Volume 9, No. 2
June 2011
ISSN: 1545-679X

Information Systems Education Journal

In this issue:

EDSIG
Serving Information Systems Educators

The **Information Systems Education Journal** (ISEDJ) is a double-blind peer-reviewed academic journal published by **EDSIG**, the Education Special Interest Group of AITP, the Association of Information Technology Professionals (Chicago, Illinois). Publishing frequency is quarterly. The first year of publication is 2003.

ISEDJ is published online (http://isedjorg) in connection with ISECON, the Information Systems Education Conference, which is also double-blind peer reviewed. Our sister publication, the Proceedings of ISECON (http://isecon.org) features all papers, panels, workshops, and presentations from the conference.

The journal acceptance review process involves a minimum of three double-blind peer reviews, where both the reviewer is not aware of the identities of the authors and the authors are not aware of the identities of the reviewers. The initial reviews happen before the conference. At that point papers are divided into award papers (top 15%), other journal papers (top 30%), unsettled papers, and non-journal papers. The unsettled papers are subjected to a second round of blind peer review to establish whether they will be accepted to the journal or not. Those papers that are deemed of sufficient quality are accepted for publication in the ISEDJ journal. Currently the target acceptance rate for the journal is about 45%.

Information Systems Education Journal is pleased to be listed in the 1st Edition of Cabell's Directory of Publishing Opportunities in Educational Technology and Library Science, in both the electronic and printed editions. Questions should be addressed to the editor at editor@isedj.org or the publisher at publisher@isedj.org.

2011 AITP Education Special Interest Group (EDSIG) Board of Directors

Information Systems Education Journal

Are Password Management Applications Viable? An Analysis of User Training and Reactions

Mark Ciampa
mark.ciampa@wku.edu
Western Kentucky University
Bowling Green, KY 42101 USA

Abstract

Passwords have the distinction of being the most widely-used form of authentication—and the most vulnerable. With the dramatic increase today in the number of accounts that require passwords, overwhelmed users usually resort to creating weak passwords or reusing the same password for multiple accounts, thus making passwords the weakest link in the chain of security. It has been recognized that instead of solely relying on their memory for passwords, users can take advantage of technology. One such technology is a password management application, which enables a user to create and store multiple passwords in a strongly protected file and then retrieve them as necessary, thus alleviating the need to memorize numerous passwords. However, few users have chosen to take advantage of these applications. Is it because users have rejected them as poor solutions, or because they were unaware of these applications and their potential benefits? Would users be more favorable towards password management applications after they received training about these applications and then used them? What limitations of these applications could be addressed to foster more widespread use? To-date no studies have provided training to users regarding these applications prior to surveying their reactions to determine if indeed these applications are suitable for the average user. This paper describes a study regarding user's training, use, and perceptions of a password management application.

Keywords: information security, passwords, password management applications, KeePass

1. INTRODUCTION

Authentication is the process of providing proof that a user is actually who they say that they are (Pastore & Dulaney, 2006). Authentication systems are based on the use of a physical token (something you have), a physical characteristic (something you are), or secret knowledge (something you know) that can uniquely distinguish a user (Burnett & Kleinman, 2006). The most common type of authentication in use today is a password (Kruger, Steyn, Medlin, & Drevin, 2008), which is based on something that is only known by the user and thus prevents imposters from impersonating the user.

Yet, despite their widespread use, passwords provide a weak degree of protection and undermine the system (Gaw & Felten, 2006). Schneier (2004) says that "systems are only as secure as the weakest password".

The weakness of passwords centers on human memory. Human beings can memorize only seven (plus or minus two) "chunks" of information (Miller, 1956). As more items are added to memory, the number of items that are forgotten increases (Neath, 1998).

Passwords place heavy loads on human memory in two ways. First, a password should be of a sufficient length and complexity that an attacker cannot easily determine it. However, long and complex passwords of this type can be difficult to memorize and can strain the ability to accurately recall them. Most users have difficulty remembering these types of strong passwords (Charoen, Raman, & Olfamn, 2008).

Second, the number of different accounts and passwords that are required today also places a load on a user's memory. Typically users have multiple accounts for different computers at work, school, and home, for various e-mail accounts, for online banking and Internet sites, to name a few, and each account has its own password. Despite research by Gaw and Felten (2006) showing that the majority of 49 undergraduate test subjects had three or fewer passwords, other studies have indicated a much higher number of passwords per user. Research cited by Vu, Proctor, Bhargav-Spantzel, Tai, Cook, and Schultz (2007) indicated that 35% of users had 3-4 passwords, 18% had 5-6 passwords, 6% had 7 to 8, and 23% of users had 9 or more passwords, while other research showed that 28% of a group had over 13 passwords each. Sasse and Brostoff reported that a group of 144 users had an average of 16 passwords (Sasse & Brostoff, 2001), while Brown, Bracken, Bracken, Zolccoli and Douglas (2004) reported a group of college students (n=218) averaged 8.18 passwords each. Choren, Raman and Olfamn noted that because users have multiple accounts requiring multiple passwords, it is "more than slightly impossible" for users to remember each password (2008).

The problem is even exacerbated by security policies in which passwords are set to expire after a period of time, such as every 45 days, and a new one must be created. Some security policies even prevent a previously used password from being recycled and used again, forcing the user to repeatedly memorize multiple new passwords for multiple accounts.

Due to the burdens that passwords place on human memory, users typically take shortcuts to help them recall their passwords. The first shortcut is to use a weak password. These may include a common word used as a password (such as "January"), a short password (such as "ABCDE"), or personal information in a password (such as the name of a child or pet). The second shortcut is to reuse the same password for multiple accounts, making it easier for an attacker who compromises one account to be able to access multiple other accounts. Research by Gaw and Felten (2006) showed that users accumulate more online accounts, as they get older, yet the number of unique passwords does not increase. As users accumulate more online accounts they are simply reusing passwords more frequently.

Schneier summarizes the issue by stating, "The problem is that the average user can't and won't even try to remember complex enough passwords to prevent dictionary attacks. As bad as passwords are, users will go out of the way to make it worse. If you ask them to choose a password, they'll choose a lousy one. If you force them to choose a good one, they'll write it on a Post-it and change it back to the password they changed it from the last month. And they'll choose the same password for multiple applications" (2004).

2. ADDRESSING PASSWORD WEAKNESSES

In order to address the weaknesses associated with passwords, different solutions have been proposed to help users overcome poor password practices. These solutions may be grouped into four broad categories.

Change how passwords are created

The first category is comprised of solutions to change how textual passwords are created. Bunnell, Podd, Henderson, Napier, and Kennedy-Moffat (1997) and Yan, Blackwell, Anderson and Grant (2004) have explored rates for different methods to generate and associate text-based passwords. Other researchers have proposed splitting a textual password into two parts: one part is written down on a paper while the second part is encoded in a mnemonic sentence (Topkara, Atallah, & Topkara, 2007).

Substitute graphical passwords

The second category of solutions is substituting textual passwords with graphical passwords. There are three advantages to graphical passwords. Graphical passwords are based on the premise that figures or images are easier for users to recall than text. Also, graphical passwords utilizing images are more difficult for an attacker to circumvent. Finally, graphical passwords may also help address a fundamental weakness of user-created textual passwords, namely that users select passwords that represent themselves and even sum up the very essence of their being in a single word (Gaw & Felten, 2006). Attackers frequently attempt to guess a textual password by using personal information about the user, which could be more difficult with a graphical password.

Proposals for graphical passwords include clicking on specific points of a scene in a particular sequence within an image (Wiedenbeck, Waters, Birget, Brodskiy, & Memon, 2005) or identifying a series of random art images (Dhamija & Perrig, 2000). Another proposal requires the user to identify specific faces (Tari, Ozok, & Holden, 2006). Users are provided a random set of photographs of different faces, typically three to seven, and are taken through a "familiarization process" that is intended to imprint the faces in the user's mind. A user must select his assigned faces from three to five different groups, with each group containing nine faces, before being authenticated. Even using personalized hand-drawn "doodles" for authentication has been proposed by Goldberg, Hagman and Sazawal (2002), Govindarajulu and Madhvanath (2007), and others.

Use alternative authentication methods

The third category of solutions for overcoming weaknesses associated with passwords is to use alternative methods of authentication. One common method is standard biometrics, which uses a person's unique characteristics for authentication and usually involves fingerprints, faces, hands, irises, or retinas. However, because standard biometrics requires a biometric hardware scanning device to be installed at each computer where authentication is required and because of the large numbers of false negatives of rejecting authorized users, standard biometrics have not been widely implemented.

To address the weaknesses in standard biometrics, a new type of biometrics known as behavioral biometrics is being developed. Instead of examining a specific body characteristic, behavioral biometrics authenticates by normal actions that the user performs. Two types of behavioral biometrics are keystroke dynamics and voice recognition. Keystroke dynamics attempt to recognize a user's unique typing rhythm by using two unique typing variables: dwell time, which is the time it takes for a key to be pressed and then released, and flight time, or the time between keystrokes. Voice recognition uses the unique characteristics of a person's voice for authentication. Voice recognition is not to be confused with speech recognition, which accepts spoken words for input as if they had been typed on the keyboard.

Make use of technology

The final category for addressing password weaknesses is to use technology. Modern Web browsers such as Firefox and Microsoft's Internet Explorer (IE) contain a function to allow a user to save a password that has been entered while using the browser (called an AutoComplete Password in IE) or through a separate dialog box that "pops up" over the browser (called an HTTP Authentication Password in IE). AutoComplete passwords are stored in the Microsoft Windows registry and are encrypted with a key created from the Web site address while HTTP Authentication Passwords are saved in the credentials file of Windows, together with other network login passwords.

Another solution in this category for addressing password weaknesses is password management applications. Called the "digital equivalent" to a written Post-It note by Gaw and Felten (2006), these programs let a user create and store multiple strong passwords in a single user file that is protected by one strong master password. Users can retrieve individual passwords as needed by opening the user file, thus freeing the user from the need to memorize multiple passwords.

Yet most password management applications are more than a password-protected list of passwords and include many additional features (Reichl, 2010). One additional feature of many password management applications is the ability to create strong random passwords through random seeding based on a user's mouse movement and random keyboard input. This enables these password managers to meet the criteria for effective password management as set forth by Kruger, Steyn, Medlin, and Drevin (2008) of both creating secure passwords and protecting the confidentiality of them. Examples of password management applications include KeePass, Password Safe, RoboForm, Access Manager, and others.

3. OVERVIEW OF THE STUDY

Despite the advantages of password management applications, relatively few users have chosen to use them. In a study by Gaw and Felten (2006), 49 users were told to bring "anything you use to help you remember your passwords (password lists, daily planners or notebooks, digital assistants, copies of bank or travel statements, copies of items in your Internet browser cache, etc.)". Only six participants brought aids, none of which was a password management application. Gaw and Felten (2006) concluded that these applications "interrupt the user's behavior" and were "relatively unpopular". However, they also stated that "technology solutions could help".

This study sought to determine the reason why password management applications are used so infrequently. Is it because users are familiar with them yet have rejected them as poor solutions, or is it because they are unaware of these applications and their benefits? If the latter is the case, would users be more inclined to use these applications once they received training and actually used them? If not, what are the limitations of these applications that could be addressed to create more widespread use?

Participants

The ideal study population is all users who have passwords. Because that obviously is not possible, a sample was selected that did not cause any serious threats to the external validity. A relatively large sample of undergraduate student participants is representative of that population. Kruger et al. notes that modern universities, with their core business focused on teaching and research, are in fact managed and operated along the same line as any business. In addition, there are a large number of confidential and privacy issues associated with student users that can directly be linked to passwords and the management of passwords (Kruger, Steyn, Medlin, & Drevin, 2008).

This study can also serve to prepare the students to be more security conscious when they enter the workforce full-time. Werner (2005) said that as employees, new college graduates will have access to critical data to perform their jobs yet they could be the weakest link in a secure computer system primarily because of inadequate education, negligence, and inexperience. The instruction and training as part of this study can not only meet the current demands of securing systems but also better prepare students for future employment in their respective fields.

Instruction and training

Because relatively few users have chosen to use password management applications, it was necessary in this study to first provide instruction and training to the student participants. Students needed an entire instructional "process" in order to understand password security and to have hands-on experience using a password management application. Only then would students be in a position to provide a reasoned response regarding their experiences and perceptions.

All student participants were required to complete a four-step process regarding password security and password management applications. First, the students read a 37-page chapter of material that included a running vignette, examples, figures, summary, and list of key terms regarding personal security and password management. Second, the students watched a 45-minute video of the chapter material. Third, the students took a 20-question assessment to determine their level of understanding of the material. Only after these steps were completed to provide the necessary foundation, the students then followed instructions how to download, install, and use a specific password management application. Once this activity was completed the students related on a survey their experiences, how likely they were to use the application, and the reasons for their decisions.

The depth of the training was considered to be an important element in this study. First, the broader background of password security was introduced to students, so they could have a context in which to understand password management applications. Second, by assessing student learning it served to validate student learning of the objectives. Third, by using different pedagogical approaches--auditory (lecture video), visual (textbook), and kinesthetic (hands-on use)—it met the needs of the different types of learners.

4. PILOT STUDY

A pilot study was first conducted prior to the actual study. A group of 21 participants read the material and viewed the lecture video. Upon completion of the video they were given a 20-question assessment (N=20, M=19, SD=0.92). Following the assessment the participants downloaded KeePass, an open source password management application, and installed it. They then were instructed to use the application to record a personal password and retrieve it for use.

The participants next were asked their opinions regarding the application in four key areas: 1) Is this an application that would help users create and use strong passwords?; 2) What are the strengths of these password programs?; 3) What are the weaknesses?; and 4) Would you use KeePass? Participant responses were open-ended narratives.

Of the 21 participants two indicated that they would use KeePass. Two additional participants indicated that they "might" or would "strongly consider" using the application. Ten participants stated that they would not use KeePass or a similar application. Their comments generally focused on three reasons: 1) no personal need for a password management application; 2) password management applications were inconvenient; and 3) the risk of an attacker stealing their master password and then having accessing to all stored passwords. Of the remaining seven participants, four provided comments but did not indicate if they would use the application personally. Three participants gave no comments.

5. STUDY

The study was conducted at a regional university and a community college. Student participants were from one of four sections of computer courses.

Of the 101 students who participated, 68 (67%) attended the university, of which 54 were male and 14 were female, while 33 (33%) students attended the community college (10 male and 23 female). A total of 61 students (60%) were employed (54 university students and 7 community college students).

All participants were required to complete a four-step process: 1) read a chapter of material regarding personal security and password management, 2) watch a lecture video, 3) take an assessment, and 4) download, install, and use the KeePass password management application. Once this activity was completed the students completed a survey regarding their experiences, how likely they were to use the application, and the reasons for their decisions.

6. RESULTS

Upon completion of reading the chapter of material regarding personal security and password management followed by viewing the video, all students were given a 20-question assessment regarding the material (N=101, M=16.67, SD=2.84). The purpose of the assessment was to both provide evidence that the students had actively engaged in reading and viewing the material and also to provide a message to the students about what they should be learning (Knight, 1995).

In order to examine student attitudes towards a password management application, four sets of survey questions were provided. These questions queried the students regarding the ease of use, benefits, and usefulness of the application.

Participant Attitudes Towards KeePass

The first set of questions was measured using a 5-point Likert scale, ranging from "1-Strongly Agree" to "5-Strongly Disagree". The analysis of the results investigated the median, mean, and standard deviation of the attitude of the students towards their experiences using the KeePass password management program. These statistical results are listed in Table 1 and the median values are illustrated in Figure 1, both of which are found in the Appendix.

The results from Table 1 indicate that participants found KeePass easy to use (Question 1, Mdn=1, M=1.90, SD=1.29). They also recognized the strengths of a password management program: it can facilitate creating unique passwords (Question 2, Mdn=1, M=1.91, SD=1.23) and strong passwords (Question 4, Mdn=2, M=2.04, SD=1.29) that can be easily organized (Question 3, Mdn=2, M=1.93, SD=1.25). This can be done without resorting to using less secure methods of recording passwords (Question 6, Mdn=2, M=2.16, SD=1.42) or relying solely on memory (Question 7, Mdn=2, M=2.18, SD=1.37) and running the risk of forgetting passwords (Question 8, Mdn=2, M=2.25, SD=1.33). Students were not discouraged from using KeePass because it required its own password to be memorized (Question 12, Mdn=4, M=3.86, SD=1.34).

These results also indicate that students were able to identify the weaknesses of a password management program. These weaknesses include: losing the master password would result in a loss of access to all passwords (Question 10, Mdn=3, M=2.75, SD=1.16), an attacker who uncovers the master password would have access to all passwords (Question 5, Mdn=2, M=2.15, 1.20), and the application and user data must be carried with the user to other computers (Question 11, Mdn=3, M=3.18, 1.33). However, the primary advantage of a password management program--increasing security--did not receive as strong a participant response (Question 9, Mdn=2, M=2.36, SD=1.29) as may be expected.

Reasons for Using KeePass

Participants were also asked to respond why they would choose to use KeePass. A list of five options was given, and participants could select all that applied to them. Table 2 illustrates reasons why participants would choose to use KeePass.

Table 2. Reasons Participants Would Use KeePass

Question	Percentage
13. I do not have to memorize multiple passwords	76.2%
14. It's easy to use	75.2%
15. I do not have to write down my passwords on paper	55.4%
16. Using KeePass makes my account safer	51.5%
17. None of the above	3.0%

Students again identified the advantages of password management programs (Question 13 and Question 15) along with KeePass' ease of use (Question 14). When the responses of Table 2 were cross tabulated by employment there was little difference for Questions 13, 14, and 16 (the largest difference between employed and unemployed students for these three questions was only 2.5%). Question 15 accounted for the largest difference, with 50.8% (31 of 61) of those employed who said that they would use KeePass because they would not have to write down their passwords, while 62.5% (25 of 40) of those not employed said that this was a reason why they would use it. When these responses were cross tabulated by gender, 25 out of 37 females (67.6%) responded that KeePass enabled them to not have to write down their passwords (Question 15) while only 31 out of 64 males (48.4%) gave this as a reason why they would use it.

Once again students did not rate using KeePass as an activity that made their accounts safer (Question 16). A cross tabulation indicates that only 46.9% of males (30 of 64) said that KeePass made their accounts safer, while 59.5% of females (22 of 37) said it made their accounts safer. In addition, 42.4% of community college students (14 of 33) said that that KeePass made their accounts safer, compared to 55.9% (38 of 68) of university students.

Reasons for Not Using KeePass

Students were also asked to respond why they would not use KeePass. A list of eight options were given, and they could select all that they felt applied to them. Table 3 illustrates reasons why students would not choose to use KeePass.

The reasons listed in Table 3 indicate that students were aware of the weaknesses of password management programs, most notably that the loss of the KeePass password to an attacker would compromise all passwords (Question 18), that KeePass' usage is limited to only computers that have access to the program and the user's data (Question 21), and that forgetting the KeePass password would restrict access to all user passwords (Question 23). Question 19 may reveal an inconvenience—the KeePass application must first be launched when a password is needed—that students considered too burdensome.

Table 3. Reasons Participants Would Not Use KeePass

Question	Percentage
18. Someone could access all of my passwords if they uncover my KeePass password	66.3%
19. It is quicker for me to type in my passwords than to open KeePass to look up my passwords	56.4%
20. I already have all of my passwords memorized	53.5%
21. I can use any computer to access my account instead of only using a computer that has access to my KeePass information	45.5%
22. I am good at memorizing passwords	35.6%
23. I am afraid I will forget the KeePass password	28.7%
24. I already use strong passwords	27.7%
25. None of the above	5.9%

Questions 20 and 22 indicate that students feel comfortable relying on their memory for password retrieval. Students were also asked to self-report the number of computer accounts they used that required a password. The number of passwords reported (N=101, M=11.58, SD=10.00) is similar with other research on the number of user passwords. The range of passwords reported was from 59 to 1.

When the responses of Table 3 were cross tabulated by school, gender, and employment status it revealed several interesting findings. Employment seemed to play a factor in student responses. In Question 20 those students not employed said that they would not use KeePass, because they already had all of their passwords memorized (70%, 28 of 40) compared to those who were employed (42.6%, 26 of 61). In addition, employed students (14 of 61) were less likely to not use KeePass because they were good at memorizing passwords (Question 22) compared to those who were not employed (22 of 40), or 23.0% vs. 55.0%. In addition, employed students were less likely (37.7%, or 23 of 61) to not use KeePass, because they were restricted to using a computer that had KeePass or their data accessible (Question 21) compared to those who were not employed (57.5%, or 23 of 40).

Students attending a community college indicated that they have much better memories (Question 22) than those attending a university (45.5% or 15 of 33 vs. 30.9% or 21 of 68), yet they do not use strong passwords (5 of 33 or 15.2%) compared to students attending a university (23 of 68 or 33.8%), as seen in Question 24.

Males also said (Question 24) they already use a strong password (21 of 64 or 32.8%) compared to females (7 of 37 or 18.9%). Yet males are more fearful of forgetting the KeePass password (34.4%, 22 of 64) than females (18.9%, 7 of 37).

Future Plans for Using KeePass

Table 4 illustrates the student's responses regarding their future plans for using KeePass. Almost 3 in 10 participants either will use or already use a password management program, while 5 in 10 remain undecided. The remaining 2 in 10 will not use the program.

The cross tabulation analysis of Table 4 reveals that there is very little difference between genders regarding if they will, will not, or have not decided to use KeePass. For those students who are employed, there also is little difference between not using Keepass or being undecided. However, there was a larger difference between those employed who said that they would use KeePass (31.1%, 19 of 61) compared to those not employed (20.0%, 8 of 40).

A larger difference is seen between students based on the school that they attended. The larger number of students attending a university (30.9%, 21 of 68) said they would use KeePass, compared to only 18.2% (6 of 33) of those attending a community college. Also, participants at a community college (60.6%, 20 of 33) were more undecided than those attending a university (45.6%, or 31 of 68).

Table 4. Participant's Plans for Using KeePass

Question	**Percentage**
I have not decided	50.5%
Yes	26.7%
No	19.8%
I already use KeePass or a similar program	3.0%

6. DISCUSSION

Prior research had indicated that relatively few users have chosen to use password management applications to create strong passwords and protect them. The study by Gaw and Felten (2006) of 49 users who were told to bring "anything you use to help you remember your passwords" revealed that only six participants brought aids, none of which was a password management application. This led Gaw and Felten to conclude that these applications were "relatively unpopular".

For this study only 3% of the student participants already used a password management application, supporting the conclusion of Gaw and Felten. Were they "relatively unpopular" because users had rejected them as being unsuitable, or because users lacked prior exposure to these applications? The results of this study seem to indicate that once users receive instruction and training regarding password management applications followed by actual use of the application, the benefits become apparent. More students indicated that they would use a password management application like KeePass (26.7%) than those who said they would not (19.8%), and half of the students (50.5%) were unsure of which action they would take. This leads to the conclusion that the reason for the small number of users of password management application is not because they have tried the application and found it to be unsuitable; instead, they simply were not familiar with the application.

The results of this study indicating that once users receive instruction and training in a security application the benefits become apparent may have broader implications for security awareness instruction and user training, particularly in higher education. Training is emphasized by many researchers, including Long (1999), Tobin and Ware (2005), Werner (2005), Witson (2003), Yang (2001) and others. Although Long (1999) advocated that security instruction should begin as early as kindergarten, most researchers state that higher education should be responsible for providing security awareness instruction, including Crowley (2003), Mangus (2002), Null (2004), Tobin and Ware (2005), Valentine (2005), Werner (2005), and Yang (2001). This instruction and training is important not only to meet the current demands of securing systems but also to prepare students for employment in their respective fields, according to Werner (2005). Long (1999) maintained that the need for organizations to develop appropriate policies requires all decision makers to have a certain level of awareness of standards for security.

One area of additional study is to examine in greater detail the responses towards security technology as it relates to gender, type of school, and employment, as well as other factors. For example, in this study 70% of unemployed students said that they would not use KeePass because they already had all of their passwords memorized, compared to only 42.6% of those employed. In addition, only 23% of employed students said that they would not use KeePass because they were good at memorizing passwords compared to 55% of those unemployed who said they were good at memorizing passwords. Additional research may reveal if there is security training instruction at workplaces that are having a positive impact on user attitudes and practices towards security.

A final area for additional study may be alternative password management applications, particularly those that are not restricted to a local computer. In both the survey data as well as student responses the need to carry both the password management application and user data with them at all times in order to have access to passwords was a barrier to acceptance. Future research may look at other types of password management applications that do not have this limitation in order to determine if these applications are more appealing to users.

7. CONCLUSION

The results of this study seem to indicate that once users receive instruction and training regarding password management applications followed by actual use of the application, the benefits of managing multiple strong passwords may become apparent. This leads to the conclusion that the reason for the small number of users of password management application is not because they have tried the application and found it to be unsuitable; instead, they simply were not familiar with the application. This may have broader implications for security awareness instruction and user training, particularly in higher education.

8. REFERENCES

Brown, A., Bracken, E., Zolccoli, S., & Douglas, K. (2004). Generating and remembering passwords. *Applied Cognitive Psychology , 18* (6), 641-651.

Bunnel, J., Podd, J., Henderson, R., Napier, R., & Kennedy-Moffat, J. (1997). Cognitive, associative and conventional passwords: Recall and guessing rates. *Computers and Security , 16* (7), 641-657.

Burnett, M., & Kleinman, D. (2006). *Perfect passwords: selection, protection, authentication.* Burlington, MA: Syngress.

Charoen, D., Raman, M., & Olfamn, L. (2008). Improving end user behaviour in password utilization: An action research initiative. *Systemic Practice and Action Research , 21* (1), 55-72.

Crowley, E. (2003). Information systems security curricular development. *Conference on Information Technology Education* (pp. 249-255). Lafayette, IN: ACM.

Dhamija, R., & Perrig, A. (2000). Deja vu: A user study using images for authentication. *Proceedings of the 9th USENIX Security Symposium.* Denver, CO.: USENIX.

Frincke, D., & Bishop, M. (2004). Joining the security education community. *IEEE Security and Privacy , 2* (5), 61-63.

Gaw, S., & Felten, E. (2006). Password management strategies for online accounts. *Symposium on Usable Privacy and Security* (pp. 44-55). Pittsburgh, PA: Association for Computing Machinery.

Goldberg, J., Hagman, J., & Sazawal, V. (2002). Doodling our way to better authentication. *Proceedings of Ext. Abstracts CHI 2002* (pp. 868-869). New York, NY: ACM Press.

Govindarajulu, N., & Madhvanath, S. (2007). Password management using doodles. *ICMI'07* (pp. 236-239). Nagoya, Aichi, Japan: ACM.

Knight, P. (1995). *Assessment for Learning in Higher education.* London: Kogan Page.

Kruger, H., Steyn, T., Medlin, B., & Drevin, L. (2008). An empirical assessment of factors impeding effective password management. *Journal of Information Privacy and Security , 4* (4), 45-59.

Long, C. L. (1999). A socio-technical perspective on information security knowledge and attitudes. *Ph.D. dissertation, The University of Texas at Austin, United States-- Texas* .

Mangus, T. (2002). A study of first-year community college students and proposed responsible computing guide. *Ph.D. dissertation, Union Institute and University, United States--Ohio* .

Miller, G. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychology Review , 63*, 81-97.

Neath, I. (1998). *Human memory: An introduction to research, data, and theory.* Pacific Grove, CA: Brooks/Cole.

