effects	4044.505	2	2022.252	9.957	.000
FEEDBACK	4044.505	2	2022.252	9.957	.000
Explained	4044.505	2	2022.252	9.957	.000
Residual	11779.266	58	203.091		
Total	15823.770	60			

Table 6

Post hoc tests: multiple comparisons of mean gain scores of students with an external locus of control in various treatment groups

Feedback Group (I)	Feedback Group (J)	Mean Difference (I-J)	Std. Error	Sig.
KOR Feedback	KCR Feedback	-7.39	4.649	.258
	EF	-19.01*	4.301	.000
KCR Feedback	KOR Feedback	7.39	4.649	.258
	EF feedback	-11.61*	4.558	.036
EF	KOR Feedback	19.01*	4.301	.000
	KCR Feedback	11.61*	4.558	.036

* The mean difference is significant at the .05 level.

The result of this study does not provide sufficient evidence to distinguish the effect of KOR and KCR. However, this result provides evidence that the EF feedback is the most effective feedback for the student with external locus of control in learning *Subject-Verb Agreement*. As discussed earlier, the corrective feedback in the EF feedback assists students to self-correct their errors, and therefore subsequently improve their performance. The insignificant difference between the KOR and KCR feedback group could be explained by the slight difference in the information provided in these two types of feedback. In addition to the knowledge on the correctness of their answer, student with external locus of control in the KCR feedback were informed of the correct answer to a question. Knowing the correct answer did not seem to be sufficient for the student with external locus of control to produce scores significantly higher than the external student in the KOR feedback group. More studies are needed to provide better understanding on the influence of the locus of control on the effect of the three type of feedback

Conclusion

Feedback is an essential element in a learning process. It provides learners with confirmation on the accomplishment of a learning purpose. A variation in the information provided in feedback seems to produce a different level of gains in learning. This study has shown that learning gains increase as the feedback provided becomes more informative. Elaborated feedback (EF), which provides verification of an answer and explanation to the correctness of an answer, has been found to be more beneficial to learners compared to the Knowledge of Results (KOR) and Knowledge of Correct Response (KCR) feedback. The KCR feedback that informs learners of the correct answer is found to be superior to the feedback that informs learners of the incorrectness of answer (the KOR feedback). The results of this study also showed the influence of the locus of control on the efficacy of the three types of feedback. On the whole, the EF benefits both the students with internal and external loci of control. Due considerations need to be accorded to learner characteristics while selecting the type of feedback in the design and development of instructional materials.

References

Bar-Tal, D., Kfir, D., Bar-Zohar,Y. & Chen, M.(1980). The relationship between locus of control and academic achievement, anxiety and level of aspiration. *British Journal of Educational Psychology*, 50, 53-60.

Caroll, S.,& Swain, M.(1993). Explicit and implicit negative feedback: an empirical study of the learning of linguistic generalizations. *Studies in Second Language Acquisition*, 15, 357-386.

Chong, P.L., 2005. Effect of feedbacks among students of different anxiety levels in mathematics. M.Ed. Thesis. Universiti Sains Malaysia. Unpublished

Clarina, R.B., Ross, S.M. & Morrison, G.R.(1991). The effects of different feedback strategies using computer-administered multiple-choice questions as instruction. *Educational Technology Research and Development*, 39(2), 5-17.

Dempsey, J.V., Driscoll, M.P.,& Swindell, L.K.(1991). Text-based feedback. In Dempsey, J., and Sales, G.(Ed.). *Interactive Instruction and Feedback*. Englewood Cliffs: Educational Technology Publications.

Ferguson, H.L.(1988). The effect of locus of control on achievement and preference in CAI tutorial systems which vary in terms of system control, learner control with coaching, and learner control with no coaching. *Dissertation Abstracts International*, 50, 4652.

Fong, S.F.(1996). Effects of different levels of feedback using multimedia computer-based learning on the learning gain of students with different cognitive styles. Master Thesis. University Sains Malaysia. Unpublished.

Fong, S.F.(2000). The effect of animation towards teaching and learning meiosis among students various physiological. PhD Thesis. Universiti Sains Malaysia. Unpublished.

Fong, S.F.& Ng, W.K.(2002). *Effects of feedback in a multimedia computer-based learning on the achievement of students with different cognitive styles*. Paper presented at the IOSTE South East Asia Regional Symposium on Science and Technology Education, Copthorne Orchid Hotel, Penang, Malaysia, 30 March to 2 April 2000.

Jaehning, W. & Miller, M.L. (2007). Feedback types in programmed instruction: A systematic review. *The Psychological Record*, 57, 219-232.

Jonassen, D.H., & Grabowski, B.L.(1993). Handbook of individual differences, learning, *and instruction*. New Jersey : Lawrence Erlbaum Associates Publishers.

Kasin, S. M.& Reber, A. S.(1979). Locus of control and the learning of an artificial language. *Journal of Research in Personality*, 13(1),112-118.

Khor, S.H.(1999). The effect of analogical teaching towards students achievement with different locus of control. Master Thesis. Universiti Sains Malaysia. Unpublished.

Kulhavy, R.W., & Stock, W.A. (1989). Feedback in written instruction: the place of response certitude. *Educational Psychology Review*, 1, 279-308.