Null, L. (2004). Integrating security across a computer science curriculum. *Journal of Competing Science is in Colleges , 19* (5), 170-178.

Pastore, M., & Dulaney, E. (2006). *CompTIA Security+ Study Guide* (3ed ed.). Indianapolis: Wiley.

Reichl, D. (2010). *KeePass features*. Retrieved Jan 16, 2010, from keepass.info: http://keepass.info/features.html

Sasse, M., & Brostoff, S. W. (2001). Transforming the 'weakest link': A human/computer interaction approach to usable and effective security. *BT Technology Journal , 19* (3), 122-131.

Schneier, B. (2004). *Secrets and lies: Digital security in a networked world.* New York: Wiley Computer Publishing.

Tari, F., Ozok, A., & Holden, S. (2006). Password management, mnemonics, and mother's maiden names: A comparison of perceived and real shoulder-surfing risks between alphanumeric and graphical passwords. *Second Symposium on Usable Privacy and Security (SOUPS).* New York, NY: ACM Press.

Tobin, D., & Ware, M. (2005). Using a windows attack intRusion emulator (AWARE) to teach computer security awareness. *10th Annual SIGSCE Conference on Innovation and Technology in Computer Signs Education* (pp. 213-217). Caparica, Portugal: SIGSCE.

Topkara, U., Atallah, M., & Topkara, M. (2007). Passwords decay, words endure: Secure and re-usable multiple password mnemonics. *Proceedings of the 2007 ACM symposium on Applied computing* (pp. 292-299). Seoul, Korea: ACM.

Valentine, D. W. (2005). Practical computer security: A new service course based upon the national strategy to secure cyberspace. *Conference on Information Technology Education* (pp. 185-189). Newark, NJ: ACM.

Vu, K.-P., Proctor, R., Bhargav-Spantzel, A., Tai, B.-L., Cook, J., & Schultz, E. (2007). Improving password security and memorability to protect personal and organizational information. *International Journal of Human-Computer Studies* (65), 744-757.

Werner, L. (2005). Redefining computer literacy in the age of ubiquitous computing. *Conference on Information Technology Education* (pp. 95-99). Newark, NJ: ACM.

Whitson, G. (2003). Computer security: Theory, process and management. *Journal of computing sciences in colleges , 18* (6), 57-66.

Wiedenbeck, J., Waters, J., Birget, J., Brodskiy, A., & Memon, N. (2005). Authentication using graphical passwords: Effects of tolerance and image choice. *SOUPS '05: Proceedings of the 2005 Symposium on Usable Privacy and Security* (pp. 25-31). New York: ACM Press.

Yan, J., Blackwell, A., Anderson, R., & GRant, A. (2004). Password memorability and security: Empirical results. *IEEE Security and Privacy , 2* (5), 25-31.

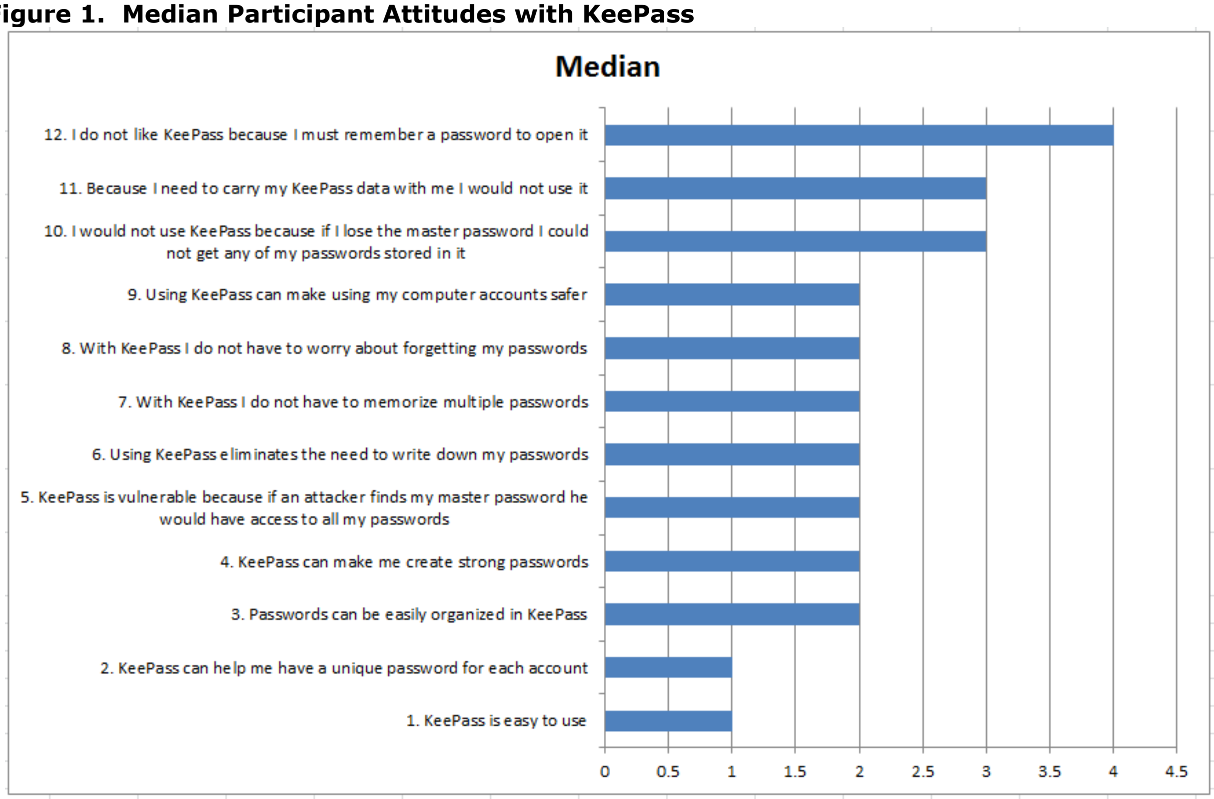
Yang, T. A. (2001). Computer security an impact on computer science education. *Journal of Computing Sciences in Colleges , 18* (6), 233-246.

Appendix

Table 1. Participant Attitudes with KeePass

Question	Median	Mean	Std Dev
1. KeePass is easy to use	1	1.90	1.29
2. KeePass can help me have a unique password for each account	1	1.91	1.23
3. Passwords can be easily organized in KeePass	2	1.93	1.25
4. KeePass can make me create strong passwords	2	2.04	1.29
5. KeePass is vulnerable because if an attacker finds my master password he would have access to all my passwords	2	2.15	1.20
6. Using KeePass eliminates the need to write down my passwords	2	2.16	1.42
7. With KeePass I do not have to memorize multiple passwords	2	2.18	1.37
8. With KeePass I do not have to worry about forgetting my passwords	2	2.25	1.33
9. Using KeePass can make using my computer accounts safer	2	2.36	1.29
10. I would not use KeePass because if I lose the master password I could not get any of my passwords stored in it	3	2.75	1.16
11. Because I need to carry my KeePass data with me I would not use it	3	3.18	1.33
12. I do not like KeePass because I must remember a password to open it	4	3.86	1.34

Figure 1. Median Participant Attitudes with KeePass



A ‘Rainmaker’ Process for Developing Internet-based Retail Businesses

Alan S. Abrahams
abra@vt.edu

Tirna Singh
tsingh05@vt.edu

Business Information Technology Department
Virginia Tech
Blacksburg, Virginia 24061, USA

Abstract

Various systems development life cycles and business development models have been popularized by information systems researchers and practitioners over a number of decades. In the case of systems development life cycles, these have been targeted at software development projects within an organization, typically involving analysis, design, programming, testing, and deployment. For business development models, phase-based approaches for developing generic businesses have been proposed. With the recent surge in popularity of online businesses, and particularly web-based hosted services for online start-ups, a gap has emerged in the information systems literature for development processes specifically tailored to developing internet-based retail businesses. In this paper, we present such a process, which we dub the ‘Rainmaker’ process for developing internet-based businesses. We demonstrate, through a real case study, how the Rainmaker model can be successfully applied.

Keywords: information systems development processes, e-commerce, entrepreneurship, web start-ups

1. INTRODUCTION

“***Rainmaker*** (n): an executive … with exceptional ability to attract clients … increase profits, etc.:” *Dictionary.com*

E-commerce courses have surged in popularity in recent years (Ngai et al, 2005; Moshkovich et al, 2006). As e-commerce educators increasingly indulge in active, experiential learning (Changchit et al, 2006; Braender et al, 2009; Kor and Abrahams, 2007; Williams and Chin, 2009; Preiser-Houy and Navarette, 2007; Terwiesch and Ulrich, 2009), an opportunity arises to reflect on the development of internet-based businesses in the classroom, and to propose reusable processes that generalize the pedagogical techniques employed. In this paper, we introduce a pedagogic model for the development of internet-based businesses, which provides a useful and general framework to students and entrepreneurs for creating an online retail business. The model is dubbed the ‘Rainmaker’ model for two reasons: it illustrates a process for generating internet-based businesses (‘making rain’), and the repeated application of parallel technology identification and assessment in the model makes it schematically reminiscent of rainfall.

We begin with a discussion of related work, and describe why traditional systems development and business development life cycles should be tailored to the internet-based business world.

Next, we describe the Rainmaker model diagrammatically. Finally, we demonstrate the application of the model to the creation of an actual internet-based business.

2. RELATED WORK

Various systems development and business development models have been popularized by information systems researchers and practitioners over a number of decades.

In the case of systems development models, these include waterfall, iterative, prototype, exploratory, spiral, reuse, and other models – for a brief survey see Green and DiCaterino (1998). System development models have been targeted at software development projects within an organization, typically involving analysis, design, programming, testing, and deployment, and usually with a focus on information systems implementation rather than business development. Models are often tailored to particular software development paradigms – for example waterfall models were initially conceived for structured software development, iterative and reuse models were recommended as more appropriate for object-oriented or component-based software, the prototyping model became popular with the advent of drag-and-drop graphical development environments, and trial-and-error-intensive exploratory models are often used in artificial intelligence application development. The Rainmaker model introduced in this paper is targeted at a Software-as-a-Service (SaaS) paradigm, with a lesser focus on software development, and a greater focus on *business operations development* and *software selection*.

Business Operations Development

Some authors have proposed information-intensive business operations development models for generic businesses – see for example Ives and Learmonth (1984) and Ives and Mason (1990), who's suggestions that information systems be developed to support a customer service life cycle were the precursors to the vibrant, modern Customer Relationship Management (CRM) software industry. The Rainmaker model specializes this, and other business development models, by providing staged guidance on the rapid assessment of and application of particular internet-based technology areas to the creation of a web-based businesses.

Software Selection

With the recent surge in popularity of online businesses, and particularly web-based hosted services for online start-ups, a gap has emerged in the literature for pedagogic models specifically tailored to developing internet-based businesses. Internet-specific development cycles for use by educators have previously been proposed (see, for example, DeVilliers and Abrahams, 2000), but the recent surge in the availability and variety of hosted business-to-business platforms has introduced a lesser reliance on custom programming of in-house solutions, and a greater trend towards trial, evaluation, and selection of a varied array of external hosted services. Software selection – that is, identification and evaluation of available hosted services for different business functions – has therefore become an increasingly significant portion of the business operations development challenge.

Doing business on the internet now provides new operating modes that were previously unavailable. For example, information systems departments would traditionally proceed in a roughly sequential, single path manner with a time-consuming process of analysis, design, and programming for a selected project. With the increasing availability of hosted online services, businesses are now able to cheaply select and test multiple technologies and approaches – indeed many services are open source and/or free (e.g. phpBB for bulletin boards; WordPress for blogging; osTicket for issue tickets; and many others). Implementation typically involves account activation and configuration, rather than analysis, design, and programming. Occasionally, software installation (on an instructor or student's web hosting account) is required instead of account activation. Rather than simply conducting rigorous testing on software development projects, businesses are reliant on the quality assurance procedures of hosted service providers, and a business's assessment process now more often encompasses evaluation of multiple competing implementations, and re-investment in approaches that proved profitable during piloting.

The rainmaker model therefore adopts a characteristically parallel model tailored to a Web 2.0 world with bountiful cheap and easy-to-deploy options that can be inexpensively tested and accepted or discarded. The Rainmaker model is unusual amongst system and business development life cycles, in that it is tailored to the development of internet-based businesses, in particular, internet-based retail businesses.

3. THE MODEL

The complete Rainmaker Model is shown in the Appendix. Figure 1 (see appendix) provides a schematic illustration of the overall Rainmaker model. In the model, teams progress through business conception, comparison to competitors, production of a website and physical product(s), promotion of their business and products, day-to-day operation of the business, and monitoring and improvement of the organization. During each of these phases, multiple implementation options are identified, then simultaneous researched or executed – hence the parallel arrows, reminiscent of falling rain. Note that each option may be researched or implemented by a different team member, but all team members report on their findings or implementation afterwards, so that everyone can learn from the experience of others. Post- or mid-implementation reporting allows all options to be regularly assessed. Promising or successful options are reinvested in.

Parallel implementation is employed for a few reasons. Firstly, it serves a useful pedagogic purpose, exposing students to multiple alternative manifestations of a technology area, and helping them build a better general understanding of the field. Secondly, it allows best of breed solutions to emerge, via low cost determination of, and verification of, multiple alternatives. The overall Rainmaker model relies intensively on a variation of Deming's Plan-Do-Study-Act cycle (Deming 1986, 1994), in an attempt to address the concern of some authors of the lack of a cyclic evaluation component in traditional SDLCs (Polito, Watson, Berry, 2001).

The schematic depicted in Figure 1 (see Appendix) shows a birds-eye view of the Rainmaker model. Our particular concern, however, was creating a process targeted specifically at developing internet-based retail businesses, and so the Rainmaker process provides more detailed elucidations of each phase, to tailor the model for this purpose. Figures 2 through 7 in the Appendix demonstrate these refinements.

In the *Conception* phase (Figure 2), various business ideas are generated, different revenue models are proposed and corporate identity is established (for example, through definition of alternative missions and visions, and creation of various alternative logo concepts). Students are assigned to functional teams, and team members are given tasks within each team. Tasks are selected from the guideline tasks provided in the remaining phases of the Rainmaker model. Task assignment may need to be revisited repeatedly during business development, as new tasks are identified, or as alternative team members are assigned to re-attempt tasks not properly completed.

In the *Comparison* phase (Figure 3), the chosen business concept is compared to competing offerings currently available in various industries, using various assessment tools. As we shall see in the case study later (§4), one such set of competitor evaluation tools should be web-hosted competitor assessment tools, which are particularly useful for understanding the sources and nature of internet traffic to a website.

In the *Production* phase (Figure 4), the focus on online business becomes especially apparent. Website production is initiated through hosting provider identification, website design, content management solution identification, and bespoke system planning. The physical retail product to be sold is prototyped if necessary, and refined. Manufacturing options (e.g. in-house versus outsourced versus drop-shipped) are considered, and suppliers are assessed.

The *Promotion* phase (Figure 5) of the Rainmaker model involves use of both traditional and web technologies for business and product promotion. Traditional media campaigns might include direct mail, print, radio, television, and other means (e.g. posters, business cards, networking at industry events and trade shows, etc.). Web-based promotion includes identification of pay-per-click, pay-per-impression, and/or pay-per-action platforms, and then instantiation of various campaigns using these platforms (e.g. using different keywords or phrases to advertise). Social media platforms are identified and campaigns are enacted. Product data feed platforms are identified and tested, to allow product data to be fed to comparison shopping engines. Email marketing platforms are assessed, and multiple email marketing campaigns are designed and launched. Where necessary, sales management / customer-relationship-management (CRM) tools are used to organize and monitor a local or remote physical sales team.

The *Operation* phase (Figure 6) addresses the listing, shipping, and returns-handling of the physical retail product(s). Online catalogues (e.g. hosted shopping carts) are assessed and implemented, and fulfillment and reverse-logistics solutions (e.g. in-house versus outsourced) are evaluated and enacted.

The *Monitoring* phase (Figure 7) encompasses monitoring internal issues (e.g. through hosted issue tracking software), monitoring customers, and monitoring the company's website. Customers are monitored by finding and deploying customer feedback management systems, and by monitoring company and product reviews both on the company's own website and on 3rd party review sites, for instance using online reputation monitoring (ORM) systems. The business's website is monitored by employing web analytics packages to assess visitor volumes, frequency, and sources, as well as ROI of individual paid-search campaigns and other web visitor metrics (e.g. click-through-rate, bounce rate, conversion rate, cost-per-visitor, cost-per-lead, cost-per-sale, top traffic sources, top keywords, profit per thousand visitors). The availability of the website is also monitored through hosted uptime monitoring solutions.

4. CASE STUDY

To demonstrate the application of the Rainmaker model to a real scenario, this section provides a case study of an actual internet business, The Online Business Guidebook that was created during an information systems senior capstone class using the Rainmaker model. This case study is intended to act as an exemplar and guide for information systems educators. We begin with some background on the Online Business Guidebook as a experiential learning project, and then describe the project's fit with the Rainmaker process.

The Rainmaker process is a pedagogic model, intended to guide students or entrepreneurs in the creation of live online businesses. Various authors have highlighted the pedagogic value of real application environments to students in information systems courses [Chase, Oakes, and Ramsey, 2007; Chen, 2006; Gabbert and Treu, 2001; Janicki, Fischetti, and Burns, 2007; Klappholz, 2008; Martincic, 2007; McGann and Cahill, 2005; Mitra and Bullinger, 2007; Scott, 2006; Song, 1996; Tadayon, 2004; Tan and Jones, 2008; Tan and Phillips, 2003]. While in many cases the real-world client is a for-profit institution, in other cases the client is a not-for-profit organizations (community partner) and students engage in 'service learning', where they undertake a real project that provides a valuable service to the community partner [Lenox, 2008; Saulnier, 2005; Tan and Phillips, 2005]. Typically, students are involved in implemented projects for real, extant clients. In our case, in an unusual twist on service learning, the students initiated and ran a brand new internet-based not-for-profit venture, christened "The Online Business Guidebook". In an earlier variation of this course – see [Kor and Abrahams, 2007] – students developed a real, live, for-profit internet-based business. For this instantiation, the instructor suggested a not-for-profit concept instead. Historic experience had indicated that for-profit student organizations were vulnerable to debilitating squabbles amongst students over ownership shares, and were occasionally seen in a negative light by recruiters, who sometimes viewed students as maverick self-starters with personal entrepreneurial agendas. The not-for-profit format was seen as more likely to engender positive sentiments amongst both students and recruiters. In the case of recruiters, we found that they viewed student participants in the not-for-profit as talented, community-minded, corporate contributors, who possessed valuable practical skills and experience that had been developed through active involvement in a real not-for-profit.

Let us now look at the application of the Rainmaker model to the Online Business Guidebook. In the following paragraphs, we describe the actual manifestation of each process in the Rainmaker model for this particular new venture. The specific tools described are illustrative of options assessed and employed by the new Online Business Guidebook venture, but this discussion is not intended to be prescriptive, and it is recommended that other options be identified, assessed, and implemented depending on the specific needs of the particular online venture being initiated. For guidance of other alternative software platforms to consider, consult the Online Business Guidebook itself, which is a good reference, by visiting:
`www.Businessguidebook.org`

In the *Conception* phase, the Online Business Guidebook idea was chosen amongst various competing alternatives. The idea was to produce and sell a step-by-step tutorial guide describing how to start and grow an online business. Different revenue models were proposed, including revenue from printed book sales, from sponsorship, from online advertising commissions (e.g. Google Adsense), and from affiliate marketing. Each was assessed via spreadsheet simulations, and continually monitored in reality as the business progressed (see Monitoring phase later), to direct promotional campaign investments (see Promotion phase later) to the most lucrative revenue stream. Corporate identity was established by agreeing a mission ("to provide public education on how to start and grow an online business") and a vision ("to reach 50,000 readers within 12 months"). Multiple alternative logo concepts were generated and a final design was chosen, which provided a tangible and credible brand for participants to relate to. Students were assigned to one of five functional teams: Finance, Sales, Marketing, Publishing & Distribution, and Web. Team leaders were appointed and each team member was assigned specific tasks from the available tasks suggested by later phases of the Rainmaker model.

In the *Comparison* phase, the Online Business Guidebook concept was compared to competing offerings, including magazines, books, websites, and tradeshows. This helped establish benchmarks on what was realistically achievable (e.g. in terms of readership, advertising rates, and other metrics), as well as clarify the organization's unique selling point. Students determined that their offering would be tutorial-based (rather than conventional entrepreneur-targeted magazines which are story-based), and "by students, for students" (being hipper and more vibey than a conventional textbook, through the use of color, icons, stock art, and actual vendor logos). Comparative websites such as compete.com, quantcast.com, and spyfu.com were used to gain insight into competitor's customer demographics, affinities of the competitor's online audience to other websites, keyword marketing tactics being employed by competitors, and other competitor activity.

In the *Production* phase students assessed and chose a hosting provider and prototyped multiple website designs before settling on their favorite. Joomla was identified from available options as their preferred content management solution, and the students set about writing and releasing multiple pieces of content in a standard format using Joomla's Article Manager. Various community-oriented features were created: a discussion forum was incorporated in the site (using PHPBB), a blog was added (using WordPress), and following suggestions by Kane and Fichman (2009), a wiki for consumer-contributed content was set up (using MediaWiki). Custom information system development was avoided wherever possible, in favor of hosted solutions which were robust and quick to deploy. For the physical product multiple prototypes (different cover designs and internal layouts) were produced, from which the most attractive was chosen. Quotes were requested from multiple different printing vendors before choosing a preferred supplier.

During the *Promotion* phase the students contemplated and ran multiple traditional and online campaigns. For direct mail campaigns, multiple postcard designs were generated, and the favorite was sent to a small pilot target audience using a web-based direct mail service, Click2Mail. Following quality concerns with the first pilot, a second pilot was conducted. Satisfactory results with the second pilot prompted reinvestment in the second campaign, to roll it out to a full scale audience. For print media, press coverage was obtained in local newspapers and the alumni magazine. Multiple business card concepts were designed, and the nicest were printed, and distributed at entrepreneurship events and industry tradeshows which the students attended. Large, full-color, portable roll-up vinyl displays were purchased to attract attention at these events or during physical on-campus or off-campus campaigns. The Monitoring phase of the Rainmaker model (see later) was run concurrently to monitor the success of each campaign: in particular, web analytics tools and customer feedback forms helped quantify responses to each campaign. For web-based promotion, the students deployed and assessed campaigns on multiple pay-per-click, pay-per-impression, and pay-per-action platforms, including Google, Facebook Advertising, and AT&T's Ingenio. Different keyword campaigns (e.g. "entrepreneur", "internet business", "start my own business") were created, each with a small initial daily budget, and reinvestment was made in successful campaigns and keywords. Email marketing platforms were assessed, and alpha and beta campaign designs were created and tested on the chosen email marketing platforms, Ace of Sales, and Mailchimp.com. Multiple hosted sales management tools were reviewed, but cost and complexity considerations led to the choice of Excel for sales management. Over 400 sales calls were conducted, by a team of 9 students using a common script and 'brag sheet'. The sales team shared successes and failures in weekly meetings, and documented all leads and results in a spreadsheet.

For the *Operation* phase, both in-house and outsourced fulfillment models were tested. For in-house fulfillment, Google Checkout was used for product listing, payment processing, and order management, and a student was responsible for shipping and returns processing using the Google Checkout administrative interface. A portion of inventory was also sent to Amazon, for storage and fulfillment from a remote warehouse. Revenues, costs, and effort of each fulfillment approach were compared.

The *Monitoring* phase involved monitoring internal and external items. For internal issues, multiple ticketing systems were evaluated. An open-source issue ticketing system (osTicket) was deployed, and used to assign tasks to team members, and monitor completion. For customer monitoring, a visual drag-and-drop tool, SmartFormer, was used to configure custom web-forms to receive feedback from readers, advertisers, and distributors. Public product reviews (e.g. on Amazon) for both the organizations own product and its competitors were also monitored, with the intention of funneling good customer suggestions into future product designs. Google Alerts was used as a basic customer intelligence gathering ("buzz monitoring" / "online reputation management") system. Google Analytics and awStats were used to monitor website traffic, assess campaign performance, and make campaign termination or reinvestment decisions.

5. LIMITATIONS

While the Rainmaker model and Online Business Guidebook example case provide a useful framework for internet-based business development, a number of limitations exist.

Firstly, though multiple hosted software categories were featured, the Online Business Guidebook organization did not necessarily pursue all available business operation development options. It is recommended that educators allow their students to exercise some level of creativity in the pursuit of existing and newly emerging alternatives.

Also, while the Rainmaker model is appropriate for retail organizations, it requires refinement or alteration for other types of internet businesses where no physical product is sold.

Furthermore, emerging hosted technology areas will need to be included in updated versions of the Rainmaker model as these new technologies arise and mature.

Regarding guidelines and timelines for execution, as well as evaluative instruments, readers are encouraged to contact the author for suggestions.

Finally, this paper does not provide a listing of vendors who provide the various platforms described in the model, nor does it provide a tutorial on how to employ each technology platform described in the model. We refer the reader instead to the Online Business Guidebook (available at no cost at: `www.businessguidebook.org`) for this information, which may be helpful to educators who are applying the Rainmaker model in practice in a classroom setting.

6. CONCLUSION

The Rainmaker process is a comprehensive, though not exhaustive, pedagogic tool for developing an internet-based retail business. The process customizes previous system and business development methods with particular assignments drawn from available modern hosted internet services. Parallelism is employed to enhance education by identifying, implementing, and comparing multiple options, thereby promoting overall industry knowledge rather than merely specific vendor familiarity. This paper used a real-world case study, The Online Business Guidebook case, to illustrate that the Rainmaker model is sufficient to adequately describe and replicate the business development process for a new online retail business. It is hoped that the Rainmaker model will provide a useful pedagogic tool for educators teaching e-commerce and entrepreneurship classes.

7. ACKNOWLEDGMENTS

The authors are particularly grateful to the major corporate sponsors of the Online Business Guidebook project – Ace of Sales, Progress Printing, LeClair Ryan, and Deloitte – and to various classes of BIT 4454 (Business Analysis Seminar in IT) at Virginia Tech. Numerous other individuals and organizations supported the class in this endeavor – we refer the reader to the Acknowledgements page of The Online Business

Guidebook for a detailed listing of the dozens of contributors to whom we are grateful.

8. REFERENCES

Braender L.M., Kapp C.M., and Yeras J. (2009). "Using Web Technology to Teach Students about Their Digital World", *Journal of Information Systems Education*, 20(2), pp. 145-154.

Changchit, C., Cutshall R., and Gonsalves G.C (2006). "Designing an Electronic Commerce Course: An Effort to Balance Theory and Practice", *Information Systems Education Journal*, 4(108), pp. 1-7.

Chase J. D., Oakes E., S. Ramsey (2007). "Using live projects without pain: the development of the small project support center at Radford University", *ACM SIGCSE Bulletin*, 39(1), pp. 469-473, March 2007

Chen, B. (2006). "Teaching Systems Analysis and Design: Bringing the Real World into the Classroom", *Information Systems Education Journal*, 4(84). pp. 1-8. http://isedj.org/4/84/. ISSN: 1545-679X.

De Villiers C. and A.S. Abrahams (2000). "A Model for Addressing the Development of Electronic Commerce Applications in Information Systems Courses", *Journal of Informatics Education Research*, 2(1), pp. 1-8, Spring 2000.

Deming, W.E. (1986). Out of the Crisis. Massachusetts Institute of Technology Center for Advanced Engineering Study. Cambridge, Massachusetts, p. 88.

Deming, W. E. (1994). The New Economics for Industry, Government, Education. Second Edition. Massachusetts Institute of Technology Center for Advanced Engineering Study. Cambridge, Massachusetts, p. 131.

Ives B., and Learmonth G.P. (1984). "The information system as a competitive weapon", *Communications of the ACM*, 27(12), December, p.1193-1201.

Ives B., and Mason R. (1990). "Can Information Technology Revitalize Your Customer Service?" *Academy of Management Executive*, 4(4), pp. 52-69.

Gabbert P., and K. Treu (2001). "Reality check: working with meaningful projects in and out of the classroom", *Journal of Computing Sciences in Colleges*, 17(2), pp. 191-198, December 2001.

Green D. and DiCaterino A. (1998). "A Survey of System Development Process Models", Center for Technology in Government, University at Albany / SUNY, February, pp. 1-13. Available at: http://www3.ctg.albany.edu/publications/reports/survey_of_sysdev/survey_of_sysdev.pdf

Janicki T.N., Fischetti D., and A.T. Burns (2007). "Incorporating Real World Projects and Emerging Technologies into One MIS Capstone Course", *Information Systems Education Journal*, 5(24). http://isedj.org/5/24/. ISSN: 1545-679X, pp. 1-8.

Kane, G.C., and Fichman R.G. (2009), "The Shoemaker's Children: Using Wikis for Information Systems Teaching, Research, and Publication". *MIS Quarterly*, 33(1), March. pp. 1-17.

Klappholz, D. (2008). "Organizing and delivering 'real projects for real clients courses'". *Journal of Computing Sciences in Colleges*, 23(4), April, pp. 254-256. ISSN:1937-4771

Kor, P. and A.S. Abrahams (2007). "Teaching Information System Students To Be Entrepreneurs: A Dot.com Case Study", *Communications of the Association for Information Systems*, 20, Article 32, October 2007, pp. 1-32

Lenox T.L. (2008). "The Value of Service-Learning in the CIS Curriculum: A Case Study". *Information Systems Education Journal*, 6(66). pp. 1-9. http://isedj.org/6/66/. ISSN:1545-679X.

Lovett, A. (ed.), (2010). "The Online Business Guidebook – Spring 2010", The Online Business Guidebook, Inc. Blacksburg, Virginia, ISBN 978-0-9843181-0-0. Also available at: http://www.businessguidebook.org/

Martincic, C.J. (2009). "Combining Real-World Internships With Software Development Courses". *Information Systems Education Journal*, 7(33). http://isedj.org/7/33/. ISSN: 1545-679X, pp. 1-10.

McGann, S., & M. Cahill (2005). "Pulling it all Together: An IS Capstone Course for the 21st Century emphasizing experiential and conceptual aspects, soft skills and career readings", *Issues in Information Systems*, 6(1), pp. 1-7.

Mitra S, and T.A. Bullinger (2007). "Using formal software development methodologies in a real-world student project: an experience report", *Journal of Computing Sciences in Colleges*, 22(6). pp. 100-108. June 2007.

Moshkovich H.M., Mechitov A.I., and Olson D.L. (2006). "E-commerce and the Undergraduate MIS Curricula: an Exploratory Study", *Journal of Information Systems Education*, 17(2), pp. 185-194.

Ngai E.W.T., Gunasekaran A., and Harris A.L. (2005). "The Maturing of E-commerce Education in Our Curricula", *Journal of Information Systems Education*, 16(1), pp. 5-8.

Polito T, Watson K., and Berry, R. (2001). An Exploratory Identification Of Differences Between Deming's PDSA Improvement Cycle And The System Development Life Cycle (SDLC) Proceedings of the 2001 Annual Meeting of the Midwest Region of Decision Sciences Institute. Dearborn Inn, Dearborn, Michigan. April 26-28 2001. Pp. 53-55.

Preiser-Houy L. and Navarrete C.J. (2007). "Exploring the Learning in Service-Learning: A Case of a Community-Based Research Project in Web-Based Systems Development", *Journal of Information Systems Education*, 17(3), pp. 273-284.

Saulnier, B.M. (2005). "Service Learning in Computer Information Systems: 'Significant' Learning for Tomorrow's Computer Professionals". *Information Systems Education Journal*, 3(10). pp. 1-12. http://isedj.org/3/10/. ISSN: 1545-679X.

Scott, E. (2006). "Systems Development Group Project: A Real World Experience", *Information Systems Education Journal*, 4(23), pp. 1-10.

Singh T., et al. (2009). The Online Business Guidebook - Fall 2009. The Online Business Guidebook, Inc. Blacksburg, Virginia. ISBN 978-1607253921.

Song, K-S. (1996). "Teaching software engineering through real-life projects to bridge school and industry", *ACM SIGCSE Bulletin*, 28(4), December, pp.59-64.

Tadayon, N. (2004). "Software Engineering Based on the Team Software Process with a Real World Project." *Journal of Computing Sciences in College*s. 19(4). pp. 133-142.

Tan J., and Jones M. (2008). "A case study of classroom experience with client-based team projects", *Journal of Computing Sciences in Colleges*, 23(5), pp.150-159, May 2008.

Tan J., and Phillips J. (2003). "Challenges of real-world projects in team-based courses", *Journal of Computing Sciences in Colleges*, 19(2). December 2003.

Tan J., and Phillips J. (2005). "Incorporating service learning into computer science courses", *Journal of Computing Sciences in Colleges*, 20(4), pp.57-62, April 2005.

Terwiesch C. and Ulrich K. (2009). Innovation Tournaments: Creating and Selecting Exceptional Opportunities. Harvard Business School Press, Cambridge, MA.

Williams J. and Chinn S.J. (2009). "Using Web 2.0 to Support the Active Learning Experience", *Journal of Information Systems Education*, 20(2), pp. 165-174.

Editor's Note:

This paper was selected for inclusion in the journal as an ISECON 2010 Meritorious Paper. The acceptance rate is typically 15% for this category of paper based on blind reviews from six or more peers including three or more former best papers authors who did not submit a paper in 2010.

Appendix 1: Process Diagrams

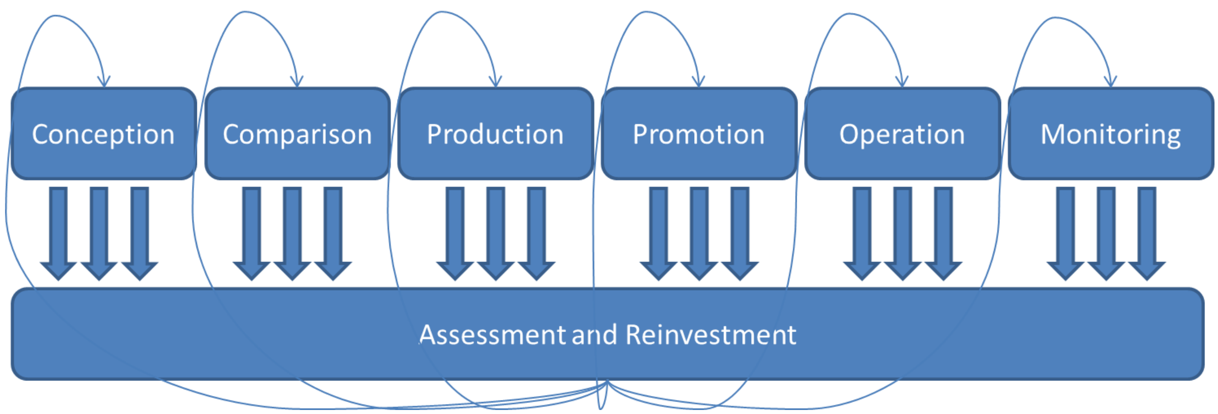


Figure 1: The Rainmaker Online Business Development Model (high level view)

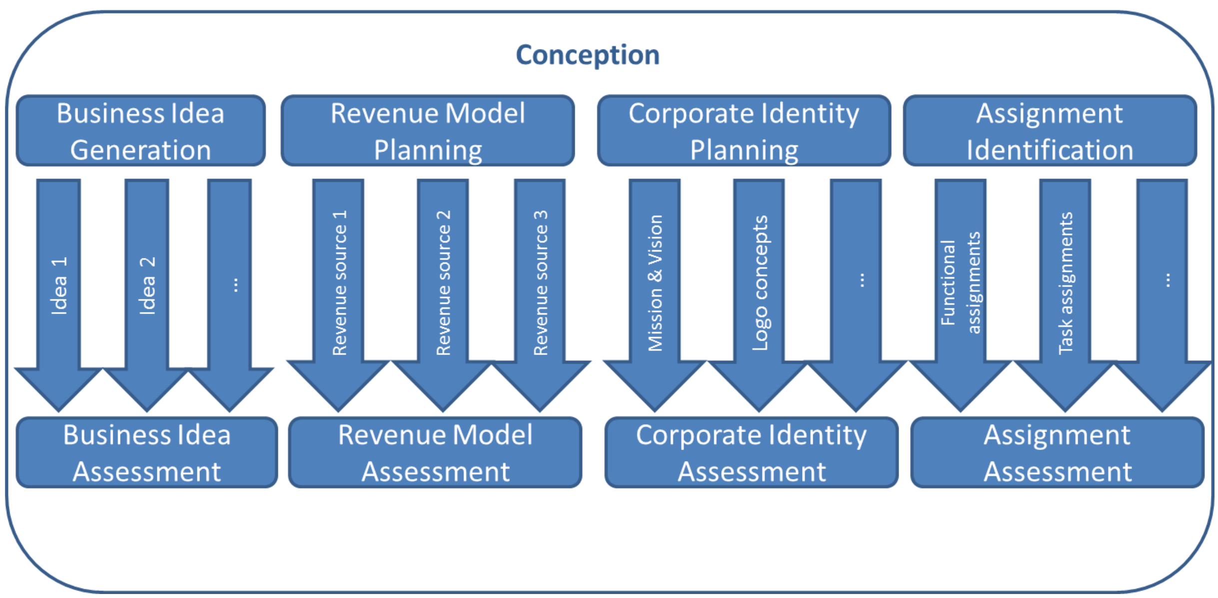


Figure 2: Conception Phase of the Rainmaker Model

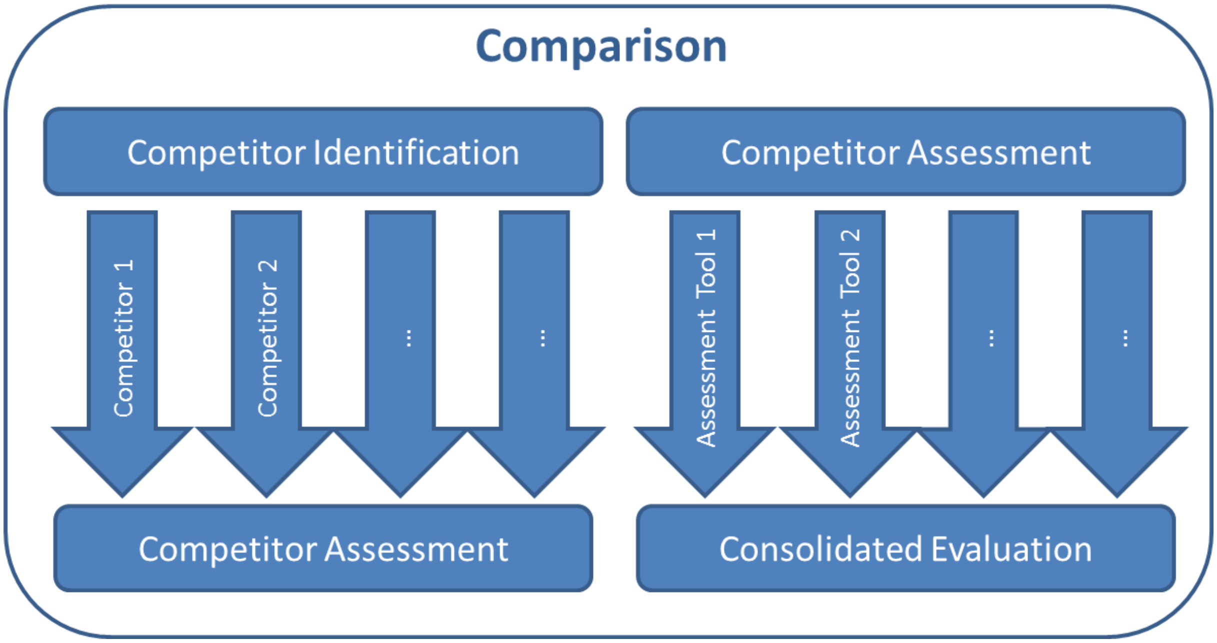


Figure 3: Comparison Phase of the Rainmaker Model

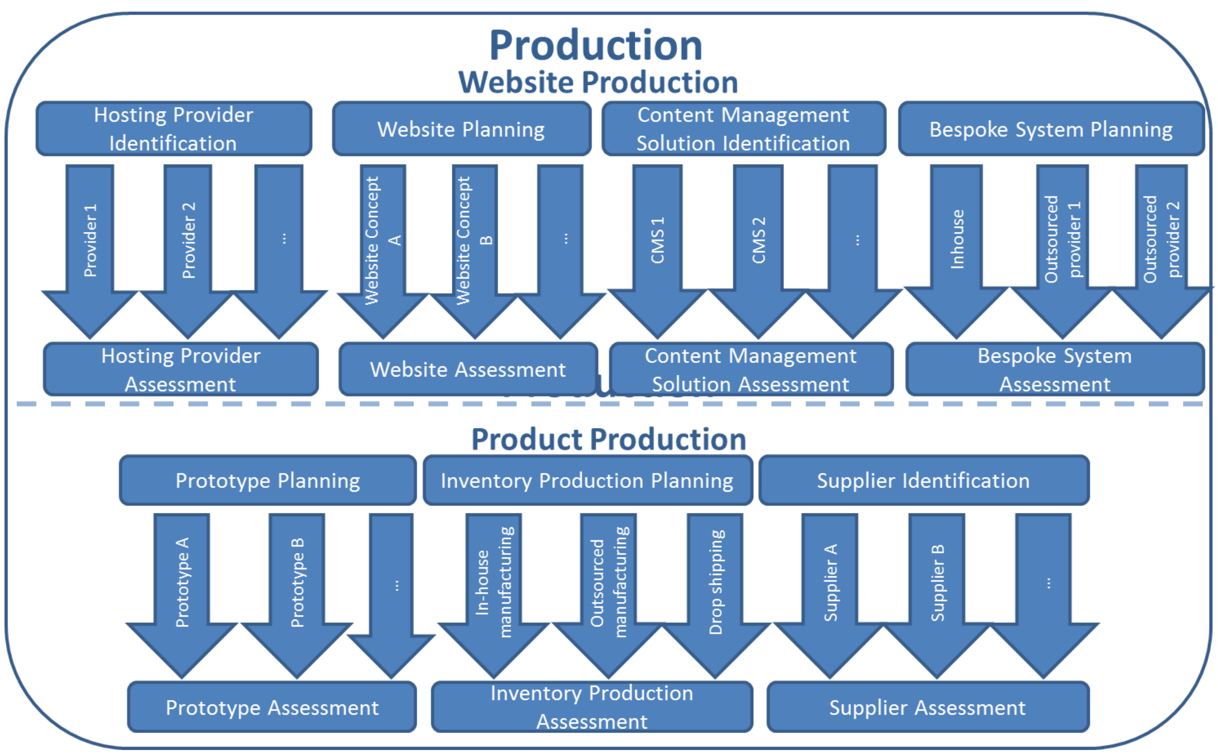
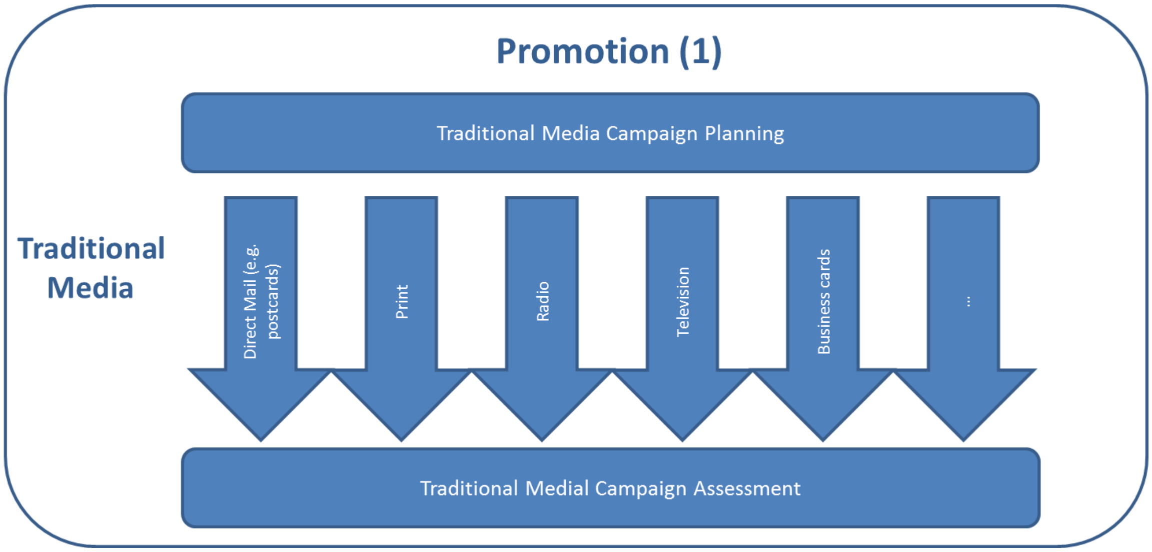
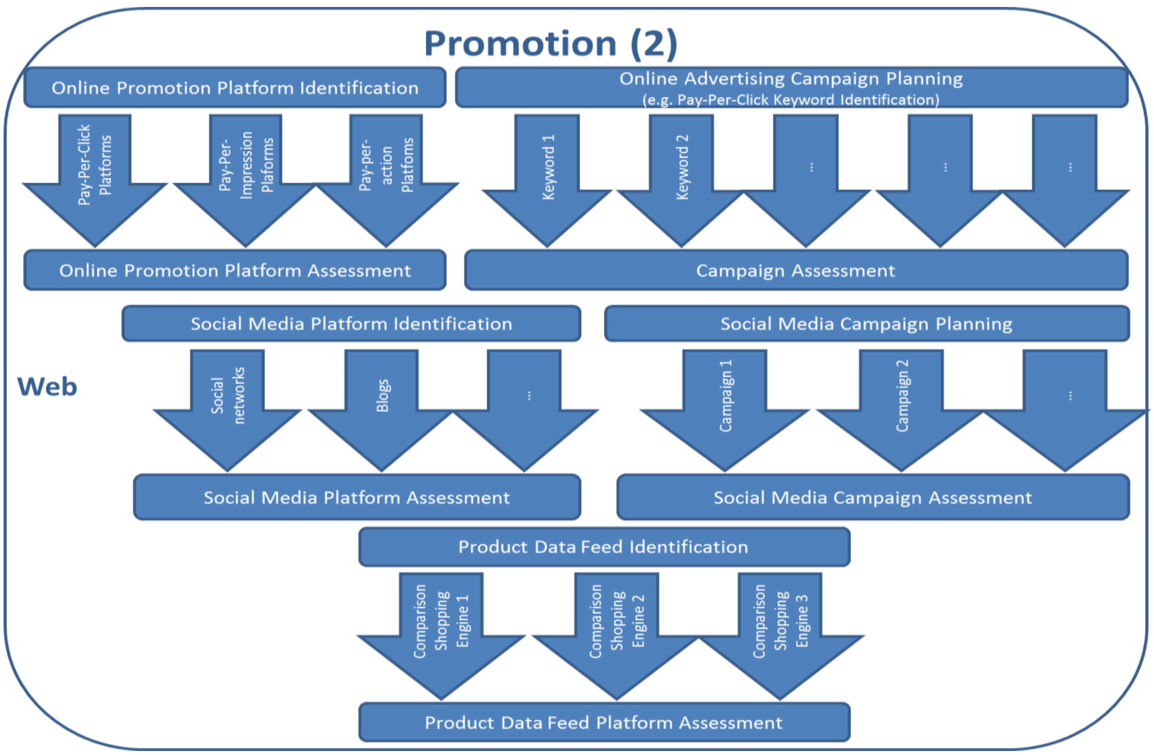


Figure 4: Production Phase of the Rainmaker Model


Promotion (1)
Traditional Media
Traditional Media Campaign Planning
Direct Mail (e.g. postcards)
Print
Radio
Television
Business cards
…
Traditional Medial Campaign Assessment


Promotion (2)
Web
Online Promotion Platform Identification
Pay-Per-Click Platforms
Pay-Per-Impression Plaforms
Pay-per-action Platfoms
Online Promotion Platform Assessment
Online Advertising Campaign Planning
(e.g. Pay-Per-Click Keyword Identification)
Keyword 1
Keyword 2
…
…
…
Campaign Assessment
Social Media Platform Identification
Social networks
Blogs
…
Social Media Platform Assessment
Social Media Campaign Planning
Campaign 1
Campaign 2
…
Social Media Campaign Assessment
Product Data Feed Identification
Comparison Shopping Engine 1
Comparison Shopping Engine 2
Comparison Shopping Engine 3
Product Data Feed Platform Assessment

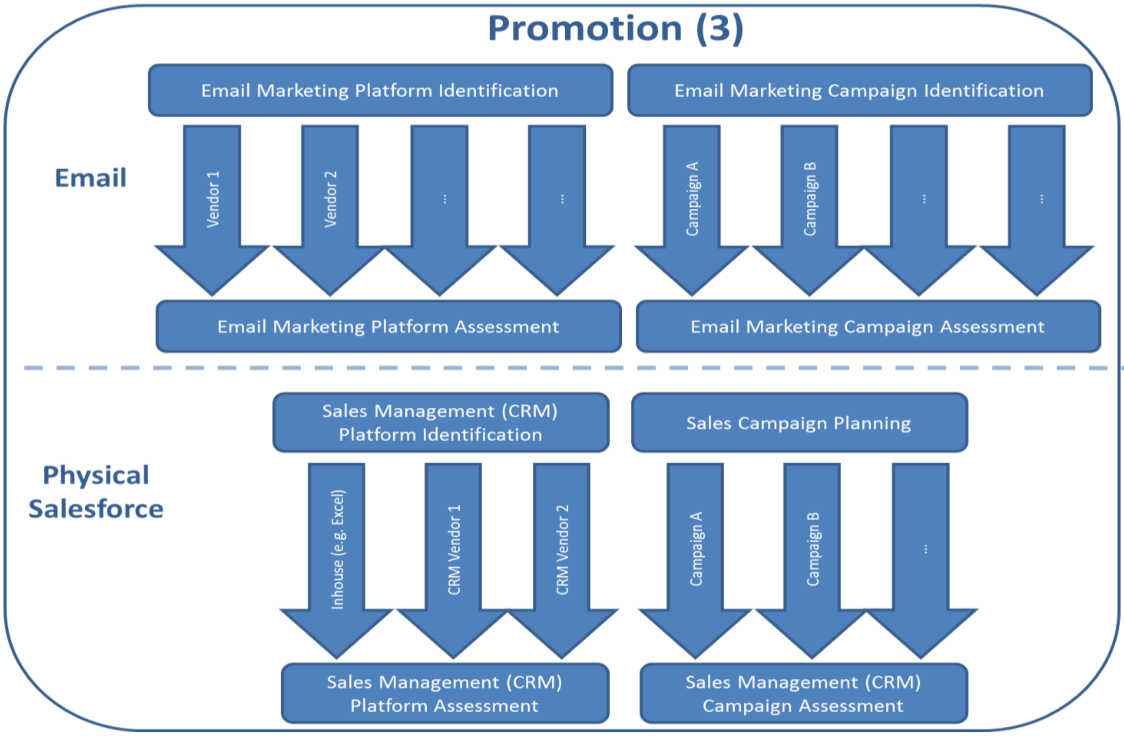


Figure 5: Promotion Phase of the Rainmaker Model

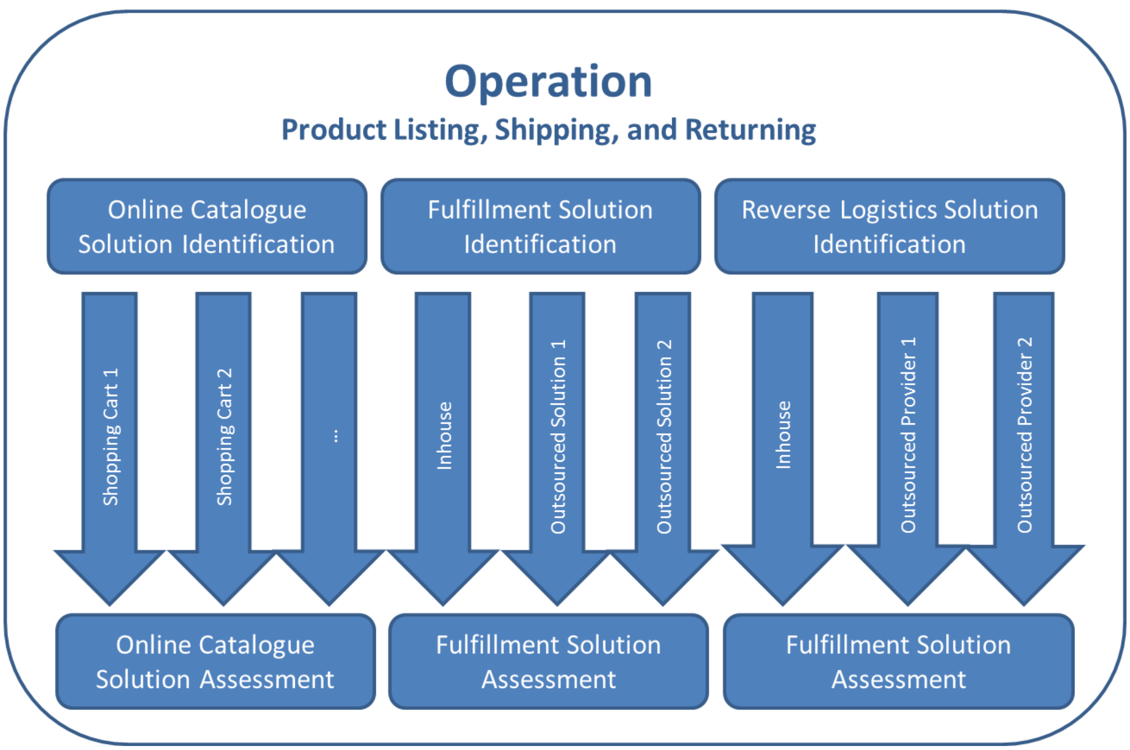


Figure 6: Operation Phase of the Rainmaker Model

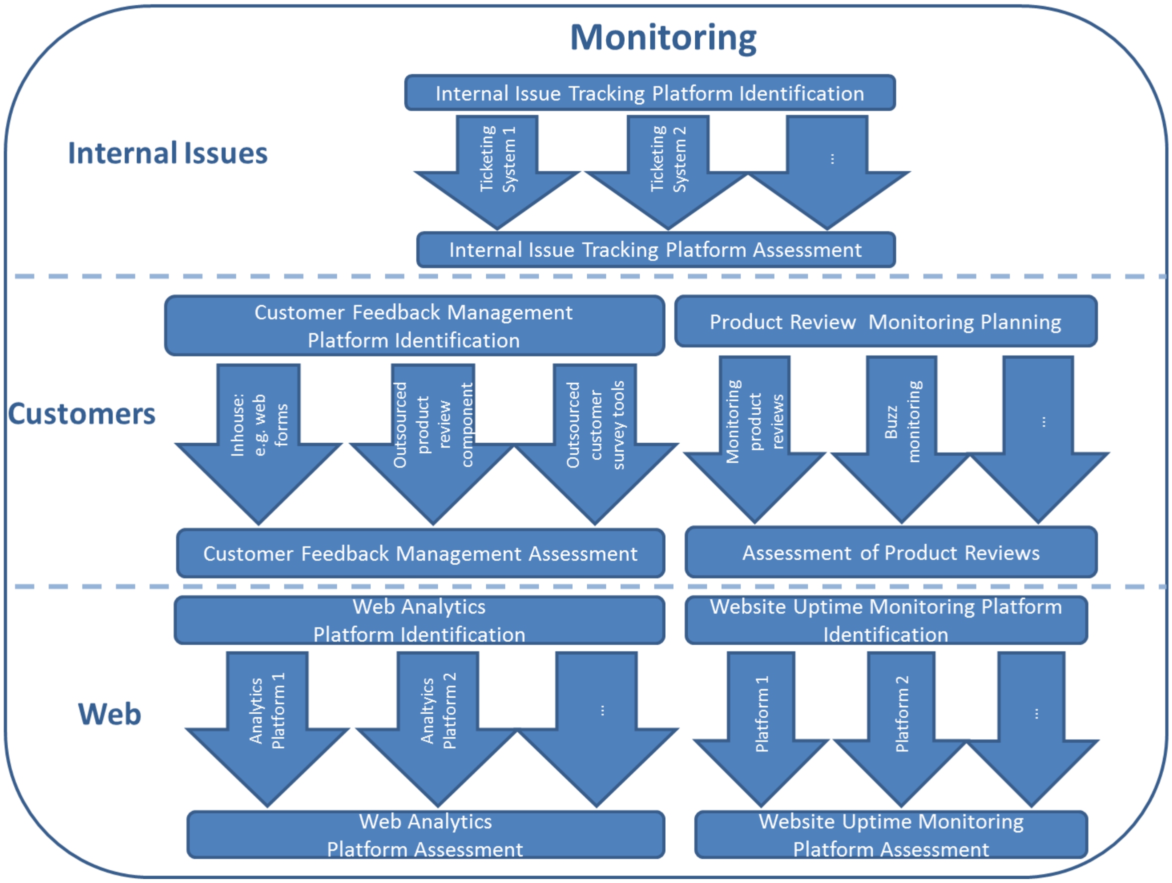


Figure 7: Monitoring Phase of the Rainmaker Model

Texting and the Efficacy of Mnemonics: Is Too Much Texting Detrimental?