Lightbrown, P.M. & Spada, N.(1990). Focus-on-form and corrective feedback in communicative language teaching: Effects on second language learning. *Studies in Second Language Acquisition*, 12, 429-448.

Lonky, E & Reihman, J. (1980). Cognitive evaluation theory, locus of control and positive verbal feedback. Paper presented at the Annual Convention of the American Psychological Association, Montreal, Quebec, Canada, September 1-5. (ERIC Document195324).

Mc Donald, J.K. & Yanchar, S.C., Osguthrope, R.T. (2005). Learning from programmed instruction: Examining implication for modern instruction technology. *Educational Technology Research and Development*, 53(2), 84-98.

Maqsud, M. (1983).Relationships of locus of control to self-esteem, academic achievement, and prediction of performance among Nigerian secondary school pupils. *British Journal of Educational Psychology*, 53, 215-221.

Merrill, J. (1987). Levels of questioning and forms of feedback : instructional factors in courseware design. *Journal of Computer-Based Instruction*, 14, 18-22.

Myint, S.K.(1995). An experimental study on interactive effects between field dependent-independent students and varying level of feedback in multimedia presentation. Master Thesis. Universiti Sains Malaysia. Unpublished.

Nishikawa, S. (1988). A comparison of the effects of locus of control with feedback strategies on factual information recall and retention during computer assisted instruction. Proceedings of Selected Research Papers presented at the Annual Meeting of the Association for Educational Communications and Technology (New Orleans, LA, January 14-19). (ERIC Document 295655).

Noonan, V. J.(1984). Feedback procedures in computer-assisted instruction: knowledge of results, knowledge of correct response, process explanations, and second attempts after errors. Dissertation Abstracts International, 45, 131-A.

Nor Hashimah Hashim. (1999). Teacher’s responses to class size at the primary level in Malaysia. *Educators and Education*, 16, 29-37.

Nor Hayati Abdul Rahman (2002). The effect of feedback on computer-based learning towards student’s performance with different mathematic anxiety. Master thesis. Universiti Sains Malaysia. Unpublished.

Park, Y.S.,& Kim, U.(1998). Locus of control, attributional style, and academic achievement: comparative analysis of Korean-Chinese, and Chinese students. *Asian Journal of Social Psychology*, 1(2), 191-208.

Rosa, E.M. & Leow, R.D. (2004). Computerised task-based exposure, explicitness, type of feedback, and Spanish L2 development. *The Modern Language Journal*, 88(2), 192-216.

Ross, S.M., & Morrison, G.R.(1991). Using feedback to adapt instruction for individuals. In Dempsey, J., and Sales, G.(Ed.). *Interactive instruction and feedback*. Englewood Cliffs: Educational Technology Publications.

Rotter, J.B. (1966). Generalized expectancies for internal versus external control of reinforcement. *Psychology Monograph,* 80 (1, Whole No. 609).

Schimmel, B.J. (1983). *A meta-analysis of feedback to learners in computerized and programmed instruction*. Paper presented at the annual meeting of the American Educational Research Association, Montreal. (ERIC Document 233708).

Schimmel, B.J.(1986). Feedback use by low ability students in computer-based education. *Dissertation Abstracts International, 47,* 68A.

Spock, P.(1987). Feedback and confidence of response for a rule-learning task using computer-based instruction. *Dissertation Abstracts International, 48,* 1109.

Tomasello, M. & Herron, C.(1989). Feedback for language transfer errors: The Garden Path technique. *Studies in Second Language Acquisition,* 11, 385-395.

Van der Linden, E.(1993). Does feedback enhance computer-assisted language learning? *Computers Education,* 21(1/2), 61-65.

Vigil, N.A.,& Oller, J.W. (1976). Rule fossilization:a tentative model. *Language Learning,* 26, 281-295.

Waldrop, P.B., Justen, J.E. III, & Adams, T.M. II.(1986). A comparison of three types of feedback in a computer-assisted instruction task. *Educational Technology,* 26(11), 43-45.

White, L.(1991). Adverb placement in second language acquisition: some effects of positive and negative evidence in the classroom. *Second Language Research,* 7, 133-161.

Yeoh, L.S. (1999). The effect of different mode of presentation on students with different locus of control. Master Thesis. Universiti Sains Malaysia. Unpublished.

About the Authors

Dr Hanafi Atan
Deputy Dean (Postgraduate Studies and Research)
School of Distance Education
Universiti Sains Malaysia
11800 Penang, Malaysia
Tel: 604-6533265
Fax: 604-6576000

hanafi.atan@gmail.com, ahanafi@usm.my, itq5hanafi@yahoo.com

Yeap Lay Hwa,
Seberang Perai Polytechnic, 13500 Penang, Malaysia

Fong Soon Fook,
School of Educational Studies, Universiti Sains Malaysia, 11800, Penang, Malaysia

sffong@gmail.com

Omar Majid
School of Distance education, Universiti Sains Malaysia, 11800, Penang, Malaysia

momar@usm.my

Wong Su Luan
Faculty of Education, Universiti Putra Malaysia, Serdang 43400, Selangor, Malaysia

wsuluan@gmail.com