Randy Ryker
randy.ryker@nicholls.edu
Information Systems

Chuck Viosca
chuck.viosca@nicholls.edu
Marketing

Shari Lawrence
shari.lawrence@nicholls.edu
Finance

Betty Kleen
betty.kleen@nicholls.edu
Information Systems

Nicholls State University
Thibodaux, LA 70310, USA

Abstract

The rapidly growing social phenomenon of texting has attracted researchers from diverse disciplines who seek to study its effects. Texting typically involves the use of abbreviations and other shortcuts to craft cell phone messages. Collectively, these abbreviations and shortcuts are referred to as "text speak." The authors observe that some mnemonics are very similar in form to various types of text speak. Based on the similarities, it is hypothesized that heavy texters will be more receptive to mnemonics and thus benefit more from them. The results of this study indicate that there is a significant relationship between heavy texting and the efficacy of mnemonics; however, the relationship is negative rather than positive as was hypothesized. Possible explanations, implications, and future research are discussed.

Keywords: texting, mnemonics, text speak

1. INTRODUCTION

For American teens, texting has become the preferred mode of communication, surpassing face-to-face contact, email, and even voice calls. This finding was confirmed in a recent survey conducted by the PewResearchCenter(Lenhart, Ling, Campbell, & Purcell, 2010) and it supports an earlier report that appeared in Wired.com (Ganapati, 2008). The Pew Report also indicates that half of all teens send 50 or more text messages a day, and one third send more than 100 texts a day. This rapidly growing social phenomenon has attracted researchers from diverse disciplines who seek to study its effects.

Some teachers, parents, and language experts have emphasized the strong negative effects of texting on literacy (Brown-Owens, Eason &Lader, 2003; Humphrys, 2007; Lee, 2002; Thurlow, 2006; Vosloo, 2009). On the other hand, researchers have recently argued that texting has either a positive effect on literacy (Plester, Wood & Joshi, 2009), or that it is neutral and is nothing to worry about (Crystal, 2009; Drouin, & Davis, 2009). Given these diverse findings and opinions, it is likely that this research stream will be active for some time.

Rather than a broad study of the effects of texting on literacy, this paper focuses quite specifically on the effects of texting on the efficacy of mnemonics. Texting typically involves the use of abbreviations and other shortcuts to craft cell phone messages. Collectively, these abbreviations and other shortcuts are referred to as "text speak" and include: acronyms (LOL, OMG), contractions (txt vs. text), shortenings (bro vs. brother), g clippings (goin vs. going), letter/number homophones (2nite), nonconventional spellings (fone vs. phone), accent stylization (elp vs. help), and initialisms (Nabisco refers to the National Biscuit Company) (Plester et al., 2009).

The authors observed that some mnemonics are very similar in form to various types of text speak. One common mnemonic for remembering a list of items consists of an easily remembered acronym, or phrase with an acronym, that is associated with the list. For example, to remember the five dimensions of employee satisfaction: variety, identity, significance, autonomy and feedback one can use the mnemonic VISA F. The authors wondered if heavy texters would be more receptive to, and thus benefit more from mnemonics; because of their heavy use of text speak. If this were found to be true, then educators should be encouraged to increase their use of mnemonics and even to create new ones if appropriate for their learning goals.

A review of the literature on texting and on mnemonics revealed no studies that examined the relationship between them. This exploratory study is a first attempt to address that area of research.

The research goal is as follows: (1) identify subjects who are high texters and low texters or talkers, and (2) conduct an experiment to test for differences in performance between these two groups when they are exposed to acronyms or other mnemonics.

Hypothesis: Heavy texters, when exposed to a new acronym or other mnemonic, will remember the content associated with the acronym or mnemonic significantly better compared to low texters or talkers.

2.METHODOLOGY

Sample

Undergraduates from a regional university in the southern U.S. were selected to participate. Nine classes were surveyed, totaling 479 participants. Each class was randomly assigned to either a control group, an acronym group, or a mnemonic group.

A questionnaire was developed to collect data about the subjects' texting behavior including: the number of texts they typically sent per day and per week as well as their grade point average (GPA) and demographics. To facilitate the pretest posttest matching of responses, subjects were also asked to write their student ID on the questionnaire.

Pretest

A pretest of the subjects' knowledge of the college of business (COB) learning goals was given at the beginning of a class along with the questionnaire. The pretest involved asking students to write the four learning goals of the COB on the questionnaire.

The grading procedure involved reading the response to each of the four goals, and assigning either zero points for no answer or an incorrect answer, one point for a partially correct answer, and two points for a fully complete answer. For example, if a subject wrote down all four goals correctly he or she would receive a score of 8 resulting in a possible range of scores from zero to eight. Each subjects' responses were graded first by a graduate student and subsequently by one of the authors. Any discrepancies were resolved.

Treatments

Once students completed the pretest, they were shown a PowerPoint presentation with a recorded narration about the COB learning goals. The use of one narrator to record PowerPoints for the three treatment groups was employed to reduce bias that could be introduced by having different instructors make the presentations. The narrator not only read the goals but also mentioned that the subjects may be tested on these goals at a later date. The control group received the following narrated bullet points:

1. Students will be effective written and oral **communicators** with the ability to use appropriate technologies to enhance their communications.
2. Students will be able to apply **critical thinking** in making sound business decisions.
3. Students will be able to demonstrate competency in the **core** business disciplines.
4. Students will demonstrate awareness of **ethical** issues in business.

The acronym group received the same narrated bullet points but with a different pattern of boldings, and were told via the narration that an acronym (CCCE) may help them remember the goals:

1. Students will be effective written and oral **C**ommunicators with the ability to use appropriate technologies to enhance their communications.
2. Students will be able to apply **C**ritical thinking in making sound business decisions.
3. Students will be able to demonstrate competency in the **C**ore business disciplines.
4. Students will demonstrate awareness of **E**thical issues in business.

The mnemonic group also received the same narrated bullet points but with yet another pattern of boldings, and were told via the narration that a mnemonic learning aid (CommCritCorE) may help them remember the goals:

1. Students will be effective written and oral **Comm**unicators with the ability to use appropriate technologies to enhance their communications.
2. Students will be able to apply **Crit**ical thinking in making sound business decisions.
3. Students will be able to demonstrate competency in the **Core** business disciplines.
4. Students will demonstrate awareness of **E**thical issues in business.

Posttests and Dependent Variables

Posttest1 was conducted at the end of the same one-hour class in which the pretest was conducted, by having the subjects once again write down the four learning goals of the COB. Posttest2 was conducted two days later, at the beginning of the next class, by having subjects write down the four learning goals of the COB.

The dependent variables of interest are related to the change in memory/awareness of the COB learning goals from the pretest to posttest1 and posttest2. The dependent variables are defined as:

diff1 = posttest1 – pretest

diff2 = posttest2 – pretest

Variable diff1 thus represents the change in memory/awareness one hour after a PowerPoint treatment, and variable diff2 represents the change two days after a PowerPoint treatment.

Texters versus Talkers

Subjects were divided into high texters, which we refer to as (texters), and low texters which we refer to as (talkers), based on quartiles. Students whose total number of text messages sent in a week fell in the fourth quartile were coded as texters, while those who fell in the first quartile were coded as talkers. Those who fell in the middle two quartiles were coded as tweeners; however, the tweeners were not a focus of this study.

To test the hypothesis the authors selected cases with texters and talkers. The file was then split into three PowerPoint treatments: a control group, an acronym group, and a mnemonic group. Then, for each of the three treatment groups, we tested the dependent variables for differences between texters and talkers.

3. RESULTS

A total of 479 students participated in the study. Of these, 245 (51%) were male and 234 (49%) were female. The average age was 21 years. They were primarily sophomores, juniors and seniors (95%). The average GPA was 2.85 on a 4-point scale. The average number of text messages sent per day was 55 and the average per week was 364.

To focus on differences between high texters and low texters or talkers, the first and fourth quartiles were selected based on the number of text messages sent per week. This resulted in 131 low texters from the first quartile, which are referred to as "talkers," and 118 high texters from the fourth quartile which are referred to as "texters". Table 1 provides a profile of the talkers and texters. The only notable difference between these two groups, other than the number of texts they send, is that the texters have been texting for an average of 5 years while the talkers have been texting for an average of only 3 years.

Table 1. Talkers and Texters

	Msg/ Week	Years Texting	Age	GPA
Talkers				
Mean	20	3	24	2.96
Median	14	2	22	3.00
Min	0	0	18	2.00
Max	50	10	59	4.00
N*	131	127	131	120
Texters				
Mean	1091	5	21	2.70
Median	1000	5	21	2.70
Min	450	2	19	1.00
Max	7000	12	25	4.00
N*	118	118	114	115

* not every subject answered every question

A comparison of talkers versus texters in each of the experimental groups after one hour revealed that talkers consistently scored higher than texters (Table 2). However, none of the differences were significant.

Table 2. Improvement After 1 Hour Posttest1 – Pretest Scores

Experimental Condition	Mean	N	F	Sig.
Control				
Talkers	2.82	40	1.32	.71
Texters	2.66	36		
CCCE				
Talkers	2.20	39	.630	.43
Texters	1.92	39		
CommCritCorE				
Talkers	2.94	52	.987	.32
Texters	2.52	42		

After two days talkers again scored higher than texters and they were significantly higher in the acronym (CCCE, .03) and mnemonic (CommCritCorE, .05) groups (Table 3).

There was no significant difference in the control group (.77). This finding is the opposite of what was hypothesized. Talkers in this study, after a two day period, appear to have benefitted more from both the acronym and mnemonic than texters.

Potential covariates such as students' GPA and gender were examined in these analyses. None of the variables were found to be significantly related to any of the treatment variables; therefore, they were excluded from all subsequent analyses.

Table 3. Improvement After 2 Days Posttest2 – Pretest Scores

Experimental Condition	Mean	N	F	Sig.
Control				
Talkers	3.05	36	.084	.77
Texters	2.90	31		
CCCE				
Talkers	2.32	31	4.95	.03
Texters	1.44	29		
CommCritCorE				
Talkers	3.20	40	3.90	.05
Texters	2.28	35		

4. DISCUSSION OF RESULTS

An interesting finding of this research is that there is a relationship between heavy texting and the efficacy of mnemonics. The surprising aspect is that the relationship is negative, i.e. heavy texters score significantly lower than talkers when both were exposed to the COB learning goals and were provided with a mnemonic to help them remember the goals.

Several possible explanations of the results can be considered. First, it is possible that high texters may be desensitized to mnemonics. In other words, heavy texters due to their heavy use of "text speak," which is similar in form to acronyms and mnemonics, do not find new acronyms and mnemonics interesting enough to server as effective memory aids.

A second possible explanation relates to the size of an individual's vocabulary. Although the English language contains over one million words, the average person's vocabulary includes no more than thirty-five thousand words (Crystal, 2007). Perhaps there is also a "text speak" vocabulary limit and heavy texters have reached their limit and are thus less likely to add a new acronym or mnemonic that resembles text speak. Additional research would need to be conducted to determine if there is some kind of "natural limit" to the number of mnemonics or "texting shortcuts" for the average person.

A third possible explanation is related to a relatively new stream of research on how living with technology is altering our brains (Carr, 2010; Small & Vorgan, 2008; Stone, 2009). Carr (2010) discussed how he believes the Internet and its frantic superficiality is destroying our powers of concentration and he cites some scientific evidence to support his beliefs. Small and Vorgan (2008) also cite evidence that "the current explosion of digital technology not only is changing the way we live and communicate but also is rapidly and profoundly altering our brains."

The heavy texters in the current study are constantly being interrupted by receiving and responding to text messages. They averaged 1091 texts per week (Table 1). For subjects who were awake for 16 hours per day that is the equivalent of about 10 texts per hour, or one every 6 minutes. Such heavy texters would appear to be in a state of "continuous partial attention" a term coined by Stone, 2009.

Continuous partial attention is different from simple multi-tasking. With simple multi-tasking, at least one of the activities is somewhat automatic or routine, like eating lunch. That activity is then paired with another activity that is automatic or with an activity that requires cognition, like writing an email or talking on the phone. We multi-task to be more productive. With continuous partial attention, on the other hand, the motivation is a desire not to miss anything. Individuals are engaged in two activities that both demand cognition. For example, people talking on the phone and driving, or texting while listening to a lecture.

Continuous partial attention describes a state where individuals scan for an opportunity for any type of contact at every given moment. This places their brain in a heightened state of stress where their adrenalized "fight or flight" mechanism kicks in. Some research suggests that the end result of such chronic and prolonged techno-brain burnout can be the reshaping of underlying brain structure (Small &Vorgan, 2008).

Regardless of whether or not heavy texting has the effect of altering the brain, the state of continuous partial attention produced by heavy texting may diminish one's ability to concentrate and thus to remember material that is presented to them.

One other issue to consider about this research is whether or not allowing the students to devise their own mnemonic would improve their performance versus having the instructor supply one. Evidence suggests that this may be the case. For example, researchers have found that subjects who produce their own mnemonics have better recall because self-generation produces better understanding (Bobrow & Bower, 1969); creates easier images (Dickel & Slak, 1983); and makes mnemonics more meaningful to the individual subject (Garten & Blick, 1974). A future research question raised by the current study would then be does the performance of heavy texters differ from low texters when individuals in both groups generate their own mnemonics.

5. RESEARCH CONTRIBUTION

If the findings of this study are confirmed, teachers of heavy texters should be cautioned about promoting the use of mnemonics as a means to memorize course material.

The implications of the study may be of interest to teachers and students in a variety of fields. Certainly the fields of Information Systems and Computer Science are well known for their heavy use of acronyms. In addition, many other disciplines make use of acronyms. This is suggested by the number of internet sites dedicated to mnemonics designed for various disciplines including: anatomy, chemistry, physiology, and biochemistry (www.valuemd.com/mnemonics.htm) , mathematics (www.onlinemathlearning.com/math-mnemonics.html), and home schooling (www.betterendings.org/homeschool/fun/mnemonic.htm).

6. LIMITATIONS

This was an exploratory experiment in a classroom setting. Although efforts were made to reduce bias in the experiment, the more controlled experimental conditions of a laboratory would be helpful to confirm the findings.

This study is also limited by the age range and educational level of the subjects enrolled in business classes of a regional university in the Southern United States.

7. CONCLUSIONS

Although there have been numerous recent studies regarding the effects of texting on literacy (Crystal, 2009; Drouin& Davis, 2009; Plester et al., 2009; Vosloo, 2009), this is the first study to examine the effects of texting on the use of mnemonics. The implications of the findings may be of interest to teachers and students of any discipline that makes use of mnemonics.

For future research, it is probably more important to confirm that there is a significant relationship between heavy texting and mnemonics rather than focusing on why the relationship is negative.

For those who are interested in why the relationship is negative, there are a number of lines for follow-on inquiry. Several of these areas for future research were mentioned in the Discussion of Results Section. One other area is worth noting. Texting is an informal mode of communication that may be considered superficial and lacking the nuances of a formal language. Students who make heavy use of texting may adopt a pattern of behavior which lacks attention to detail and this may explain why they have more difficulty remembering material associated with mnemonics.

Additional research along one or more of these lines of inquiry would help to clarify and extend this study.

8. REFERENCES

Bobrow, S. A. & Bower, G. H. (1969).Comprehension and recall of sentences, *Journal of Experimental Psychology,*80, 455-461.

Brown-Ownes, A., Eason, M., &Lader, A. (2003). What effect does computer-mediated communication, specifically instant messaging, have on 8th grade writing competencies? Retrieved November 27, 2008, from http://web.archive.org/web/20030821214021/http://www.usca.edu/medtech/courses/et780/may03/groupprojects/cmc-im.html

Carr, N. (2010). The Shallows: What the internet is doing to our brains. W.W. Norton & Company.

Crystal, D. (2007). Words WordsWords. Oxford University Press, USA.

Crystal, D. (2009). Txting: The Gr8 Db8. Oxford University Press, USA.

Dickel, M. J. & Slak, S. (1983). Imagery vivdness and memory for verbal material.*Journal of Mental Imagery*, 7(1), 121-126.

Drouin, M., & Davis, C. (2009). R u txting? Is the use of text speak hurting your literacy? *Journal of Literacy Research*, 41(1), 46-67.

Ganapati, P. (2008). Texting finally more popular than calling among U.S. mobile users. *Wired.com*. Retrieved on October 1, 2008, from http://blog.wired.com/gadgets/2008/09/us-finally-catc.html

Garten, J. A. & Blick, K. A. (1974).Retention of word-pairs for experimenter-supplied and subject-originated mnemonics.*Psychological Reports*, 35, 1099-1104.

Humphrys, J. (2007). I h8 txt msgs: How texting is wrecking our language. *Daily Mail*, September 24. Retrieved February 24, 2009, from http://www.dailymail.co.uk/news/article-483511/I-h8-txt-msgs-How-texting-wrecking-language.html

Lee, J. (2002). I think, therefore IM. *New York Times*, September 19, G.1. Retrieved January 10, 2009, from http://query.nytimes.com/gst/fullpage.html?res=9F06E5D71230F92AA2575AC0A9649C8B63&sec=&spon=&pagewanted=all

Lenhart, A., Ling, R., Campbell, S., & Purcell, K. (2010). Teens and mobile phones. *The Pew Research Center's Internet & American Life Project*. Retrieved May 5, 2010, from http://www.pewinternet.org/Reports/2010/Teens-and-Mobile-Phones.aspx

Plester, B., Wood, C., & Joshi, P. (2009). Exploring the relationship between children's knowledge of text message abbreviations and school literacy outcomes. *British Journal of Developmental Psychology*, 27(1), 145-161.

Small G. & Vorgan, G. (2008). Your iBrain: How technology changes the way we think. Scientific American. Retrieved December 9, 2010, http://www.scientificamerican.com/article.cfm?id=your-ibrain

Stone, L. (2009). Beyond simple mulit-tasking: Continuous partial attention. Retrieved December 9, 2010, from http://lindastone.net/2009/11/30/beyond-simple-multi-tasking-continuous-partial-attentntion/

Thurlow, C. (2006). From statistical panic to moral panic: The metadiscursive construction and popular exaggeration of new media language in the print media. *Journal of Computer-Mediated Communications*, 11(3), 667-701.

Vosloo, S. (2009). The effects of texting on literacy: Modern scourge or opportunity? *Shuttleworth Foundation Issue Paper*. Retrieved February 28, 2010 from http://vosloo.nte/wpcontent/uploads/pubs/texting_and_literacy_apr09_sv.pdf

Cloud Computing in the Curricula of Schools of Computer Science and Information Systems

James P. Lawler
lawlerj@aol.com
Pace University
New York, New York 10038, USA

Abstract

The cloud continues to be a developing area of information systems. Evangelistic literature in the practitioner field indicates benefit for business firms but disruption for technology departments of the firms. Though the cloud currently is immature in methodology, this study defines a model program by which computer science and information systems students can learn needed skills in cloud computing strategy and technology. The study emphasizes problem-solving skills relative to elements of performance, process and security of cloud computing systems that are limiting investment in the cloud computing paradigm. This study benefits educators in schools of computer science and information systems considering curricula enhancement for the cloud and will also benefit technology departments of firms that will be needing skilled students once cloud computing becomes mainstream in industry.

Keywords: business process management (BPM), cloud, cloud computing, cloud service provider (CSP), computer science and information systems curricula, infrastructure-as-a-service (IaaS), platform-as-a-service (PaaS), program management methodology, service-oriented architecture (SOA), software-as-a-service (SaaS)

1. INTRODUCTION

Cloud computing is defined as:

> "Model of computing for enabling convenient, on-demand network access to a shared pool of configurable computing resources, [including] networks, servers, services, storage and [systems] that can be provisioned rapidly and released with minimal management effort or [cloud] service provider [CSP] interaction" (National Institute of Standards and Technology [NIST], 2009); or

> "Style of computing where elastic and scalable [technology] enabled [resources] are delivered as a service to external [clients through] Internet technologies (Smith et.al., 2009, p. 22)."

Cloud computing is delivered in high level models of the following:

> Infrastructure-as-a-Service (IaaS), furnishing on-demand services, such as CPU, networking and storage (e.g. Amazon – Elastic Compute Cloud / EC2 [hardware]);

> Platform-as-a-Service (PaaS), furnishing services, such as application framework and development tools to deploy, host and maintain systems (e.g. Google – App Engine or Microsoft Azure [software and tools]); and

> Software-as-a-Service (SaaS), furnishing pay-as-you-go remote services to deploy, host and manage network systems accessible to clients on the Internet (e.g. Cisco WebEx, Google

Mail, or salesforce.com [systems]) (Yachin and Patterson, 2009).

Cloud computing might be concurrently considered as delivered in models of application-as-a-service, database-as-a-service, information-as-a-service, infrastructure-as-a-service, integration-as-a-service, management-as-a-service, platform-as-a-service, process-as-a-service, security-as-a-service, storage-as-a-service and testing-as-a-service (Linthicum, 2010, p. 11).

Cloud computing is deployed either as a public (CSP) cloud, a private (business firm behind firewall of firm) cloud, or a hybrid of public and private clouds (National Institute of Standards and Technology, 2009). Features of the cloud are effectively in fast elasticity for faster resource scalability, increasingly on-demand resource self-service, location-independent pooling of resources in a multi-tenant or single tenant plan, pay-as-you-go for resource subscription, and ubiquitous network access of the resources. Cloud is essentially an evolution of autonomic computing, clustering, grid computing, utility computing and virtualization that includes connectivity to resources and services hosted on the cloud of the Internet instead of on local technology (CIO, 2009). Literature indicates the cloud to be the latest major phase in information technology (Neal, Moschella, Masterson and O'Shea, 2009, p. 4), though it is not a new technology (Conry-Murray, 2009).

Cost efficiency is a cited benefit of cloud computing because business firms may not have to buy further hardware and software (McMillan, 2009) or further invest in generic systems if not internal staffing – a capital expense model vs. a operational expense model, especially in data center server virtualization and in the public cloud (Babcock, August, 2009). Effectiveness in faster deployment of features of current or new systems for frameworks in the cloud – cloud-as-service (CaaS) – might improve the business operations platforms or business processes of firms (Fingar, 2009) as they respond to customers in the marketplace (Neal, Moschella, Masterson and O'Shea, 2009, p. 40). Firms may not have to further invest in flexible infrastructure if resource scalability of systems is managed in the cloud in minutes during peak periods (Hurwitz, Bloor, Kaufman and Halper, 2010), and they may not have to invest in over-provisioning of systems in non-peak periods (Reese, 2009), furnishing processing responding to the market. Functionality of the infrastructure if not innovation of leading edge technology might be a benefit to business firms leveraging services of the cloud (Akamai, 2009). Literature indicates the cloud to be beneficial especially for medium and small-sized firms (i.e. <$500 million in revenue) that do not have funds for investing in large-sized infrastructures or innovative methods (Gage, 2009).

Estimates indicate that 31% of business firms considered cloud computing in 2008 (Babcock, November, 2009, p. 2), but 52% of firms considered if not dedicated funds to the cloud in 2009 (Korzeniowski and Jander, 2009, p. HB14). Deployments of firms are expected to be increasing 66% in IaaS, 63% in SaaS and 59% in PaaS in 2010 (Greengard, 2009), and 31% of firms indicated SaaS to be the highest investment in the cloud since 2008 (Dubey, Mohiuddin, Rangaswami and Baijal, 2009). Deployments are higher on private clouds instead of public clouds (Babcock, April, 2009) and are expected to be higher into 2012 (Burt, 2009). Deployments of cloud computing are expected to be a $160 billion market in 2012 (Crossman, 2009) in a growth of 25% of all incremental investment in technology in 2012 – a growth indicated to be the largest since the Internet (Hamm, 2009). Cloud computing is clearly considered to be developing as an enabling model for improving the processes of firms, such that schools of computer science and information systems might include it in curricula.

2. BACKGROUND

Despite the bullish estimates on cloud computing, 48% of business firms in the earlier forecasts have not considered nor dedicated funds to the cloud (Korzeniowski and Jander, 2009, HB14) as of the first quarter of 2010. Exclusive of cloud service provider (CSP) technology firms, business firms are hesitant in investing in cloud computing, due to concern on governance and maturity (Kontzer, November, 2009). Frequently indicated in the literature are problems of integration of non-cloud services and systems, performance of cloud systems, privacy of proprietary information in cloud infrastructures and systems, and risk and security of cloud technology (Korzeniowski and Jander, 2009, p. HB5). 63% of business firms indicated performance, and 75% indicated security, as major problems in migrating systems to the cloud (Waxer, 2009), and firms further indicated hesitancy in forecasting cost savings from cloud computing systems

(Johnson, 2009). Cloud essentially is in its infancy, as indicated in Figure 1 of the Appendix.

Governance of cloud computing services is in its infancy (Linthicum, August 31, 2009). Integration of cloud delivered services and non-cloud on-premise services and systems, in an effective mix of processes, is a concern for business firms in the management of processes serviced by non-cloud and cloud systems (Smith et.al., 2009, p. 10). Performance of the cloud continues to be a concern for firms, as indicated in non-reliability of systems of CSP IaaS and PaaS technology firms (Linthicum, August 21, 2009). Portability of systems resident with CSPs is an issue (Linthicum, November, 2009). Protection and security of cloud information and infrastructure of systems, in conformance with firm controls, metrics, governmental regulations and industry standards defined on non-cloud systems, are especially indicated to be a problem of CSP systems (Rash, 2009). Neither performance nor security of public cloud systems is managed by internal technology departments of the business firms, a further problem. Though governance might furnish standards in the management of integration and interoperability of non-cloud and cloud systems, the performance of cloud systems and the security of cloud technology, standards are not currently established for the cloud (Korzeniowski, 2009). This immaturity of the cloud is limiting investment in cloud computing systems, except for certain SaaS systems (CIO Insight, 2009).

Though the cloud is in its infancy, business firms, especially large-sized firms, might experiment in cloud computing services if circumstances fit for them. Firms might explore the cloud if information, processes and services are independent of other information, processes and services and if they are new systems; if information, processes and services in the cloud are easy in integrating with non-cloud on-premise information, processes and systems and with cloud systems; if infrastructure is fully functional for non-cloud systems; if platform is Internet with a browser non-native interface to the Web; and if security is not a high requirement (Linthicum, 2010, p. 33). Inevitably firms might explore a hybrid of public and private cloud systems (Babcock, September, 2009), as competitive firms explore the cloud and as CSP technology firms improve maturity of standards and the offerings of the technology. Literature is indicating the cloud to be a "potentially game-changing technology" (Kontzer, August, 2009) for technology departments of business firms. In order to invest in a cloud computing plan, the technology departments of the firms might have to further invest in the skills of its staff (Babcock, November, 2009, p. 1) – skills in cloud computing strategy and in technology. This study introduces a model program of skills that might be integrated into the curricula of schools of computer science and information systems that will be furnishing the future staff of technology departments in the firms.

3. FOCUS

The focus of this study is to define a model program by which educators in computer science and information systems might instruct students in the skills needed in cloud computing strategy and in the technology.

The model program is an enhancement to an earlier model on business process management (BPM), program management methodology and service-oriented architecture (SOA) published by the author (Lawler, Benedict, Howell-Barber and Joseph, 2009), and is founded on this earlier model, inasmuch as SOA facilitates a foundation for cloud computing systems (Krill, 2009). The program is also an enhancement to the IS Curriculum Model, furnishing business, analytical, inter-personal and technical skills. The model program of this study is especially focused on interactions of internal technology departments and business departments of business firms in initiatives of cloud computing. It is focused on potential problem-solving skills relative to performance and reliability, and privacy, risk and security, of infrastructure-as-a-service (IaaS), platform-as-a- service (PaaS) and software-as-a-service (SaaS) public, private and hybrid cloud computing systems. It is further focused on regulatory requirements on the systems. Few publications have focused on an integrated model program for learning skills needed in a cloud computing strategy (Beard, 2009) - often publications have focused on problematic technology (Silverstone, 2010).

This study will benefit instructors in schools of computer science and information systems considering curricula enhancement for cloud computing strategy and technology, and it will benefit indirectly technology departments of firms that will need skilled students as the departments brace for the disruption of their organizations envisioned by cloud pundits (Carr, 2009).

4. METHODOLOGY

From July 2009 to March 2010, the author of this study, who is of the Seidenberg School of Computer Science and Information Systems of Pace University in New York City, conducted a literature survey of practitioner publications (e.g. Computerworld) on experimental projects of business firms relative to cloud computing. The projects were indicated to be infrastructure-as-a-service (IaaS), platform-as-a-service (PaaS) or software-as-a-service (SaaS) public, private or hybrid systems in small and medium-sized firms (i.e. <$500 million in revenue in 2008) and large-sized firms (i.e. >$500 million in revenue in 2008). The features of the projects were indicated to be those of cloud computing systems: fast elasticity for faster resource scalability, location-independent pooling of resources in a multi-tenant or single tenant plan, on-demand resource self-service, pay-as-you-go for resource subscription and / or ubiquitous network access of the resources (National Institute of Standards and Technology, 2009), as was feasible to learn from the publications. The author concurrently conducted a survey of publications of cloud computing provider (CSP) technology firms on recommendations relative to cloud computing, but filtered the findings for hype and bias by including technology-agnostic recommendations relative to the cloud from publications of leading technology consulting organizations (e.g. Gartner Group). All information and recommendations from all of the publications of the survey were further filtered for creditability and feasibility by a colleague of the author who is a technology-agnostic industry practitioner.

The cloud computing projects of the business firms and the technology firms, including the recommendations of the technology firms and the technology consulting organizations, discerned from the practitioner publication survey were evaluated by the author for apparent skills applied, or not applied but needed, by the firms on the systems. The author identified business, analytical, inter-personal and technical skills to the systems from his earlier model on business process management (BPM), program management methodology and service-oriented architecture (SOA) (Lawler, Benedict, Howell-Barber and Joseph, 2009), which was founded on the IS Curriculum Model. He included courses from the earlier model and other courses or modules relative to cloud computing to the applied or needed skills, as to the scope of skills technology departments of business firms might need in computer science and information systems students studying cloud strategy and the technology. He further evaluated the curricula of the Seidenberg School, and other schools of computer science and information systems in the northeast corridor of the country, as to the scope of teaching cloud topics. Most of the other schools have however limited programs in cloud topics.

The model program on cloud computing is presented in the next section of this study.

MODEL PROGRAM FOR CLOUD COMPUTING

The model program for cloud computing proposed for the curricula of computer science and information systems consists of business, culture, methodology, research and technology course modules, whose contents correspond to the domain fundamentals, foundational knowledge and skills, and information specific knowledge and skills of the IS 2009 Curriculum Model. The contents of a number of the modules correspond moreover to the contents of the earlier 2008 service-oriented architecture (SOA) model (Lawler, Benedict, Howell-Barber and Joseph, 2009). The program may begin with a mix of modules for freshman students in year 1 and continue with a further mix of modules for sophomore, junior and senior students in years 2, 3 and 4, dependent on other non-cloud computing modules of the established curricula.

The business modules displayed in Table 1 in the Appendix are essentially focused on business process management (BPM) and cloud computing inter-dependency.

The culture modules in Table 2 are focused on the impact of cloud computing on the culture of business firm organizational staff, including the internal technology department staff.

The methodology modules in Table 3 are focused on cloud computing and service-oriented architecture (SOA), and on frameworks of program management methodology, for managing cloud computing projects with organizations teams.

The research modules in Table 4 furnish industry practices on the projects, as learned from practitioner publications and if feasible from industry project internships.

The technology modules in Table 5 furnish a sampling of technologies, tools and utilities and a sampling of standards that might be applied on pseudo projects by students.

Several of the technologies, tools and utilities might be granted by the technology firms to the schools through partnerships with the universities (National Science Foundation, 2009 and Yahoo!, 2009).

Finally, the program might be enhanced for inclusion of cloud architect, cloud developer, cloud engineer, cloud project manager and cloud strategist career tracks in business client firms, as furnished in Table 6 of the Appendix, inasmuch as the literature of practitioner publications indicates a demand for professionals if not students in the tracks in Table 6.

5. IMPLICATIONS

"Cloud Computing is more of an opportunity than a threat. Ignore an opportunity long enough, it becomes a threat." (Boreel, 2008)

The model program defined for educating on cloud is designed on the foundation that cloud computing is currently a durable initiative. Firms in industry continue evolving on incremental methods and projects on the cloud on the implication that cloud computing is the future, but firms have to begin learning cloud computing skills in order for the cloud to be the future (Erlanger, 2009, p. 3). The model program is formulated on the implication that the curricula of computer science and information systems might be current with experimental if not holistic projects of the firms in the inclusion of cloud computing strategy and technologies, so that students might learn marketable skills in tandem with industry.

The model program is founded on the implication that business process management (BPM) is essential in a cloud initiative. Firms have to include particular process requirements in cloudification initiatives (Vizard, 2009), so that innovation investment is maximized on the cloud (Mitchell, 2009). Literature indicates a movement of the profession from the technical requirements to business process requirements (Erlanger, 2009, p. 2). Technologists have to learn more business skills than technical skills. The model program is formulated on the implication that students might learn more business skills, along with the nuts and bolts the technologies.

The model program is further founded on the implication that governance is important in the management of a cloud computing initiative. Governance in the cloud is not distinct from governance in service-oriented architecture (SOA) except for the increased risk management of cloud projects, services and systems, especially public systems, relative to performance, process and security on the cloud. Governance does the ownership and provisioning of services on cloud and non-cloud external and internal systems (Worthington, 2009). Firms might formalize governance in a program management methodology. The program in the study is formulated on the implication that students might learn program management methodology skills that integrate project management techniques.

The model program is formulated moreover on the implication that service orientation is important in the initiative of a cloud computing strategy. SOA might have furnished a foundation of a platform of "on demand" services for a cloud computing strategy (Krill, 2009) that includes non-cloud and cloud systems. The program is formulated on the implication that students might learn service orientation skills and SOA as a prerequisite to studying cloud computing topics.

Lastly, *the model program for cloud computing is flexibly formulated on the implication that the courses and modules of study might have to be improved for the manner in which firms in industry migrate to the cloud*. Small-sized firms might move into the cloud with their own practices and systems as early as 2010, but large-sized firms might move noticeably into the cloud with their strategies and systems as late as 2011 – 2012 (Smith et.al., 2009, p. 10). Technology firms will move into the field with next-generation technologies that ostensibly suit cloud computing themes (Global Services, 2009). Standards will be new too. The proposal of this study is formulated on the implication that cloud will be a journey, with numerous paths that will require the flexibility of instructors in schools of computer science and information systems that pioneer in programs for improving the cloud computing skills of students, inasmuch as cloud computing is considered now one of the top technologies of 2010 (Currier, 2009).

6. LIMITATIONS AND OPPORTUNITIES IN RESEARCH

This study is constrained by the current immaturity of initiatives in the cloud. Most of the documented projects are software-as-a-service (SaaS) systems, are not public but private systems, and are of small and medium-sized firms not large-sized firms, and if in large-sized firms are not perceived to be strategic systems; and most schools of computer science and information systems do not have a model program for cloud computing strategy and technology. The publication survey of firms in the 2009 – 2010 study might be followed up by a case study of a firm or firms in a new 2011 – 2012 study, once firms further invest in initiatives in cloud computing of higher complexity, as in hybrid or public systems, or in strategic systems. The evolving field of cloud computing is ideal for a planned research study, from which results will be even more helpful to schools of computer science and information systems and to technology departments of firms.

7. CONCLUSION

This study of the cloud can be beneficial to instructors considering enhancement of the curricula of computer science and information systems. The model program defined for cloud computing in the study is founded on a model of business process management (BPM), program management methodology and service-oriented architecture (SOA) that can improve the IS Curriculum Model. Though cloud computing systems and standards are currently in an immature stage, students might learn problem-solving skills relative to performance, process and security that might be eventually helpful to technology departments of firms that will need the skills once cloud computing becomes mainstream in the market. Further planned research on initiatives in the cloud will be helpful in improving the model program of the study. This study furnishes a framework for the further research.

8. REFERENCES

Babcock, C. (2009). Why ‘private cloud’ computing is real – And worth considering. *Information Week*, April 11,1,5.

Babcock, C. (2009). 2 in 3 information technology (it) managers have cloud funding. *Information Week*, August 25, 1-2

Babcock, C. (2009). Hybrid clouds: No easy concoction. *Information Week*, September 5, 1.

Babcock, C. (2009). How cloud computing changes information technology organizations. *Information Week*, November 28, 1-2.

Beard, H. (2009). Cloud Computing Best Practices for Managing and Measuring Processes for On-Demand Computing, Applications and Data Centers in the Cloud with Service Level Agreements (SLAs). Office of Government Commerce, London, United Kingdom.

Akamai and cloud computing: A perspective from the edge of the cloud”, (2009). *Akamai White Paper*, 1.

Boreel, M. (2008). Our collaboration platform Sogeti. *Sogeti Group*, April.

Burt, J. (2009). Enterprises will invest in private clouds first: Gartner. *eWeek*, December 1, 1.

Cameron, B., & Purao, S. (2009). Enterprise integration: An experiential learning model. *Proceedings of the Information Systems Education Conference (ISECON)*, Washington, D.C., November, 1,5.

Carr, N. (2009). The many ways cloud computing will disrupt information technology (it). *Infoworld*, March 26, 3.

Cloud, defined, (2009). *National Institute of Standards and Technology (NIST), Information Technology Laboratory*, Draft Definition, Version 14.

Conry-Murray, A. (2009). From Amazon to IBM, what 12 cloud computing vendors deliver. *Information Week*, September 5, 1.

Crosman, P. (2009). Securing the clouds. *Wall Street and Technology*, December, 23.

Currier, G. (2009). On the rebound: Cloud computing and new mobility technologies gain focus as information technology (it) executives look beyond cost-cutting to new productivity and growth opportunities. *CIO Insight*, December, 29.

Dubey, A., Mohiuddin, J., Rangaswami, MR., & Baijal, A. (2009). Enterprise software survey 2008. *McKinsey and Company and Sandhill Enterprise Software*, February 18.

Erlanger, L. (2009). The tech[nical] jobs that the cloud will eliminate. *Infoworld*, July 22, 2,3.

Fingar, P. (2009). Dot.Cloud: The 21st Century Business Platform. Meghan-Kiffer Press, Tampa, Florida, 27.

Forecast: Cloud computing looms big on the horizon, (2009). *CIO*, CIO Focus, White Paper.

Gage, D. (2009). Big software: Finally dead?. *CIO Insight*, October, 15.

Greengard, S. (2009). Trends for 2010: Piecing together a strategy. *Baseline*, December, 18.

Hamm, S. (2009). Cloud computing: Big bang for business. *Business Week*, June 15, 044.

Hurwitz, J., Bloor, R., Kaufman, M., & Halper, F. (2010). Cloud Computing for Dummies. Wiley Publishing, Indianapolis, Indiana, 15.

Johnson, C. (2009). Hazy prospects for cloud services: Information technology (it) buyers uncertain about benefits, more worried about security. *CIO*, September 1, 28.

Kontzer, T. (2009). The cloud steps up to the plate: Cloud-enabled services emerge as a replay of the internet's rise as a business tool. Anything as a service (xaas) is a potentially game-changing technology that could reshape information technology (it). *CIO Insight*, August, 45,48.

Kontzer, T. (2009). The cloud's sky-high hype: The buzz around cloud computing is alive and well. But chief information officers (cios) need to proceed carefully – and make sure they have the right understanding of what the 'cloud' actually is'. *CIO Trends*, November, 22.

Korzeniowski, P. (2009). Cloud standards are a work in progress. *Information Week*, November 9, HB8.

Korzeniowski, P., & Jander, M. (2009). Cloud security: Business users see cost savings from cloud computing, so they are finding ways to get past this number 1 concern. *Information Week*, November 9, HB5,14.

Krill, P. (2009). SOA gets an obituary. *Infoworld,* January 5, 2.

Lawler, J.P., & Howell-Barber, H. (2008). Service-Oriented Architecture: SOA Strategy, Methodology and Technology. Taylor & Francis Group, Boca Raton, Florida, 27-59.

Lawler, J.P., Raggad, B., & Howell-Barber, H. (2008). Methodology for educating information systems students on the new paradigm of service-oriented architecture (soa) Technology. *Information Systems Education Journal*, 6(52), 3-10.

Lawler, J.P., Benedict, V., Howell-Barber, H., & Joseph, A. (2009). Critical success factors in the planning of a service-oriented architecture strategy for educators and managers. *Information Systems Education Journal*, 7(94), 26-30.

Linthicum, D. (2009). Which cloud platforms deliver reliable service?. *Infoworld*, August 21, 1.

Linthicum, D. (2009). Service governance for the cloud: Ignore at your peril. *Infoworld*, August 31, 1-2.

Linthicum, D.S. (2009). Application design guidelines for cloud computing. *Infoworld*, November 17, 2.

Linthicum, D.S. (2010). Cloud Computing and SOA Convergence in Your Enterprise: A Step-by-Step Guide. Pearson Education, Boston, Massachusetts, 11,33.

McMillan, R. (2009). Researchers find a new way to attack the cloud. *Infoworld*, September 3, 1.

Mitchell, R.L. (2009). Wrapped in complexity: One-off integrations lead to trouble as saas use multiplies. But information technology (it) will get help this time. *Computerworld*, March 23, 23.

Moore, G. (2002). Crossing the Chasm: Marketing and Selling Disruptive Products to Mainstream Customers. HarperCollins Publishers, New York, New York.

National Science Foundation awards millions to fourteen universities for cloud computing research, (2009). *National Science Foundation Press Release*, April 23, 1-6.

Neal, D., Moschella, D., Masterson, M., & O'Shea, D. (2009). Doing business in the cloud – Implications for cost, agility and innovation. *CSC*, White Paper, August, 4,11,40.

Providers look to refine value of their cloud services: Report, (2009). *Global Services*, August 5, 2.

Rash, W. (2009). Is cloud computing secure? Prove it: The cloud can be a secure place for meeting your information technology (it) needs, but proving it is another matter entirely. *eWeek*, September 21, 8.

Reese, G. (2009). Cloud Application Architectures: Building Applications and Infrastructure in the Cloud. O'Reilly Media, Sebastopol, California, 12.

Silverstone, A. (2010). Clearer definition, new metrics for cloud security. *CSO*, January / December, 30-31.

Smith, D.M. et.al. (2009). Hype cycle for cloud computing, 2009. *Gartner Research Note*, July 16, 10, 22.

Topi, H., Valacich, J.S., Kaiser, K., Nunamaker, J., Sipior, J.C., & de Vreede, G. et.al. (2009). IS 2009: Curriculum Guidelines for Undergraduate Degree Programs in Information Systems. Association for Computing Machinery (ACM) and Association for Information Systems, New York, New York and Atlanta, Georgia.

Vizard, M. (2009). How to build private clouds: The challenge is to determine how to recast internal information technology (it) operations into more of a service Model. *Baseline*, May, 16.

Waxer, C. (2009). Can you trust the cloud?: Outages at a cloud computing service could hurt your business. The question is whether your in-house systems can do better. *Computerworld*, May 25 / June 1, 24.

Where does hype end and reality begin with cloud computing? (2009). *CIO Insight*, October 23, 1.

Woodward, B., Sendall, P., & Ceccucci, W. (2009). Integrating soft skill competencies through project-based learning across the information systems curriculum. *Proceedings of the Information Systems Education Conference (ISECON)*, Washington, D.C., 2.

Worthington, D. (2009). Governance in the cloud less than hazy. *SD Times*, May 15, 22.

Yachin, D., & Patterson, R. (2009). Market & analysis overview: Cloud computing. *IDC White Paper*, September, 1.

Yahoo! partners with four top universities to advance cloud computing systems and applications research, (2009). *Yahoo! Finance Press Release*, April 9, 1.

APPENDIX

Figure 1: Adoption of Cloud Computing in 2010

Source: Neal, Moschella, Masterson and O'Shea, 2009 [Adapted from Moore, 2002]

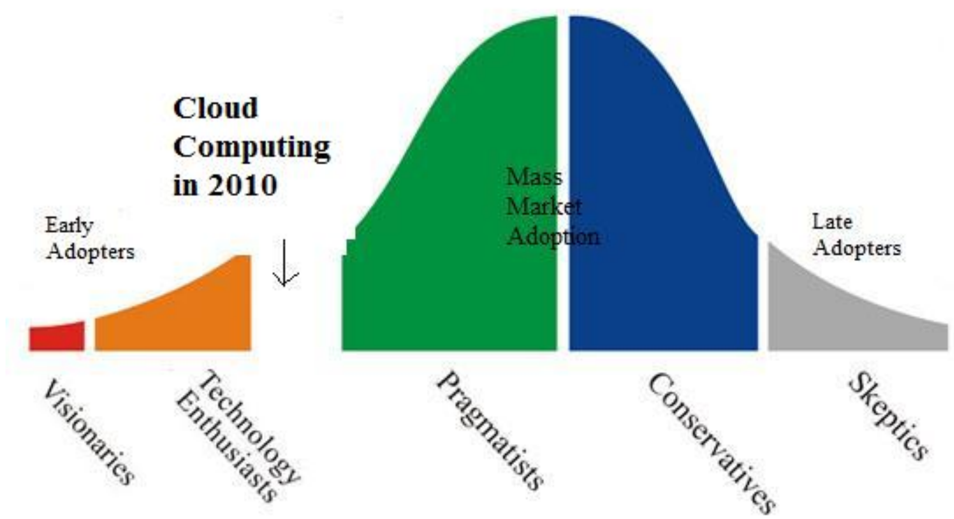


Table 1: Model Program for Cloud Computing – Business Module

Module (Topics)	**Content (Examples)**		**IS 2009 Knowledge Areas ***					**Year**
			General Computing	**Domain**	**Foundational**	**Information Specific**		
Business								
Business Process Management (BPM)	Business Objectives Critical Few Objectives (CFOs) Customer Centricity-Problems in Processes Competitive Differentiation of Core Processes Strategic Performance Management – Process Thinking			x		x		1
Business Process Management (BPM) and Cloud	Candidate Data, Processes and Services for Cloud Computing Cloud Computing Cost Model			x		x		1
Cloud Deployments of Processes	Private Cloud Public Cloud Hybrid Cloud					x		1
Cloud Models	Infrastructure-as-a-Service (IaaS) Platform-as-a-Service (PaaS) Software-as-a-Service (SaaS)					x		1
Industry Regulations	Gramm-Leach-Biley Act (GLBA) Health Insurance Portability and Accountability Act (HIPPA) Statement on Auditing Standard 70 (SAS-70)			x				4

*Topi, Valacich, Kaiser, Nunamaker, Sipior and de Vreede et.al., 2009

Table 2: Model Program for Cloud Computing – Culture Module

Module	**Content**		**IS 2009 Knowledge Areas**					**Year**
			General Computing	**Domain**	**Foundational**	**Information Specific**		
Culture								
Change Management	Changing the Culture of Business Firm Organizations (e.g. Technology Department)				x			1
Organizational Sectors	Corporate Staff Business Staff Governance Staff Technology Staff				x			2
Planning for Cloud	Centers of Excellence in Cloud Computing					x		4

Table 3: Model Program for Cloud Computing – Methodology Module**

Module	Content	IS 2009 Knowledge Areas				Year
		General Computing	Domain	Foundational	Information Specific	
Methodology						
Cloud Computing, Service Orientation and Service-Oriented Architecture (SOA)	Design of Cloud Services Expansion of Cloud Services Governance of Cloud Services				x	1
Program Management Methodology***	**Framework of Governance** **Framework of Communication** **Framework of Product Realization** Analysis and Design Phases Development Phase Integration and Testing Phases Deployment and Implementation Phases Multiple Iterations **Framework of Project Management** **Framework of Architecture** **Framework of Data Management** **Framework of Service Management** **Framework of Human Resource Management** **Framework of Post Implementation**				x	2,3,4
Program Staff Team Playing	Corporate, Business, Governance and Technology Staff of Business Firm and Staff of Cloud Service Provider (CSP) Technology Firm			x	x	3

** Lawler, Benedict, Howell-Barber and Joseph 2009
*** Lawler and Howell-Barber, 2008 and Lawler, Raggad and Howell-Barber, 2008

Table 4: Model Program for Cloud Computing – Research Module

Module	Content		IS 2009 Knowledge Areas					Year
			General Computing	Domain	Foundational	Information Specific		
Research								
Independent Project Study of Cloud Systems			x	x	x	x		3,4
Best-of-Class Practices in Industry	IaaS, PaaS and SaaS Systems			x		x		3,4
Practitioner and Scholarly Publications				x		x		3,4
Instructor as Study Supervisor			x	x	x	x		3,4
Industry Project Internships ****	Experiential Learning Projects		x	x	x	x		4

**** Cameron and Purao, 2009

Table 5: Model Program for Cloud Computing – Technology Module

Module	Content	IS 2009 Knowledge Areas				Year
		General Computing	Domain	Foundational	Information Specific	
Technology						
Cloud Computing as Design Patterns					x	1
Infrastructure	**CPU** Network -Servers Storage **Platforms** **Services**	x				2,3
Cloud Computing and SOA	SOA and Service-Oriented Enterprise (SOE)				x	1
Cloud Computing Information Model	Clustering vs. Replication Metadata Privacy				x	2
Cloud Computing Infrastructure of Services	Grid Computing Transactional Computing	x			x	2.3
Languages	AJAX Force.Com APEX Google GQL Java Microsoft C# Microsoft Office Web Apps	x				2,3,4
Platforms of Cloud Technology Firms		x				2,3,4
Product Specific Cloud Technologies	Amazon Web Services Google Docs SalesForce.Com	x				3,4
Technology Process Management				x	x	4

Risk Management and Security of Cloud Systems	**Cloud Computing Security Strategy** Data Security Host Security Network Security **Cloud Computing Security Techniques** Detection and Forensics Encryption Identity Management **Disaster Recovery Planning**				x	x		3,4
Standards on Cloud	Cloud Camp Cloud Computing Interoperability Forum Cloud Computing Use Cases Group Cloud Security Alliance Distributed Management Task Force Object Management Group Open Cloud Manifesto Open Group Service Integration Maturity Model (OSIMM) Open Group SOA Work Group SOA Governance Framework (Sample)		x			x		3
Systems Management of Cloud	**Capacity Planning** Expected Demand Impact of Load					x		4
Cloud Scaling	Dynamic Scaling Proactive Scaling Reactive Scaling					x		4
Monitoring of Systems			x			x		3,4
Cloud Computing "Bill of Rights"	Business Firms (Data) Technology Firms (Interfaces) Business Firms and Technology Firms (Service Levels) **Contracts for Business Firms** Lock-In vs. Portability **Service Levels for Cloud Systems** Availability Performance Security			x		x		4
Utilities	Product-Specific Utilities		x					3,4
Trends	Careers for Cloud Computing Practitioners -Compensation and Employment Forecast for Practitioners Impact of "Everything as a Service on Cloud" on Information Technology Departments of Business Firms			x		x		4

Table 6: Model Program for Cloud Computing – Modules with Career Tracks

Module (Topics)	Content (Examples)	Skills				Career Tracks					Year
		Business Skills	Analytical Skills	Interpersonal Skills	Technical Skills	Cloud Architect	Cloud Developer	Cloud Engineer	Cloud Manager	Cloud Strategist	
Business											
Business Process Management (BPM)	Business Objectives Critical Few Objectives (CFOs) Customer Centricity-Problems in Processes Competitive Differentiation of Core Processes Strategic Performance Management – Process Thinking	x	x							x	1
Business Process Management (BPM) and Cloud	Candidate Data, Processes and Services for Cloud Computing Cloud Computing Cost Model	x	x			X				x	1
Cloud Deployments of Processes	Private Cloud Public Cloud Hybrid Cloud	x	x			X				x	1
Cloud Models	Infrastructure-as-a-Service (IaaS) Platform-as-a-Service (PaaS) Software-as-a-Service (SaaS)	x	x		x	X				x	1
Industry Regulations	Gramm-Leach-Biley Act (GLBA) Health Insurance Portability and Accountability Act (HIPPA) Statement on Auditing Standard 70 (SAS-70)	x								x	4

Module	Content	Skills				Career Tracks					Year
		Business Skills	Analytical Skills	Interpersonal Skills	Technical Skills	Cloud Architect	Cloud Developer	Cloud Engineer	Cloud Manager	Cloud Strategist	
Culture											
Change Management	Changing the Culture of Business Firm Organizations (e.g. Technology Department)		x	x		x			x	x	1
Organizational Sectors	Corporate Staff Business Staff Governance Staff Technology Staff	x	x	x					x		2
Planning for Cloud	Centers of Excellence in Cloud Computing	x	x	x	x	x			x	x	4

Module	Content	Skills				Career Tracks					Year
		Business Skills	Analytical Skills	Interpersonal Skills	Technical Skills	Cloud Architect	Cloud Developer	Cloud Engineer	Cloud Manager	Cloud Strategist	
Methodology											
Cloud Computing, Service Orientation and Service-Oriented Architecture (SOA)	Design of Cloud Services Expansion of Cloud Services Governance of Cloud Services		x			x	x		x		1

Program Management Methodology*	**Framework of Governance** **Framework of Communication** **Framework of Product Realization** Analysis and Design Phases Development Phase Integration and Testing Phases Deployment and Implementation Phases Multiple Iterations **Framework of Project Management** **Framework of Architecture** **Framework of Data Management** **Framework of Service Management** **Framework of Human Resource Management** **Framework of Post Implementation**		x	x	X	x		x	x	x	x	x		2,3,4
Program Staff Team Playing	Corporate, Business, Governance and Technology Staff of Business Firm and Staff of Cloud Service Provider (CSP) Technology Firm				X			x	x	x	x			3

Module	Content	Skills				Career Tracks					Year
		Business Skills	Analytical Skills	Interpersonal Skills	Technical Skills	Cloud Architect	Cloud Developer	Cloud Engineer	Cloud Manager	Cloud Strategist	
Research											
Independent Project Study of Cloud Systems		x	x	x	x	x	x	x	x	x	3,4
Best-of-Class Practices in Industry	IaaS, PaaS and SaaS Systems	x	x		x	x	x	x	x	x	3,4
Practitioner and Scholarly Publications		x	x		x	x	x	x	x	x	3,4
Instructor as Study Supervisor		x	x	x	x	x	x	x	x	x	3,4
Industry Project Interships****	Experiential Learning Projects	x	x	x	x	x	x	x	x	x	4

Module	Content	Skills				Career Tracks					Year
		Business Skills	Analytical Skills	Interpersonal Skills	Technical Skills	Cloud Architect	Cloud Developer	Cloud Engineer	Cloud Manager	Cloud Strategist	
Technology											
Cloud Computing as Design Patterns			x		x	x	x		x		1
Infrastructure	**CPU** Network -Servers Storage **Platforms** **Services**		x		x		x	x			2,3
Cloud Computing and SOA	SOA and Service-Oriented Enterprise (SOE)	x	x		x	x			x	x	1

Cloud Computing Information Model	Clustering vs. Replication Metadata Privacy		x	x		x		x	x		x			2
Cloud Computing Infrastructure of Services	Grid Computing Transactional Computing			x		x				x				2,3
Languages	AJAX Force.Com APEX Google GQL Java Microsoft C# Microsoft Office Web Apps			x		x			x					2,3,4
Platforms of Cloud Technology Firms				x		x			x	x	x			2,3,4
Product Specific Cloud Technologies	Amazon Web Services Google Docs SalesForce.Com			x		x			x	x	x			3,4
Technology Process Management			x	x	x	x					x	x		4
Risk Management and Security of Cloud Systems	**Cloud Computing Security Strategy** Data Security Host Security Network Security **Cloud Computing Security Techniques** Detection and Forensics Encryption Identity Management **Disaster Recovery Planning**			x		x		x	x	x	x	x		3,4
Standards on Cloud	Cloud Camp Cloud Computing Interoperability Forum Cloud Computing Use Cases Group Cloud Security Alliance Distributed Management Task Force Object Management Group					x		x	x	x	x	x		3

	Open Cloud Manifesto Open Group Service Integration Maturity Model (OSIMM) Open Group SOA Work Group SOA Governance Framework (Sample)													
Systems Management of Cloud	**Capacity Planning** Expected Demand Impact of Load		x	x		x				x				4
Cloud Scaling	Dynamic Scaling Proactive Scaling Reactive Scaling			x		x				x				4
Monitoring of Systems				x		x		x		x		x		3,4
Cloud Computing "Bill of Rights"	Business Firms (Data) Technology Firms (Interfaces) Business Firms and Technology Firms (Service Levels) **Contracts for Business Firms** Lock-In vs. Portability **Service Levels for Cloud Systems** Availability Performance Security		x	x	x	x					x	x		4
Utilities	Product-Specific Utilities			x		x			x					3,4
Trends	Careers for Cloud Computing Practitioners -Compensation and Employment Forecast for Practitioners Impact of "Everything as a Service on Cloud" on Information Technology Departments of Business Firms		x	x	x	x		x	x	x	x	x		4

Note: Cloud manager is cloud project manager.

IS/IT Education vs. Business Education: The Plight of Social Collapse in Technical Business Environments

Brian Hall
hall@champlain.edu
Champlain College
Burlington, VT 05401, USA

Abstract

In an age when information, management and technology are supposed to be hand-in-hand, there is often a rift between these elements when considering people and comportment. The detachment is caused by a distressing lack of understanding between Information Systems (IS)/Information Technology (IT) students and professionals and those they interact with in the business world. Eventually, this deficiency manifests itself in various ways including a collapse in communication and interaction. This paper is a discussion and sample case of a major oversight in curricula, of preparing students socially for immersion in technical business environments. The omission of cultural literacy on both sides of the equation, in IS/IT programs and in business and management programs (be they technically focused or not), is argued as the underlying cause of many problems in information professions and a source of management contention.

Keywords: technical business environments, technology culture, cultural literacy, social skills, information programs, curricula, success

1. INTRODUCTION

In the information age there is a realization people must come to: if you work, you will interact with technology and the people responsible for it. Information technologies permeate all hierarchical levels of organizations and therefore involvement is expected from a wide range of workers and groups (Mraovic, 2003). Business managers will be involved in the implementation of information systems (IS). Information technology (IT) managers and their staff will be meeting with managers and end-users when systems are analyzed and designed. With this guaranteed social interaction ahead for students in information and business programs, suitable preparation should be a principle educational concern.

It is the responsibility of any educational institution to properly prepare their students for the environments they will eventually inhabit. In IS and IT education the majority of time is spent on technical detail, while in business education it is most spent on business fundamentals. Though this is to be expected, there is a weakness on both sides, of social preparation for the information arena. The result is that the players involved in projects and on teams neither appreciate nor understand the social frameworks and culture of these environments and how to behave within their borders. Additionally, business and technology managers often struggle to appropriately communicate with their staff and coworkers. With relationships and interaction strained, social collapse and project failure are probable.

There is an eminent need for IS/IT and business curricula to include required core courses discussing social culture and communication in technical business environments (TBEs). This is logical considering the fact that most industries, businesses and environments are bound to IS. Workforce trends show skills related to working with external parties are increasing in importance and that is what employers are searching for in IT professionals (Abraham, Beath, Bullen, Gallagher, Goles, Kaiser, & Simon, 2006).

The lack of supportive courses to meet this call is a deceivingly costly oversight, which is often attributed to related but non-foundational reasons in the professional realm. Ergo, without acknowledging the fundamental problems in IT departments and with IS projects, and without pinpointing their origin, they will persist indefinitely into the future. Giving students the keys and educating them on best practices of communicating and interacting in these situations and environments will go a long way in improving project success and inner/inter-departmental relations in their professions.

For the purpose of this research, a review is given of the ongoing struggle in IS/IT management. Then, a summary is given of the current trend in IS/IT and business education with respect to how much effort is being spent on social preparation. Next, results of a sampling case are presented. The methodology was to randomly sample current curricula in various information and business disciplines. Using course descriptions and content analysis, the classes were aggregated and rated according to their social preparation characteristics and prescription. The resulting data can then be applied against commonly cited symptoms of project failure and departmental problems to expose a foundational cause, which if acted upon harbors potential benefits for information systems and technology as a profession. Ultimately, this should validate the social imperative and provoke discussion of how these educational programs can be improved to include this missing core component.

2. BACKGROUND

IT department struggles and IS project failure have been some of the most talked about topics in this facet of academia for decades and with good reason. They are a persistent plight. There are conflicting views of the reliability of the Standish Group studies on project rates (Standish Group International, 1994-2009), but no matter ones' conclusions of Chaos or Bull reports, the fact is information system implementation is a tall order and complete success is rare. There are academic concentrations in Project Management, expensive software packages to help manage projects so failure is "less" likely and consulting firms to help "improve" the rate of success. Yet, anyone who has worked in these TBEs can aver that the situation has not improved significantly, and is ironically chaotic from year to year. Additionally, one can hardly examine any tech news outlet without coming across columns on management issues related to IT, both with people and technology.

When reasons for such a plight are given, many are admittedly symptomatic and not causal (IT Cortex, n.d.). Also, opinions run rampant through firms, publications and blogs as to the causes of and remedies for the sickness of project failure and departmental problems. Be they realistic or not, there are arguments for and against every statement. A wealth of studies, discussion and reasoning later, the difficulty remains. And it will remain until the foundational constructs of these environments are unearthed and examined. The debate should not be solely one of finance, time, communication, quality control, resource planning or management. These are symptomatic layers built on a weak base.

Deeper investigation reveals the difficulty extends from departmental and managerial business relationships and behavior. This is often discussed in a broader context about social capital in organizations. Peppard (2007, p.341) described it this way:

> "The central proposition of the theory around social capital is that this network of relationships constitutes a valuable resource for the conduct of social affairs in an organization. Crucially, social capital operates outside of formal organizational structures. However, how we structure organizations can impede the development of social capital; it may encourage fragmentation rather than integration. For example, IT specialists tend to have their own language and codes of practice. Often, little trust exists between IT specialists and employees from within the business. Indeed, it has been suggested that there can be a cultural difference between employees from the IT function and those from the rest of the business (Ward & Peppard, 1996)."

Here are good questions to ask current information and business students or even experienced professionals. Have you ever come across useful information about: The types of people you'll be working with as an aspiring technology or business operative? The managerial tendencies of those above or around you? The expectations of being an IT worker? Perceptions of team members throughout an IS implementation? Interactions between IT departments and business personnel? Communication practices in a TBE? Chances are probably not or minimal exposure, yet these disciplines are about INFORMATION.

The upshot is that students go into the workforce and collide with an unexpected force no matter their position in the chain. For example, the lower tiers may not know how to communicate effectively with those above them or achieve success in terms of business (Brockway & Hurley, 1998). Those in managerial and higher tier positions may not know how to respect, communicate with, or influence and manage technical teams and projects. These environments have developed into societies with no social compendium hence the plight deepens.

This leads closer to the base issue, which is cultural illiteracy. One side does not understand business culture and social frameworks. The other side does not understand technical culture and social frameworks. They are then expected to interact and communicate to achieve success, in spite of the fact that they do not understand the realms they are operating within and between. National Education Association (NEA, Retrieved 2010) research states that the first key to best practices of teaching and learning in education is cultural competency. The same can be argued for managing and interfacing in business. If we know or do this for the classroom's sake, why not for the department's sake or the project's sake. Motivation and attention are paramount in the educational sphere to direct students toward learning the culture and social constructs of these technical business environments, and how to behave and communicate within them.

3. THE TREND

Of the several observations that should be recounted before digging into the curricula and statistics, the first is this elementary question of literacy. Thomas and Blackwood (2009) argue that exposing non-IS majors to computer literacy courses has potential to improve students' perception of technology in business. It is also fact that most technology programs expose their students to business literacy usually by mandating or offering some fundamental business courses. If the need for this operating level literacy and training is perceived, the need for cultural and environmental literacy and training should not be ignored, though it currently is.

One way this lack of attention reveals itself is talked about as a crisis in computing, the much-discussed decline in student interest in IS/IT programs over the past decade. Though things may be turning around, some of the reasons given for the decline: "fear of becoming isolated in jobs perceived to involve little human contact, little public understanding of the broader dimensions of the computing field, doubts about the relevance of computing particularly as it is taught, lack of excitement and currency in the undergraduate curriculum", are interesting in this context (McGettrick, Cassel, Guzdial, & Roberts, 2007, p.330).

Students obviously need to be made aware that certain cultural and social skills are needed, but faculties need to ensure educational investment. Beard, Schwieger, and Surendran (2010) pointed out that many studies show this need and remarked that Management Information Systems (MIS) students may have an edge because their programs have additional business offerings in "soft skill set" areas. But even if certain programs may have an edge, they also acknowledged, "acquiring soft skills remains somewhat elusive" (Beard, et al., 2010, p.9). Also, just as needed courses remain elusive, so do fundamental concepts within the courses actually being offered. Students are requested to learn these concepts and skill sets in a small percentage of their coursework, and they are not given the context or tools with which to make application.

Students then have the difficulty of crossing into professions. Again, Beard, et al. (2010, p.5) stated that academicians from IT disciplines should be working with those from other disciplines to "ensure graduates possess not only technical skills and knowledge but also business knowledge and soft skills". The author agrees with this sentiment, but taking it one step further asks where this preparatory request is translated into curricula. In the Association for Information Systems Wiki ("Commentary," n.d.) a question was posed that asked, "What do employers (and potentially other stakeholders) want?" A Midwestern faculty member replied: "The ideal candidate is one that has it all - excellent communication, leadership, and social skills, and at the same time a geek." Again, this "soft skill set" is something that is clearly desired and sometimes requested as a primary need, expected by employers (Overby, 2006), but in reality most effort is spent on "geek" and very little on communication, social skills and cultural literacy. This makes the exchange from student to professional increasingly difficult.

Another observation can be made based on the IS curriculum model (http://blogsandwikis.bentley.edu/iscurriculum/index.php). Whether looking at the 2002 or 2010 model, social frameworks and culture of TBEs are not included specifically, but there are some related topics within the foundational knowledge and skills section and sprinkled elsewhere. Though there is plentiful discussion of leadership and collaboration, communication and negotiation, one must ask how a person can effectively do those things without cultural understanding and social competency. Additionally, no matter the model a university may be using to build its curricula, the training for these skills is commonly distilled into generalized courses with titles such as Organizational Behavior, Management and Organizations, Project Management, Leadership Communication, Global Business Environments, Business Communication and IT Management. The trend to generalize and minimize cultural knowledge and social skills in IS/IT and business educational programs is ongoing and so are the professional consequences.

4. SAMPLE CASE

Methodology

In order to determine the amount of coursework in IS/IT and business programs geared towards cultural literacy and social preparation for TBEs, a sampling study was done. Degree curricula were sampled from 19 universities and 38 programs in Information Systems, Management Information Systems, Information Technology, Computer Information Systems, Computer Information Technology, Information Systems Management, Information Science, Technology Management and Business Administration (some including concentrations in IT), totaling just shy of 1200 courses.

Sample Profile:

- 15 public and 4 private institutions
- By region of accreditation there are 6 from the Southern region, 6 North Central, 3 Middle States, 2 Northwest, 1 West, and 1 New England
- 18 are accredited by the AACSB and 1 by the ACBSP
- Size of student enrollment and programs varied
- 25 Bachelor and 13 Masters programs
- All programs available on-campus, 4 available online

Programs were then filtered.

Criteria

1. Are there any courses listed related to social culture and communication in technology environments?

2. How many? Out of?

3. Rate each course as Generic, Inclusive, Specific or Potential using Content Analysis.

Generic - provides some general concepts applicable in a variety of situations, not technology specific.

Inclusive - either includes some related discussion of social frameworks and relational behavior in technology or has potential for inclusion of more specific topics. Typically broader based but does include some related material. Range may vary.

Specific - course is specifically about social and cultural aspects of IS/IT business environments.

Potential - open/topical course that has the potential to include such topics.

4. Are the courses given by the home department or an external department?

Considerations

This study was done with the understanding that much deeper research would be needed to get a complete framework for reference. This would include a thorough review of all the syllabi, a task that time did not permit for this particular project. However, the author would still caution that going by class description or even syllabi does not necessarily guarantee what is discussed in each particular class.

Course totals were based on courses listed as available or required by the program using information provided publicly. This did not always include the total coursework required for completion of the degree or what courses may have been available outside of the department.

General education and elective courses were not always included in the totals because they were not part of the required or listed curriculum and therefore no guarantee exists as to which exact classes are taken. They have been included when possible and if listed in some manner with the specific degree information.

In some cases, Project and Technology Management courses were not tallied as related if their descriptions did not indicate any social or cultural inclusion but focus entirely on very broad or technical aspects of the topics. In the same respect, some Business Communications and Organizational Behavior courses were not related since based on their descriptions, the material was too unfocused.

Table 1 - Total Coursework Related

	IS/IT	Business
Related	5.0%	5.0%
& Required	3.1%	3.3%
& Elective	1.9%	1.7%
& Generic	2.9%	3.9%
& Inclusive	1.5%	0.8%
& Specific	0.1%	0.0%
& Potential	0.4%	0.4%
& Internal	3.1%	4.3%
& External	1.9%	0.8%

Table 2-Characteristics of Related Courses

	IS/IT	Business
Required	61.8%	65.4%
Elective	38.2%	34.6%
Generic	58.8%	76.9%
Inclusive	29.4%	15.4%
Specific	2.9%	0.0%
Potential	8.8%	7.7%
Internal	61.8%	84.6%
External	38.2%	15.4%

Table 3 - IS/IT Total Coursework Related

	Undergrad	Grad
Related	4.5%	7.8%
& Required	2.4%	6.8%
& Elective	2.1%	1.0%
& Generic	3.0%	2.9%
& Inclusive	1.0%	3.9%
& Specific	0.0%	1.0%
& Potential	0.5%	0.0%
& Internal	2.6%	5.8%
& External	1.9%	1.9%

In all, a delicate effort was made to list all courses that would be potentially related in any way to culture, social studies, communication or behavior in TBEs. When exact details were not provided or unclear, the courses were given the benefit of the doubt and included.

Table 4 - Course Averages

	IS/IT	Business
Related	1.48	1.73
Out of *	30.82	34.33
Percentage	4.8%	5.0%
IS/IT	**Undergrad**	**Grad**
Related	1.59	1.33
Out of *	35.94	17.17
Percentage	4.4%	7.8%
* Does not always include entirety of coursework required to complete degree		

Points of Interest

- One of the more interesting outcomes was that the random sampling showed 5% of courses as related in both IS/IT and business programs.
- There are surprising similarities when comparing IS/IT and business programs. This is largely due to the fact most related classes are shared.
- About two-thirds of the classes are required, one-third are elective.
- The majority of classes are generic.
- Only 1 course of the almost 1200 sampled was specifically related to these topics, and it was a graduate course.
- Inclusive courses are mainly at the discretion of the instructors and though perhaps given the benefit here, many are borderline generic.
- Current courses harbor little potential due to their out-dated structure.
- Though many related courses are offered internally, external departments give a sizable number, making them less likely to instill a suitable skill set.
- In IS/IT programs, graduate students have a higher probability of being exposed to these topics, whereas undergrad chances diminish. Undergrads have nearly half the related

courses, nearly three times less required courses, most are generic, nearly four times less inclusive, none are specific, and they are split between internal and external.

- On average, programs have about 1.5 courses related to social and cultural frameworks.

- If all coursework for a degree was included, related percentages drop 2% on average assuming an average 130 credit-hours/undergrad and 40/grad. For example, assuming a Bachelor IS degree of 130 credit-hours, related courses would account for 3.4% as opposed to 5%, and lowers all other percentages. However, for this study only listed curricula was included.

5. THE FACTS

To better understand the connection between curricula and professional environments, the common hardships of the environments should be represented along with the symptoms. It is commonly accepted that IT departments are highly subject to turmoil and IS projects to failure. Almost every report and survey says the same thing about what is wrong with IT environments and what is needed to fix them. Jeff Ello (2009a) wrote that almost every source on the subject can be summarized in a couple sentences, which includes the belief that though smart and creative, IT pros are "antisocial, managerially and business-challenged", among other things. Along with the author, Ello also believes such stereotypes stem from a lack of understanding of the people and the culture, and that if one does understand, it makes working with them a much easier job.

Organizations sometimes attempt to build up relationships between IT and the rest of the business by using relationship managers and other in-between positions; though they typically make things more convoluted or are minimally effective. Some organizations offer internal training to improve knowledge of the business or IT. But, if collective competencies and coordinated knowledge are to be developed, then they must overcome the requirement for business and IT people to work together (Peppard, 2007). Also, what might be a good plan or idea for social integration can quickly turn into social irritation if respectful and competent communication is not proffered. Organizations put a lot of effort and time into increasing communication, relieving anxiety, boosting visibility and deterring pessimism in relation to technology departments (Brandel, 2010). Social irritants along with company effort and time investments can be reduced with pre-profession educational skill building, making it less irritating for the industry itself.

It is true that many technical workers may be socially unskilled and the same can be said for business associates, especially when dealing with technical departments. The good news is that skills can be taught, but what makes it hard is preconceived stereotypes on both sides and differing definitions of competence. Everyone in these situations acts and reacts to these perceptions (Ello, 2009b). Business personnel and IT pros alike need to be reminded of the social particulars of the opposing group if success is to be achieved.

However, the common ending is much like the popular Nut Island Effect detailed by Levy in the *Harvard Business Review* (2001). JoAnn Hackos (2004) commented on the five stages of the effect and their application to information development teams. Her assessment, along with many others, is that there lacks some form of understanding between senior management and team workers, which leads to an utter breakdown in communication. She also states in closing that "we need to get out of the office ourselves to learn directly from team members and to interact with colleagues in our own field and associated fields" and by doing this "we are continuously exposed to new thinking, decreasing our isolation, and providing us with challenges" (Hackos, 2004). The author believes this should not only apply to managers and workers, but technical and non-technical coworkers and teams. Appropriate training and interaction exercises throughout educational programs, focused on cultural understanding and behavior in technical environments, would help with professional de-isolation.

One could use project surveys to argue statistical facts of failures. Like, the 2009 Standish Group study claims approximately 68% of projects fail or are challenged and 63% over-run, the Robbins-Gioia Survey (2001) claims 51% failure in ERP implementations, *The State of IT Project Management* (Huber, 2003) claims 59% are over budget and only 16% hit all targets, and BCS research stating only one in eight are truly successful (McManus, J., Wood-Harper, T., & BCS 2008). But departing from that approach, it will just be accepted that there are problems. The symptoms (sometimes listed as causes in surveys) and risks of problems, improvement factors and characteristics of success have been extrapolated from these surveys and concisely listed.

Symptoms and Risks

- Stereotypes
- Differing definitions
- Isolation
- Lack of direction
- Lack of continuity
- Bad communication between relevant parties (over 50%)
- Lack of planning of scheduling, resources and activities
- Inadequate co-ordination
- Mismanagement of progress
- Overall poor management, business specific and technology specific
- Lack of attention to the human and organizational aspects of IT
- Poor articulation of requirements
- Inadequate attention to business needs and goals
- Management commitment
- Lack of client/user involvement
- Inadequate project management
- Failure to manage expectations
- Conflict among stakeholders
- Shortage of knowledge/skills in the project team
- Improper definition of roles and responsibilities
- Staff turnover
- Unrealistic expectations
- Technology illiteracy
- Poor delegation

Improvement Factors

- Greater management support
- Commitment from users
- Greater control over resources
- More project management training
- Stable project management methods
- Greater understanding of PM on the part of top management, teams, and clients
- Ability to adapt

Characteristics of Success

- Leadership
- Integrity
- Understanding of IT
- Written communication
- Problem solving
- Understanding business processes and strategy
- Ability to manage change
- Well qualified in project management techniques
- Communicates goals
- Attention to detail and high-level issues

6. DISCUSSION POINTS

Departments and Projects

The survey lists presented in *The Facts* section do not spell out causes of problems in the professional world of IS/IT. They are symptoms and relief methods, which can all be linked to a common fundamental origin. Professional IS/IT and business relationships and projects regularly break down because IS/IT and business students are not provided with cultural knowledge, communicative practices, skills and training needed for a technical business environment. Granger, Dick, Jacobson, and Slyke (2007, p.304) state it is "possible that many IS curricula do not reflect the evolving demands of today's (and tomorrow's) IS professionals" when arguing fundamental causes of a decrease in IS-related enrollments. It may also be a fundamental cause of this instability in IT environments. Cultures and social constructs evolve as demands do, and if those nuances are not accounted for in curricula then understanding and preparation are lost. The result is that technical departments and projects suffer and succumb to the plight.

Even though IT has changed drastically over the last decade, the requirements for IS/IT success have not changed as many people may think. As Brockway and Hurley (1998, p.203) said years ago:

> "success...requires high degrees of interaction among IT and business people discussing business direction and the information systems required to support it. In order to participate, the IT organization needs to have staff that understand both the capabilities of systems in general and the essence of the business they work in and can hold their own in complex and sensitive discussions about the interactions of the two. IT needs staff that know and understand the business, have a point of view about the future, and apply this knowledge to engender support and confidence across the business. Business functions need a similar set of skills. The two groups need to understand and respect each other."

Curricula

As courses and descriptions were analyzed there were some notable details for discussion:

- Generic project management and other generic courses are not designed to expose social and cultural frameworks of TBEs, even if they are "for information technology".

- Programs may contain one or two courses that are related but don't emphasize social nuances in technical environments.

- Based on their current descriptions, some courses have the potential to incorporate these topics if only updated and structured to include them.

- If there are courses available that supply some of these aspects, they are likely hidden among the electives and among the least taken.

- Some entire computer and information science, information systems, information technology catalogs do not posses one class related to cultural and social frameworks.

- Generally, students take at least one course in each subject they won't use as often or at least not daily, like calculus, economics, statistics, history, in a beneficial effort to be well rounded. Yet, many are not offering any courses on the skills and knowledge they need to possess and use every day while interacting in a technical environment.

- Based on analysis, little or no time is spent discussing the society at the bottom of the IT chain but only upward towards managers, executives, or the global community. Environments are only seen on a macro level, yet the coders, engineers, administrators, analysts and business personnel need to be understood as well.

- There exists practically every type of management or strategy focused course on human resources, organization, business, ethics, international business, strategy, IS, project, finance, risk, supply chains. Yet, not one focused on technical personnel or technical environment management.

- Even if there are one or two courses that touch on this subject in the average curriculum, is that enough? Or the right type?

- Of the programs sampled, 6 required internships, 4 encouraged them, 2 were programs for working professionals, equaling about 30% exposure at best. Though internships were not the focus of this study, these professional preparatory experiences can greatly enhance a student's understanding of social and cultural elements, and programs should require or at least encourage them as part of their curricula.

Exposure and Communication

This responsibility to broaden the social skills and cultural understanding of both technical and business students is a necessity but has yet to garner much attention. Previous conclusions have been similar to the findings of this study, that on average only a couple of hours out of an entire undergraduate degree are spent on highly sought after soft skills (Russell, Russell, & Tastle, 2005). They, along with other writers and faculty have issued calls for curricula enhancements, but not much has changed.

If courses in IS/IT programs are expanded to include more aspects of the humanities, the exposure to these topics may help increase enrollments just as other types of disciplinary exposure have (Granger, et al., 2007, p.306). They go on to say that seemingly many courses can be boring to students, especially early on, and that perhaps the diversity of curricula should be examined and modified to "provide an interesting and stimulating student experience." This particular subject matter should be one such modification. It would provide fundamental understanding, stimulate thought and properly prepare students interactively.

Right now the main source of information on these topics is sporadic and through technical news outlets (e.g. tech magazines and online publications such as *Computerworld* and *CIO*) geared towards professionals already in the workforce, not students in the classroom. This is like trying to vaccinate someone already infected. The best way to bring success and change to these environments is to send the students out already prepared for the professional expectation. Curricula modifications must be made and the need for a higher quality of cultural and social preparation, along with the right tools, must be communicated to today's students who are tomorrow's professionals.

7. CONCLUSIONS

Just as organizational culture has an impact on IT (Zhao, 2004) and vice versa, cultural competency and business impact each other. One should never forget the human side of organizational agility in businesses (Crocitto & Youssef, 2003) nor the learning of it in related educational programs. The success of an IT department, IS implementation, or IS/IT and business programs can be determined by how well the cultures are understood and level of respect in social interplay. This means educational institutions with various IS/IT and modern business programs should develop and instill courses into their core requirements focused on cultural literacy and social adeptness in technical business environments.

It is true that one cannot imagine and discuss every possible social situation or cultural setting, but one can properly prepare and act by developing, learning and applying best practices of understanding and interaction within these environments. However, with little or no effort being applied in this area, as evidenced by this sample case and other investigations, managerial, departmental and project complications will remain unless there is rudimentary change. Simply put, if improvement is desired in technical business professions, then students of these programs must be sent into the workforce prepared for cultural and social immersion.

8. REFERENCES

Abraham, T., Beath, C., Bullen, C., Gallagher, K., Goles, T., Kaiser, K., and Simon, J. (2006). IT Workforce Trends: Implications for IS Programs. *Communications of the AIS*, 17, 2-43.

Beard, D., Schwieger, D., and Surendran, K. (2010). A Value Chain Approach for Attracting, Educating, and Transitioning Students to the IT Profession. *Information Systems Education Journal*, 8(7). http://isedj.org/8/7/. ISSN: 1545-679X. (A preliminary version appears in The Proceedings of ISECON 2009: §3312. ISSN: 1542-7382.)

Brandel, M. (2010). Salaries stall, workloads rise, and IT gets squeezed. *Computerworld*, Retrieved April 2010 from http://www.computerworld.com/s/article/347538/The_Big_Squeeze

Brockway, D., & Hurley, M. (1998) Achieving IT success. *Information Management & Computer Security*, 6(5), 199-204.

Commentary on the Three Questions. (n.d.). Retrieved May 2010 from http://enrollments.aisnet.org/Commentary%20on%20the%20Three%20Questions.ashx

Crocitto, M., & Youssef, M. (2003). The human side of organizational agility. *Industrial Management & Data Systems*, 103(6), 388-397.

Ello, J. (2009a). Opinion: The unspoken truth about managing geeks. *Computerworld*, Retrieved April 2010 from http://www.computerworld.com/s/article/9137708/Opinion_The_unspoken_truth_about_managing_geeks

Ello, J. (2009b). Opinion: The unspoken truth about why your IT sucks. *Computerworld*, Retrieved April 2010 from http://www.computerworld.com/s/article/9141609/Opinion_The_unspoken_truth_about_why_your_IT_sucks

Gorgone, J., Davis, G., Valacich, J.S., Topi, H., Feinstein, D., and Longenecker, H.E. (2003). IS 2002 Model Curriculum and Guidelines for Undergraduate Degree Programs in Information Systems. *The DATA BASE for Advances in Information Systems* (34:1).

Granger, M., Dick, G., Jacobson, C., Slyke, C. (2007). Information Systems Enrollments: Challenges and Strategies. *Information Systems Education Journal*, 18(3), 303-311.

Hackos, J. (2004). The Nut Island Effect. *Center for Information Development Management e-newsletter*, Retrieved March 2010 from http://www.infomanagementcenter.com/enewsletter/200409/feature.htm

Huber, N. (2003). Hitting targets? The state of UK IT project management. Retrieved May 2010 from http://www.computerweekly.com/Articles/2003/11/05/198320/Hitting-targets-The-state-of-UK-IT-project-management.htm. *ComputerWeekly.com*.

Levy, P. (2001). The Nut Island Effect: When Good Teams Go Wrong. *Harvard Business Review*, 79(3), 51-59.

Topi, H., Valacich, J., Wright, R., Kaiser, K., Nunamaker, Jr. J., Sipior, J., Vreede, G. (2009). IS 2010 Curriculum and Guidelines for Undergraduate Degree Programs in Information Systems. *IS 2010 final submission*, AIS & ACM.

IT Cortex (n.d.). Failure Causes. Retrieved April 2010 from http://www.it-cortex.com/Stat_Failure_Cause.htm

McGettrick, A., Cassel, L., Guzdial, M., & Roberts, E. (2007). The Current Crisis in Computing: What Are the Real Issues?. *SIGCSE'06*, March 7–10, 2007. Covington, Kentucky, USA. ACM 1-59593-361-1/07/0003. 329-330.

McManus, J., Wood-Harper, T., & BCS (2008). A Study in project failure. Retrieved May 2010 from http://www.bcs.org/server.php?show=conWebDoc.19584

Mraovic, B. (2003). Social audit and accountability in IT management. *Managerial Auditing Journal*, 18(3), 166-179.

National Education Association (n.d.). Research Spotlight on Best Practices in Education. Retrieved May 2010 from http://www.nea.org/tools/17073.htm; Research Spotlight on Teaching and Learning. Retrieved May 2010 from http://www.nea.org/tools/17246.htm

Overby, S. (2006). How to Hook the Talent You Need. *CIO*, 19(22), 40-54.

Peppard, J. (2007). The conundrum of IT management. *European Journal of Information Systems*, 16, 336-345.

Russell, J., Russell, B., and Tastle, W. (2005). Teaching Soft Skills in a Systems Development Capstone Class. *Information Systems Education Journal*, 3(19). http://isedj.org/3/19/. ISSN: 1545-679X. (Also appears in The Proceedings of ISECON 2004: §2224. ISSN: 1542-7382.)

Standish Group International (1994-2009). The Chaos Report. http://www.standishgroup.com

Robbins-Gioia LLC. (2001). The Robbins-Gioia Survey. http://www.robbinsgioia.com/

Thomas, J., Blackwood, M. (2009). Computer Literacy and Non-IS majors. *Proc ISECON*, v26 (Washington DC): §3753 (refereed). EDSIG.

Zhao, F. (2004). Management of information technology and business process re-engineering: a case study. *Industrial Management & Data Systems*, 104(8), 674-680.

Incorporating Capstone Courses in Programs Based upon IS2010 Model Curriculum

Ken Surendran
ksurendran@semo.edu
Department of Computer Science

Dana Schwieger
dschwieger@semo.edu
Dept. of Accounting and Management Information Systems

Southeast Missouri State University
Cape Girardeau, MO 63701-4799, USA

Abstract

Currently, most CS and IS (CIS/MIS) curricula include a capstone course to help achieve some of the program objectives such as soft-skills development. Since the scope of the IS2010 model is limited to the consideration of high-level capabilities, the recommendation lists only core courses common to all Information Systems programs and some sample elective courses. This list does not include a capstone course. In this paper, the authors examine the implications of key characteristics of IS2010 – i.e., *reaching beyond the schools of management and business* – in formulating a suitable capstone course. Based on their experiences in teaching capstone courses, they discuss the various ways in which capstone courses can be facilitated and analyze the issues influencing course design. They then suggest various strategies for incorporating capstone courses into CIS programs based upon the new IS2010 curriculum and provide a sample course outline.

Keywords: IS2010 model curriculum, Capstone course, CS/CIS and MIS programs, IS program threads

1. INTRODUCTION

Most programs in Computer Science (CS) and Information Systems - Computer Information Systems (CIS) / Management Information Systems (MIS) - culminate in a capstone course. Among other things, a capstone course provides an opportunity for students to undertake a significant project under supervision (Clear, Young, Goldweber, Leiding & Scott, 2001) in which students apply what they have learned in their program of study. It also helps in demonstrating the achievements of program objectives (Murray, Perez & Guimaraes, 2008; Schwieger & Surendran, 2010,).

We (the authors) collectively have over 20 years experience in teaching capstone courses in CS/CIS/MIS programs. We recognize the need to revise our capstone courses in light of the new IS2010 model curriculum (Topi, Valacich, Wright, Kaiser, Nunamaker, Sipior & de Vreede, 2010) which, through its broad key characteristics, cuts across the usual departmental silos. This well thought out model curriculum, with just seven core courses addressing the high-level of IS capabilities, offer considerable flexibility for designing IS programs with several threads emphasizing different application domains. It is more challenging to come up with a somewhat generic capstone course in such a flexible program. Following a systematic analysis, we present a capstone course for a CIS program that is undergoing revisions in light of the new IS2010 model curriculum. Currently, we have yet to identify all of the CIS threads. Hence, we limit the scope of this paper to just the capstone course.

For lack of space, we are forgoing a section on literature review regarding capstone courses. (For a review of capstone course literature, we refer the reader to Clear, et al. 2001.) In the next section, we discuss the current IS programs at the authors' university and the relevance of the IS2010 model in our current curriculum development plans. In Section 3, we identify the important issues surrounding capstone courses and the various ways in which this course is currently facilitated. In Section 4, we suggest, based on the previous sections, a few strategies for incorporating a capstone course into a CIS program that reflects the characteristics of the IS2010 model. Finally, we present a high-level course description of a capstone course that is generic yet flexible enough for adoption in our revamped, multi-threaded CIS program.

2. IS2010 AND OUR IS PROGRAMS

At the authors' institution, there are two Information Systems programs: MIS in the College of Business, having AACSB accreditation, and CIS in the College of Science.

MIS

The MIS program was first designed using IS1997 model curriculum and input from industry (Ehie, 2002). The program was later revamped to reflect the IS2002 extensions. The intent of the MIS program was limited to the management and business domains. Like most MIS programs, our program has experienced a steady decline in student enrollment.

CIS

The CIS program was designed to be more like an Applied Computer Science (ACS) degree. Differing from its sister programs, this program has fewer higher level CS courses relative to the CS program and very few overlapping courses with the MIS program. Unlike CS, CIS has no requirements for science courses other than those required under the general education requirements for all majors. Instead, the CIS program requires students to complete a minor, or another major, in an unrelated area of study. Like MIS, the CIS program is experiencing, as of late, a decline in enrollment numbers.

Revising IS Programs

When the CS and IS departments started seeing declining enrollments, these declines were initially credited to the dot com burst along with offshore outsourcing (Rajaravivarma & Surendran, 2006). However, soon realizing that these declines were permanent, many institutions decided to redesign their curricula (e.g., McGann, Frost, Matta & Huang, 2007).

Based upon periodic reviews, the IS Curriculum Task Force came up with the current IS2010 model curriculum (Topi, et al., 2010) that is flexible, domain-independent and well structured. Similar to the intent of the above CIS program, it allows, unlike IS2002, the inclusion of any application domain (i.e., going beyond schools of management and business).

IS2010 specifies a set of structured outcome expectations starting with high-level IS capabilities which are translated into three categories of knowledge and skills: foundational, IS specific and domain fundamentals. The framework has only seven core courses and provides descriptions of a few elective courses. Obviously, a capstone course is outside its scope and specifying one might be considered prescriptive.

Implications to MIS and CIS

In the case of the MIS program at the authors' institution, the domain remains business focused. The revisions, therefore, have to do with courses that are IS-specific.

The original intent of the CIS program (entitled Applied Computer Science (ACS) at that time) has been to provide a generalized curriculum in the applied aspects of computing or informatics (Duben, Naugler & Surendran, 2006) to complement the CS program. Although this CIS program is attempting to address the domain fundamentals of IS2010 (by requiring a minor or another major), it lacks courses that link computing with the application domains pertaining to those minors or majors. Hence, in the case of CIS, we expect the revision to be extensive since we need to address both the IS specifics (revising its core courses) and domain fundamentals (designing *domain-specific IS* courses).

We will start with identifying candidate domains with each becoming a CIS Thread. The initial academic threads for consideration, other than business, are: arts/entertainment, healthcare, law/security and science. The respective domain interface courses will have to be jointly designed in conjunction with faculty from the concerned departments. The idea of requiring a minor may be retained as it can be absorbed into the respective thread.

Irrespective of how these CIS threads are formulated, we intend to retain a capstone course in the program that is equally flexible to implement. Because we are designing a capstone course ahead of formulating the various CIS threads, flexibility is of paramount importance. Otherwise, this bottom-up design might require refinements as new threads are added.

3. CAPSTONE COURSE

A capstone course, as the name implies, is intended to provide students with a culminating and integrative learning experience. Depending upon the circumstances, the students in a capstone course develop a product or carryout a research study. Usually, students enrolled in computing curricula work in groups on a client-sponsored project (Williams, Bair, Borstler, Lethbridge & Surendran, 2003) to offer real-world experience. Capstone courses can provide a comprehensive experience for the students addressing soft skills, experiential learning, conceptual elements as well as career readiness (McGann & Cahill, 2005). Like other courses, a capstone course will have a set of learning outcomes pertaining to both technical and professional skills. Clear et al. (2001) considered the following issues that normally require attention in facilitating a capstone course: *goals of the courses, characteristics of projects, project deliverables, sponsors, teams, prerequisites and preparation, grading and assessment, administration and supervision, reflection, analysis and review*. Similarly, the main issues pertaining to the design of a capstone course in our context include: type of capstone course, student learning outcomes, nature of the project to suit different IS program threads, matching assessments (including project deliverables), and selection of topics. In the following sections, we examine these issues in some detail.

Types of Capstone Courses

Capstone courses vary depending on the educational objectives of the program. We discuss below, based on one of the author’s experiences, three capstone course variations as well as alternatives available for providing such culminating experiences.

Regular capstone: Senior level capstone courses are most often offered for three credit hours. In this proposed course, a team of four or more students work for a semester (or two quarters) on a client-sponsored project (Surendran & Young, 2001). The instructor interacts with the client to identify projects and lets the students select the projects on which they wish to work. As the students start working on the projects, the instructor supervises the teams closely. The students struggle to manage time, in view of their other courses, along with the requirements of the capstone course and project.

This course is taught much like any other course following a class syllabus. However, some of the class time is allocated for working on the client-sponsored project deliverables. Class time is also allocated for student presentation of several of the intermediary products including: project scope and plan, requirements specification, design specification and user interfaces. Because they work in teams, this course offers considerable opportunities for students to hone their professional (soft) skills. At the end of the semester, students present their products to the client and faculty for final evaluation.

It is also possible to design a university-wide capstone course (Schwieger & Surendran 2010) where students from different educational backgrounds work together on a project. Even though such a course is difficult to coordinate, they offer considerable flexibility and integrative learning opportunities for the students. For instance, two CIS students can work on the IS components of a capstone project assigned to a group of students from another application discipline (i.e. music, biology, history, etc.). In such projects, the IS students get considerable opportunities to develop professional skills.

Intensive capstone: In a CIS program that focuses on developing work-ready graduates, the capstone course has the equivalent of six credit hours (Surendran & Young, 2001). In this experience, one or two students, based at the client site, work full time on a capstone project for a semester. The students identify the client and a project. The client and the instructor work together to scope the project and then expectations are communicated to the students. Even though the students work at the client site, they meet with the instructor once a week to discuss the progress and intermediary deliverables (e.g., analysis and design documents). The students follow a project plan. They also interact with the client constantly and follow the client's house-standards in developing the product. At the end of the semester, the student(s) present their products to the client and the entire faculty in the department who then evaluate the work. This can work better if the client is located somewhat close to the university. However, in view of the current workplace technologies, proximity to the client is not necessarily a factor.

Product-driven capstone: In programs where the emphasis is not on system development but on using and supporting enterprise applications, the capstone course can be centered on a comprehensive domain-specific product (e.g., an Enterprise Resource Planning system). In this experience, the students learn an enterprise product and the associated tools provided to carry out simple maintenance suggested by the instructor (Surendran, Somarajan & Holsing, 2006). Students normally work on these exercises in pairs. Such a course requires extensive instruction and close supervision.

Alternatives: Two alternatives to a system development project-based capstone course are "Research on CIS topics" or an "Internship in CIS." Both of these options are ideal for students who are capable of working independently and have specific goals in mind. The first one may be especially appropriate for students who intend to pursue graduate coursework. In regards to the second alternative, instead of trying to simulate a system development apprenticeship (Surendran, Hays & Macfarlane, 2002) through a project-based capstone course, an appropriately instructor-managed internship in CIS could provide a more realistic apprenticeship experience to the students.

Student Learning Outcomes

Student Learning Outcomes (SLOs) for capstone courses vary depending upon where the IS program is located (since program outcomes are somewhat departmentally dependent) and the role of the course in the program. Often times, capstone courses include outcomes that help achieve some of the program objectives that are hard to achieve in other courses (e.g., (1) demonstrate fundamental IS (or system development) skills on a non-trivial project, and (2) demonstrate the ability to communicate effectively). While using the capstone experience course as a program assessment tool, Murray et al. (2008) considered both the general SLOs and [discipline-] specific SLOs.

Our university offers two programs under IS: MIS that is located in the College of Business and CIS that is located in the College of Science. Several of their SLOs overlap as students work client-sponsored projects in both courses. We can group the programs' SLOs under two categories: those pertaining to technical skills development and those pertaining to professional skills (soft skills) development.

Example SLOs for technical skills:

1. Apply concepts and techniques (or knowledge from their major discipline) for developing quality software products.
2. Create analysis and design documentation pertaining to the system being developed.
3. Discuss project management and communications management issues in software development.
4. Discuss the various testing concepts for establishing quality assurance.

Example SLOs for professional skills:

1. Obtain practical experience with working on an information systems development project in a team environment.
2. Orally present the intermediate system artifacts (generated during analysis and design) for review and evaluation.
3. Carry out research on a recent development in the field of software development and present it to the class.

Other possible SLOs: Different learning outcomes may be needed when the capstone course does not involve a system development project. This is especially true if the course is offered as part of a university-wide general education requirement offered through the College of University Studies (http://ustudies .semo.edu/handbook/misc/objectives.html. In such situations, it may be necessary to have two sets of learning outcomes, one that is major specific and the other that is common (generic) to all majors. Some examples of generic SLOs are:

1. Demonstrate capabilities for critical thinking, reasoning, and analyzing.
2. Demonstrate effective communication skills.
3. Demonstrate the ability to integrate the breadth and diversity of knowledge and experience.
4. Demonstrate the ability to make informed, intelligent value decisions.
5. Demonstrate the ability to function responsibly (ethically) in one's professional environment.

Projects for Different CIS Threads

In order to simulate real-world experience, we use client-sponsored projects (as opposed to instructor-specified ones) in a capstone course. These clients are from an actual business or industry. The instructor plans these projects prior to assignment, identifying and scoping client-sponsored projects to suit student-team sizes and their workloads (Williams, et al., 2003). One student in each team takes on the role of a manager while the instructor takes the role of a project director overseeing all of the class' projects.

Current projects

Currently, the instructor compiles all of the project outlines from the clients and presents them to the students during the semester prior to the assignment. The students form their own teams and choose their projects. These projects come from various application domains and are sponsored by different organizations. Thus, some projects may require students to research and learn new tools. In some cases, the students may have to seek additional domain knowledge. Listed below is a sample of the projects (the application system and the type of sponsoring organizations) the authors supervised in past years from different institutions:

- **Workflow management systems** (IT functions from telecom, auto-parts manufacturing, and food product companies)
- **Purchase order systems** (Wooden cabinet manufacturing and radiator manufacturing companies)
- **Sales system** (Web hosting service provider)
- **Inventory management system** (Regional food-bank – non-profit)
- **Maintenance management** (Two local IT service companies)
- **Contract management system** (Medical-equipment supply company and software consulting company)
- **Trucking and dispatching system** (Wooden cabinet manufacturing company)
- **Labor scheduling** (Local gas-station chain and library at the University)
- **Time clock system** (Wooden cabinet manufacturing company)
- **Training management system** (Athletics Department at the University and law membership training enforcement – by court house)
- **Billboard management system** (Local advertising company)
- **Flight data simulation** (Aircraft manufacturing company)
- **Optic bench (2-D) simulation** (Printer manufacturing company)
- **Chat facility** within online instruction suite (Learning technology unit at the University)
- **Online logic puzzle** (Local IT consulting company)
- **Diagnostic articulations test systems** (Paramedical training unit at the University)
- **Academic music search system** (Music Department at the University)
- **Set list / gig manager** (Local IT consulting company)
- **Test score evaluation** (Elementary school)
- **Course management system** (Small Business Development Center at the University)
- **Scholarship management system** (Financial Aid Department at the University)

CIS Threads and Projects

Program Threads, or focus areas (application domains) outside CIS (i.e. Music, History, Biology, etc.) may be a convenient mechanism to exploit the domain enhancements characterized in the IS2010 Model Curriculum Guidelines. Each CIS thread will focus on one CIS/application domain combination. Each thread will require a few courses from the application domain and two or three CIS specific courses. The capstone course will be common to all CIS threads with appropriate projects chosen for the students in the different threads. For instance, students in the Music Thread could be given, from the above list of projects, "*Academic music search system"* or the *"Set list/gig manager"* projects. Student groups in the Education Thread could be given the "*Test score evaluation project*." Student groups in the Art/Entertainment Thread could be given the "*Online logic puzzle,*" the *"Flight data simulator,"* or the "*Optical bench simulator project."* Health Thread student groups could be given the "*Diagnostic articulation test system*" or the "*Medical equipment supplier's contact management system."* Business Thread students can be given several of the standard business management projects, especially those from manufacturing organizations.

Assessments and Deliverables

The main assessment component is the completion of the client-sponsored project by the teams. Table 1 provides a breakdown of the project for assessment and grading purposes.

Table – 1: Assessment and Grading

Assessment	Weight
Project Presentations in Class (four)	20%
Ethics Presentation	10%
Project Reports 3 parts: Analysis - 5% Design - 10% User interface – 5%	20%
Project Review (participation)	5%
Project Demo to Advisory Board	10%
Final Project Report	10%
Personal Reflection (Individual)	5%
Thread Related Assessments	20%

The assessments will involve oral presentations, a demonstration of the project, and written reports (delivered in four/five stages). For details on the deliverables, see Schwieger & Surendran (2010).

Topics

Currently, the CIS/MIS programs do not have a project management course. In these programs, the primary system development workflows are taught in the software engineering/systems analysis and design courses. The support workflow topics, however, are taught in the capstone course: project management, communications management, quality assurance and configuration management. During the last academic year, additional topics such as multi-cultural work environment and global ethical & psychological perspectives were added to course content. A faculty from Global Studies facilitated these sessions. See Appendix-A for a sample schedule.

4. STRATEGIES FOR THE CAPSTONE COURSE IN A PROGRAM BASED ON IS2010 MODEL

The IS2010 model identifies (Topi et al., 2010) *designing and implementing information systems solutions* as one of the IS specific knowledge and skills. The importance of this skill is highlighted by the observation (Topi, et al., 2010) that *the industry would prefer graduates with the ability to integrate high performance in design and implementation along with strong business [domain-specific] capabilities*. Several of the seven core courses and sample elective courses such as *Application Development, Human Computer Interaction,* and *Enterprise Systems* are intended to develop these particular IS specific knowledge and skills. The new CIS curriculum will include these three electives as core courses since they address the basic implementation knowledge and skills.

Most present day IT professionals do not get the opportunity to develop new systems from scratch. Instead, they work on enhancements to and customizations of larger enterprise systems in various application domains (Surendra & Denton, 2009). In view of such realities, a flexible capstone course needs to evolve. Five possibilities described below are suggested based upon existing practices.

Conventional

Under this option, the capstone course students carry out client-sponsored projects from scratch. The main difference is that faculty need to ensure that the projects are from application domains relevant to the CIS thread (as discussed under section 3).

Enterprise System Centered

Capstone courses, under this category, are classes in which students practice a thread-specific (domain-specific) enterprise system and carry out enhancements (customization). Here, a pair of students work on a set of enhancements to an existing system using the development tools prescribed for the enterprise system. Projects on ERP systems (e.g., SAP and Business Dynamics) are examples for the CIS Business Thread. Open source enterprise systems (e.g., Angel) may be relevant for the Education Thread.

General Education Flavored

Most majors have capstone courses with projects in their respective fields. Last semester, we used a university-wide capstone course for CS, CIS, and MIS majors in which students from two other majors participated. A university-wide capstone course, offered under the general education umbrella, has the potential for allowing students from unrelated disciplines to work on projects that span different domains. These days, such projects may have IS/IT components. In such a university-wide capstone course, one or two CIS students can work on IS aspects of the project (pertinent to their thread) along with a team of non-CIS major students. Usually, instructors from the respective disciplines jointly facilitate such courses.

Apprenticeship

Several large IT organizations offer internship programs (Computer Weekly, 2010). Likewise, most university CS, CIS, and MIS programs have internship courses. Such courses could be turned into capstone courses where a student carries out a set of activities centered on specified learning objectives. The student will then report to an instructor from the university periodically while working under the supervision of a mentor in an organization.

Cross-Discipline Independent Study

Most programs offer independent study courses where a student learns advanced topics. It is possible to have a capstone course that is more like an independent study where a student works (or a pair of students work) on developing a system tool or carrying out a research study involving IS in a particular thread. Here, it is possible that the students will be interacting with two instructors, one from CIS and the other from the thread area.

5. HIGH-LEVEL DESCRIPTION

It is helpful to use different mechanisms for offering flexible, multi-threaded capstone courses in IS programs based upon the IS2010 model. However, it is not possible to incorporate all of the learning objectives of a conventional capstone course into all of the forms of capstone courses. Perhaps one approach to addressing the objectives would be to describe a capstone course with a few learning outcomes pertinent to IS and a particular thread area (application domain). A provision could be included to add additional learning outcomes depending upon the chosen course offering mechanism.

Catalog Description

University catalog course descriptions are usually rather brief. In Figure-1, however, we provide a longer version of a possible catalog description for a multi-threaded capstone course. In this description, we do not discuss any specific topics to be covered. See section 3 for possible topics including global perspectives and ethics.

Figure – 1: Catalog Description

> This course offers a choice of flexible learning mechanisms including: system building/enhancement projects, apprenticeships, and independent studies in applying IS knowledge in the chosen domain. System development skills are integrated throughout the course via requirements analysis, system design and implementation, managing enterprise systems specific threads, and managing projects. Students may work on client-sponsored projects or instructor-specified studies as individuals, in pairs, or in larger teams including members from other majors. Students apply professional heuristics and tools essential to the system development process throughout the course.

Learning Objectives

The capstone course, described above, provides evidence for assessing the following program objectives:

1. Demonstrate an understanding of information system fundamentals.

2. Demonstrate IS development / enhancement skills on a non-trivial project to the satisfaction of a client in a chosen application domain.
3. Be prepared to enter the workforce as an entry level information system specialist in a chosen application domain.

Furthermore, additional learning outcomes could be included for specific mechanisms. For instance, we could include the following two outcomes under all of the mechanisms except for the independent study mechanism:

1. Demonstrate the ability to communicate effectively.
2. Demonstrate critical thinking skills.

In the case of the enterprise centered mechanism, we could include:

1. Demonstrate understanding and use of an enterprise system in an application domain.
2. Demonstrate the use of tools to carry out enhancements to an enterprise system.

In the case of the independent study mechanism, we could include:

- Demonstrate the ability to learn advanced topics and apply IS skills to develop tools for use in an application domain.

6. CONCLUSION

The new IS2010 model provides considerable opportunity to enhance the IS program for reaching out to all application domains. At the authors' institution, we are in the process of revising our CIS program to have several threads for incorporating different application domains. In this paper, we described, based on our experiences, a process for developing a possible capstone course in IS programs that are based on the IS2010 model. We suggest a flexible approach to cater to the variations in the level of implementation skills that includes use of client-sponsored projects, enterprise system based projects, instructor-directed apprenticeships in industry, and cross-discipline focused independent study. We also provide an enhanced catalog description and a set of flexible learning objectives for a capstone course.

Because we have taken a mixed design approach (top-down for core and bottom up for capstone), we will continue with our efforts to refine the course as we identify the various CIS threads. Likewise, cross-discipline (CIS and application domain) courses will be designed for those threads as well.

7. REFERENCES

Clear, T., Young, F., Goldweber, M., Leiding, P., & Scott, K. (2001). Resources for Instructors of Capstone Courses in Computing, Working Group report at ITiCSE-2001. *SIGCSE Bulletin –Inroads*, *33*(4), 93-113.

Computer Weekly (2010). Microsoft launches apprenticeship scheme for 3,000 young people. Retrieved June 3, 2010 from http://www.computerweekly.com/Articles/2010/04/22/241012/Microsoft-launches-apprenticeship-scheme-for-3000-young.htm.

Duben, A. J, Naugler, D. R., & Surendran K. (2006). Agile Computing Curricula. *Information Systems Education Journal*, *4*(53), http://isedj.org/4/53/ ISSN: 1545-679X.

Ehie, I. C. (2002). Developing a management information systems (MIS) curriculum: Perspectives from MIS practitioners. *Journal of Education for Business*, *77*(3), 151-158.

McGann, S. T., & Cahill, M.A. (2005). Pulling it all Together: An IS Capstone Course for the 21st Century Emphasizing Experiential & Conceptual Aspects Soft Skills and Career Readiness. *Issues in Information Systems*, *6*(1), 391-397.

McGann, S. T., Frost, R. D., Matta, V., & Huang, W. (2007). Meeting the Challenge of IS Curriculum Modernization: A Guide to Overhaul, Integration, and Continuous Improvement. *Journal of Information Systems Education*, *18*(1), 49-62.

Murray, M., Perez, J., & Guimaraes, M. (2008). A Model for Using a Capstone Experience as One Method of Assessment of an Information Systems Degree Program. *Journal of Information Systems Education*, *19(*2), 197-208.

Rajaravivarma, R., & Surendran K. (2006). Coping with Offshore Outsourcing and Enhancing Student Retention. *Information Systems Education Journal,* *4*(92), http://isedj.org/4/92/. ISSN: 1545-679X. (A preliminary version appears in *The Proceedings of ISECON 2005:* §2364. ISSN: 1542-7382.)

Schwieger, D., & Surendran, K. (2010). Enhancing the Value of the Capstone Experience Course. *Information Systems Education Journal,* *8*(29), http://isedj.org/8/29/. ISSN: 1545-679X. (A preliminary version appears in *The Proceedings of ISECON 2009:* §3313. ISSN: 1542-7382.)

Surendra, N. C., & Denton, J. W. (2009). "Designing IS Curricula for Practical Relevance: Applying Baseball's Moneyball's Theory. *Journal Information Systems Education*, *20*(1), 77- 86.

Surendran, K., Hays, H., & Macfarlane, A. (2002). Simulating Software Engineering Apprenticeship, *IEEE Software, 19*(5), 49-56.

Surendran, K., Somarajan, C., & Holsing, D. (2006). Incorporating ERP into MIS Curriculum: Some Insights. *Information Systems Education Journal*, *4*(25), ISSN: 1545-679X http://isedj.org/4/25/ , July 2006

Surendran, K., & Young, F. H. (2001). Teaching Software Engineering in a Practical Way. *The New Zealand Journal of Applied Computing and Information Technology*, *5*(2), 75-79.

Topi. H., Valacich, J. S., Wright, R. T., Kaiser, K., Nunamaker, J. F., Sipior, J. C., & de Vreede, G. J. (2010). IS 2010: Curriculum Guidelines for Undergraduate Degree Programs in Information Systems. *Communications of the Association of Information Systems*, *26*(18), 360-429.

Williams, J. C., Bair, B., Borstler, J., Lethbridge, T. C., & K. Surendran (2003). Client Sponsored Projects in Software Engineering Courses. *Proceedings of the 34th SIGCSE Technical Symposium on Computer Science Education*, SIGCSE 2003, ACM. 401-402.

Appendix –A: Topics and Assessment Schedule

Week	Topics
1	Course overview, Project Management (plan)
2	Presentation (1) Project Scope and Plan
3	Communications Management
4	Presentation (2) Requirements Specification; Requirement Spec. due
5	Project Management (control)
6	Working in the global village (Overview of cultural divergence)
7	Presentation (3) Prelim results of the project; Req, Analysis Report due
8	Influence of psychological value in group efforts
9	Global perspectives on Ethics – Case Studies distributed
10	Presentation (4) Project Design (some parts)
11	Presentation of ethical case studies (5) Project Design Report due
12	Project review meeting (Submit Progress report)
13	Project review, planning session for final product/findings presentation
14	Preliminary project presentation to class; formal presentation weekend
15	Project demo of final product/findings to public
16	Project documentation along with system binary due

Predicting Success in the Introduction to Computers Course: GPA vs. Student's Self-Efficacy Scores

Joseph T. Baxter
Management Information Systems
jbaxter@daltonstate.edu

Bruce Hungerford
Management Information Systems
bhungerford@daltonstate.edu

Marilyn M. Helms
Management
mhelms@daltonstate.edu

Dalton State College
Dalton, GA 30720, USA

ABSTRACT

This study examines whether students' final grades in an introductory college business computing class correlate with their self-reported computer skill levels provided at the beginning of the course. While significant research effort has been devoted to studying the effects of student self-efficacy on course outcomes and studying the moderating effects of various demographic variables (such as age and gender) and experience variables (such as computer access at home), there is a dearth of studies examining a student's grade-point-average (GPA) as a predictor of final course success in the introductory computing class. For the fundamentals of computer applications course at the medium-size state college, student self-perceptions of their own computer abilities explained very little of the variation in the final course grade outcomes. GPA, however, was a more powerful predictor (adjusted R^2 = 0.365) of the final class grade as well as the students' grades on individual course modules. Students' perceptions of their own computer abilities added very little additional predictive value, increasing the full model's adjusted R^2 only to 0.393. Given the predictive power of GPA relative to course success, discussion is included concerning ways to use this information to offer additional assistance to lower performing students. The study contributes to the existing literature and refutes the value of self-assessment of skills and abilities as a sole predictor of success. Although the literature has suggested non-traditional or adult students may have more difficulty with the computer course, our findings do not support this. Areas for future research are suggested.

Keywords: information literacy, business student, introduction to computers, self-efficacy, computer literacy

1. INTRODUCTION

Information literacy research is growing due to the Internet, digital media, and the pervasiveness of personal computers. With electronic media and devices proliferating, what encompasses computer literacy and fluency becomes a changing construct and universal definitions still do not exist. McDonald (2004) agrees the definition of computer literacy continues to change as technological innovations are adopted by the marketplace. There is broad agreement, however, that college students need computer and information literacy as part of their studies to be competitive as graduates in an environment that increasingly relies on information technology.

The challenge for universities is to ensure their students meet a minimum level of competency when using constantly changing technology. McDonald (2004) further suggests universities incorporate flexible testing tools to measure basic computer skills such as an Internet-based, interactive skills test. Hawkins and Oblinger (2006) indicate technology is nearly ubiquitous on campus; and, although conversations about the digital divide are relatively uncommon today, it remains incorrect to assume all students own a computer or have an Internet connection.

2. LITERATURE REVIEW

Colleges have traditionally used the freshman- or sophomore-level course in microcomputer applications/introduction to computers to accomplish basic computer literacy. The purpose of this research is to determine if predictors exist for student success in this course. Most studies focusing on students' skill and success in the introduction to computers course at the college level examine a variety of experience variables, demographic variables, and students' self-reported skill levels on a variety of microcomputer applications. This latter variable is termed self-efficacy.

Self-Efficacy

Self-efficacy is a social cognitive construct popularized in the 1970s and later formally defined by Bandura (1986) as "people's judgments of their capabilities to organize and execute courses of action required to attain designated types of performances" (p. 39). Bandura (1986) argued beliefs about efficacy influence a person's choice of activities, the level of effort an individual is willing to expend, their persistence even in the presence of difficulties, and their overall performance. Self-efficacy remains a key concept in social cognitive theory.

Computer Self-Efficacy

Computer self-efficacy is a derivation of self-efficacy in general (Bandura, 1986, 1997) and has been defined as "...a judgment of one's ability to use a computer" (Compeau & Higgins, 1995, p. 192). It has also been studied and found to be a determinant of computer-related ability and the use of computers (Hasan, 2003). Hasan and Jafar (2004) empirically examined computer-learning performance and used the definition of self-efficacy as referring to an individual's judgment of their own capabilities to organize and execute courses of action to attain designated performance.

Compeau, Higgins, and Huff (1999) found that those individuals with a higher computer self-efficacy participate in computer-related activities, expect success in these activities, persist and employ effective coping behaviors when encountering difficulty, and exhibit higher levels of performance than individuals low in computer self-efficacy. Karsten and Roth (1998a) found that the construct captures the competence and confidence management information systems (MIS) professors hope to provide their students. The construct has been studied in depth by a number of MIS researchers (see Agarwal, Sambamurthy, & Stair, 2000; Marakas, Johnson, & Clay, 2007; Marakas, Yi, & Johnson, 1998).

Students gain self-efficacy from a variety of sources (Bandura, 1997), including their personal successes and failures, observing the successes and failures of friends and classmates, encouragement, and confidence or anxiety when faced with tasks. Computer self-efficacy is dynamic, changing as students gain new information and computer-related experiences (Gist & Mitchell, 1992), but Karsten and Roth (1998b) found that it is the kind of experience, and not just the experience per se, that changes perceptions of computer self-efficacy. Marakas, Yi, and Johnson (1998) studied the construct and separated task-specific measures of computer self-efficacy from general computer self-efficacy. They defined the task-specific measures as "...an individual's perception of efficacy in performing specific computer-related tasks within the domain of general computing" (Marakas, Yi, & Johnson, 1998, p. 128).

Hasan (2006a) further described the construct as a judgment of efficacy (or success or skill) in performing a well-defined computing task using a particular application, including word processing, spreadsheet, or database programs. Qutami and Abu-Jaber (1997) studied gender and cognitive learning styles to determine students' self-efficacy in computer skills. Shiue (2003) studied the effect of cognitive learning styles and prior computer experience on students' computer self-efficacy in computer literacy courses, while Albion (2001) used pre- and post-tests and found that students' self-efficacy, after completing computing courses, was influenced by their personal ownership of computers. Hsu and Huang (2006) found that computer use and interest had a significant, direct effect on student computer self-efficacy. They further found that computers at home and work had indirect effects on computer self-efficacy as did trend motivations and interests. Computers are pervasive in business and education, and Buche, Davis, and Vician (2007) agree it would be easy to assume that all individuals embrace technology. However, their study found that 30 to 40 percent of individuals experience some level of computer anxiety.

Self-Efficacy, Computer Literacy, and Demographics

Studies of the first computer course have explored gender differences in self-efficacy and attitudes toward computers (Busch, 1995). Busch (1996) added group composition and cooperation variables to his subsequent study. Introductory information systems course-related factors were also studied as indicators of computer self-efficacy (Karsten & Roth, 1998a), while Houle (1996) studied student differences (including a variety of demographics characteristics) in his research on understanding student differences in computer skills courses.

Davis and Davis (2007) surveyed 58 students in technology teacher education and training to determine self-perception of their competency in five constructs made up of 43 elements related to personal computer knowledge and skills. While gender did not make a difference, they did find a statistically significant difference between the perceived competencies of the participants based on age range, with students 35 years old or younger perceiving a higher level of competence compared to those 36 years old or older. Further analysis within construct variables revealed instances of statistically significant differences based on gender and age range.

Divaris, Polychronopoulou, and Mattheos (2007) agree an accurate assessment of the computer skills of students is a pre-requisite for success in other areas, including e-learning. They studied 50 post-graduate students and calculated competence scores and gathered socio-demographic characteristics. Using both descriptive statistics and linear regression modeling, the authors found that competence scores were normally distributed but that gender and use of e-mail were significant predictors of computer literacy.

Students believe they are computer literate according to Wilkinson (2006). Her research compared students' perceptions with reality and found that students did not perform well on pre-tests of Microsoft Office™, but improved their post-test scores with instruction. She found that a comparison of student classifications regarding perceptions with the reality of computer productivity yielded no significant differences but did find significant differences between Caucasian students and ethnic minorities.

Goh, Ogan, Ahuja, Herring, and Robinson (2007) investigated the relationship among computer self-efficacy, mentoring, and the gender of students and their mentors. Students with male mentors reported significantly higher computer self-efficacy as compared to those students with female mentors. Kuhlemeier and Hemker (2007) studied the impact of secondary students' use of the Internet and the computer at home on the digital skills they need in school and found that home access to e-mail and students' use of home computers for various tasks, including surfing, e-mailing, chatting, and text processing were related to Internet and computer interest. Ballantine, Larres, and Oyelere (2007) studied the reliability of self-assessment as a measure of computer competence. They agreed recent research on the topic has employed self-reported ratings as the sole indicator of students' computer competence. They compared the self-assessment to results on objective tests and found that students significantly over-estimated their level of computer competence. Interestingly, they found that students' home and high school computer use did not affect the results, and they questioned the use of self-assessment as a measure of computer competence.

In their study of the digital divide, Tien and Fu (2008) used multiple regression and logit models and found that demographic and socioeconomic family background did not predict computer skills of first year college students. They did find that different kinds of computer knowledge affect student learning with knowledge of software helping students learn the most. Some differences in computer knowledge were found among female students, minorities, and those with blue-collar or unemployed parents. These students were at a disadvantage in digital understanding. Banister and Vannatta (2006) suggest colleges must develop strategies to assess technology competencies of beginning college students and then move beyond such assessments to provide student support for achieving technological competencies. They found that various methodologies have been used to measure a student's computer competencies but agree that there are no standardized scales for assessing competence.

In her study of the introduction to computers course, Webster (2004) examined the relationship between computer use confidence and computer literacy scores before the course began and repeated the test at the end of the course to assess gains in computer usage confidence and literacy. She found that prior computer classes and computer usage positively influenced literacy scores and confidence. In addition, she found that hours using the computer for e-mail purposes also influenced confidence scores. After completing the introductory course, students had higher confidence and literacy ratings than the control group.

In their study of the reported experience, comfort level, and perceived information technology skills of 233 college students, Messineo and DeOllos (2005) found that students view their computer competence differently depending on whether they are using the technology for personal or course-related tasks. They discovered that even when the expressed levels of experience and comfort for some forms of technology were high, exposure to and confidence with more advanced applications were lacking. They agreed faculty members may make false assumptions about student preparedness, which hinders their students' success. Their research found differences by gender and race/ethnicity and suggested faculty should be aware of the varied skill levels and experiences of their students.

Cassidy and Eachus (2002) developed a computer user self-efficacy scale. They agree self-efficacy beliefs have been identified as a success factor for completing tasks. With the increasing reliance on computer technologies in all aspects of life, it is important to measure the construct. Their research found a significant positive correlation between computer self-efficacy and computer experience. Familiarity with computer software packages was a significant predictor of computer self-efficacy, and computer ownership and training increased efficacy. In their study, males reported higher results than females. This supports the findings of Varank (2007) who found that gender was significant for predicting computer attitudes but not perceived skills. In Mayall's (2008) study of technology self-efficacy among high school students, no statistically significant differences based on gender were detected in either pre- or post-tests.

Stephens (2006) found that subjects with low computer self-efficacy will avoid interacting with computer technology when given a choice or opportunity. Oblinger and Hawkins (2006) suggest that when faculty, staff and administrators see how easily students use technology, they may mistakenly assume students have more than adequate IT competency. They question whether students are competent or just overly confident and caution having no fear is not the same as having knowledge or skill. Stephens (2005) developed a decision support system built around a self-efficacy scale that can be implemented to perform training needs assessment. The system can determine who requires training and which training mode is most appropriate.

Computer Course and Instruction

In their research, Creighton, Kilcoyne, Tarver, and Wright (2006) ask two related questions: Is a freshman-level microcomputer applications/introduction to technology course obsolete? Are students, especially new freshmen, enrolling in the course already computer literate? Their research found that students enrolling in such courses were not literate in general computer technology and spreadsheet applications, but were computer literate in the more familiar and often used word processing, e-mail, and Internet applications. They found the higher the ACT score, the better the students scored on the objective pre-test exam and the performance-based post-test exam, but found only a weak relationship between taking a previous computer course and pre-test scores.

Hollister and Koppell (2008) studied the information technology course in an assurance of learning program in an undergraduate program at an AACSB accredited business school to redesign the content and pedagogy of the computer literacy course. Mykytyn (2007) agrees that while colleges of business have dealt with teaching computer literacy and computer application concepts for many years, teaching tool-related features in a lecture in a computer lab may not be the best instructional mode. He suggests problem-based learning as an alternative for teaching computer application concepts, operationally defined as Microsoft Excel™ and Access™. Ballou and Huguenard (2008) studied an introduction to computer course with both a lab and lecture component and found that higher levels of perceived computer experience positively affected lecture and lab homework and exam scores.

Hindi, Miller, and Wenger (2002) investigated students' perceptions of computer literacy skills they had obtained prior to enrolling in a university to develop implications and recommendations for teaching a college-level computer course. Students perceived themselves better prepared in word processing than they were in spreadsheet and database applications. However, computer self-efficacy measures suffered from degradation of their explanatory power over time (Marakas, Johnson, & Clay, 2007).

The proposed research model developed by Hasan (2006b) makes a clear distinction between general and application-specific computer self-efficacy and found that both had negative effects on computer anxiety. A model by Thatcher and Perrewe (2002) found that computer anxiety mediates the influence of situation-specific traits on computer self-efficacy.

Karsten and Schmidt (2008) in their ten-year study of business student computer self-efficacy found that when controlling for changes over time, students have lower computer-self efficacy in 2006 than in 1996. It was surprising that increased use of computers and technology over time did not lead to higher self-efficacy scores. Sharkey (2006), in her study of information fluency and computer literacy, found that universities are responding with a more rapid integration and adoption of technology and emphasizing information use and retrieval. Findings on self-efficacy and computer skills acquisition among graying workers by Reed, Doty, and May (2005) suggested older participants' beliefs about their efficacies in acquiring computer skills were lower than their actual abilities.

GPA

Research has considered a number of demographic variables as determinants of student performance in various business courses. Trine and Schellenger (1999) studied determinants of student performance in an upper level corporate finance course and found that GPA, the financial accounting grade, basic finance grade, math ACT, a self-motivation factor, an information processing factor, and sharing living quarters with non-family members were all significant in determining the student's course performance. Typically the list of variables is more limited to gender, age, or ethnicity.

Wilson, Ward, and Ward (1997) found that both self-reported and actual data on ACT scores, GPAs, and grades earned in specific courses were similarly correlated with accounting course performance. Christensen, Fogarty, and Wallace (2002) studied the directional accuracy of self-efficacy and performance in accounting courses mid-way through the academic term. They found the more conservative a student's self-efficacy of their skill levels and abilities, the higher the second exam score and final course grade, even when controlling for cumulative GPA in accounting courses, average exam performance, number of accounting classes completed, and the extent of involvement in extracurricular activities. There is a dearth of studies focusing on these demographic characteristics, specifically GPA, in the introduction to computers course.

Table 1 (Appendix A) summarizes variables affecting computer self-efficacy and/or performance for selected research.

3. METHODOLOGY

The literature on students' self-efficacy in general and computer self-efficacy in particular considers a number of demographic characteristics and skill levels. However, inconsistencies remain as to which variables have the most predictive power. Our study extends the research on this topic and gathers data on student's self-efficacy in a number of specific computer applications. The research examines if the self-rated skill sets are predictive directly or indirectly of the knowledge-level scores of the various computer applications or of the final, overall course grade and whether such predictions are significantly improved by the inclusion of demographic and prior experience variables. The model shown in Figure 1 (Appendix B) indicates our hypothesized relationships and leads to our hypotheses.

H1: Student self-assessment of skills and abilities with Windows XP™ and selected Microsoft Office™ applications (Word™, Excel™, PowerPoint™, and Access™) and related skills and abilities is **not** a predictor of final course outcomes (final overall letter grade).

H2: Student demographic and experience data from the survey and the college Banner™ database system (including gender, age, high school computer training experience, prior college computer training experience, bachelor or non-bachelor degree candidate, previous attempts of the computer literacy course and overall/current GPA) are **not** predictors of final course outcomes (final overall letter grade).

H3: Student self-assessment of skills and abilities with Windows XP™ and selected Microsoft Office™ applications (Word™, Excel™, PowerPoint™, and Access™) and available student demographic and experience data from the survey and the college Banner™ database system are **not** predictors of final course outcomes (final overall letter grade).

The course chosen for study was an introduction to computer applications course. Appendix C includes an abbreviated course syllabus highlighting topic coverage and weights. The course covers multiple modules, including hardware and operating systems, productivity applications such as word processing, database and spreadsheet software, information literacy, networking, and the Internet. A survey instrument was developed based on key topics and constructs from the literature review. The survey was reviewed and further edited by management information system faculty for completeness and accuracy.

Survey Instrument

Students reported their name, e-mail address, and various phone numbers, as well as computer courses (identified by name) they had completed in high school, college, and/or technical school. They also completed a list of other computer training and experiences. On a five-point Likert-type scale, they rated their level of knowledge on various applications from 1 (none) to 5 (expert), with points in between for novice, intermediate, and advanced. No specific definitions of these terms were given. Applications included Windows XP™, MS Word™, MS Excel™, MS Access™, and MS PowerPoint™, as well as e-mail, Internet searching, and general computer hardware/ software concepts.

The next questions focused on student's personal goals as an outcome of the class and included their expected grade and a place for open-ended comments on what they hoped to achieve and the knowledge they expected to gain. The final three pages asked students to check their specific skills for each of the applications, including the computer operating systems, word processing, spreadsheets, databases and presentation software (See Appendix D for a copy of the survey instrument). Specific skill variables for each application were developed from the textbook used in the course.

Survey Population and Sample Demographics

Self-reported data were collected from 259 students in a freshman/sophomore-level microcomputer applications and introduction to information technology course at a medium-size, AACSB-accredited state college. The course is required for all business majors and is an elective for a number of associate and bachelor's degree programs. Due to incomplete and missing data, 207 student surveys and records were used for the analysis.

Additional gathered information, as shown in Table 2 (Appendix A), included status as bachelor's or non-bachelor's degree student, overall GPA, gender, age (as date of birth), and overall course grade (all of which were gathered from the campus Banner™ database system following course completion). The age was separated to account for traditional versus non-traditional students using the breakdown used by Justice (2001) in her study of traditional and nontraditional-age college students. Justice (2001) defined traditional-age students as between 18 and 23 years of age and nontraditional-age college students as age 24 and above (through age 64). These are the age ranges used in this analysis. Individual student's scores on each individual computer package (word processing, spreadsheet, database, and operating system) were obtained from the professor of record's lab and lecture grades.

Research Design

Analysis was conducted by regressing students' course grades (GRADERCD) on their self-rated level of knowledge of Windows XP™ (WINXP), MS Word™ (WORD), MS Excel™ (EXCEL), MS Access™ (ACCESS), MS PowerPoint™ (PPT), email (EMAIL), Internet search (INTSRCH), and hardware and software concepts (HWSW); on demographic variables gender (GNDR) and age (AGE); and experience variables representing students' prior computer training in high school (HS_NONE) or college (COLL_NONE), bachelor's and non-bachelor's degree students (DEGRRCD), previous course attempt (PREVATT), and overall/current GPA. All independent variables except DEGRRCD were included because of their importance as a skill needed by students or their mention in the literature. Standard stepwise linear regression was used with criteria of probability of F to enter set at ≤ .050 and to exit ≥ .100. Three variables were found to be significant; see Table 3 in Appendix A. The resulting model is summarized in Tables 3, 4, and 5 (Appendix A).

4. FINDINGS

Since student self-assessments of Hardware/Software Concepts and MS Word™ knowledge and skills are significant in the model, **the first hypothesis is rejected**; student self-assessment of skills and abilities with Windows XP™ and Microsoft Office™ applications (Word™, Excel™, PowerPoint™, and Access™) and related skills and abilities **is** a predictor of final course outcomes.

Since GPA is significant in the model, **the second hypothesis is rejected**; student demographic and experience data from the survey and the college Banner™ database system (including gender, age, high school computer training experience, prior college computer training experience, bachelor or non-bachelor degree candidate, previous attempts of the computer literacy course and overall/current GPA) **are** predictors of final course outcomes.

Since two self-assessment items (Hardware/Software Concepts and MS Word™) and one demographic item (GPA) are significant in the model, **the third hypothesis is also rejected**; student self-assessment of skills and abilities with Windows XP™ and selected Microsoft Office™ and related applications and available student demographic and experience data from the survey and the college Banner™ database system **are** predictors of final course outcomes. The more interesting finding, however, is the small additional explanatory power associated with including Hardware/Software Concepts (adjusted R^2 increase = 0.016) and MS Word™ (adjusted R^2 increase = 0.012) compared with GPA (adjusted R^2 increase = 0.365).

The relative explanatory power of GPA and other factors is indicated in Table 6 (Appendix A). The increase in R^2 for course grade and individual computer package lecture and lab grades for GPA is greater than the combined R^2 increase for all other significant variables in all cases but one. In the one exception, Windows XP™ Lecture, GPA provides the single greatest increase in R^2 and enters the model first.

Comparing GPA and grade using the longitudinal data for this course, we find students with a 0.0 to 1.6 GPA are predicted to make an F in the course and a 1.7 to 2.1 GPA are predicted to make a D. Students with a 2.2 GPA or higher are predicted to pass the introduction to computers course with a grade C or better. GPAs of 2.8 to 3.2 are predicted to earn a course grade of B, while students with a 3.3 GPA or higher would be predicted to earn an A in the overall course.

Given the overwhelming importance of GPA in explaining course outcomes, it is probable student anxiety or "technophobia" regarding skills-based introduction to computer classes should not be an issue for good students (defined as having a high overall grade-point average). This result should be encouraging for the growing cadre of "non-traditional" or adult students returning to college, particularly in the current economic downturn. These students, who did not grow up with computers as did the traditional Millennial Generation college students of today, may feel at a disadvantage in the course or some level of stress upon entering the course. Faculty can reassure students about the similarity of the learning process in the computer course to other courses and stress that study skills and other study preparation resources are more important to course success than prior skills or perceived computer expertise.

5. DISCUSSION & AREAS FOR FUTURE RESEARCH

When assisting students with lower overall grade point averages, professors of the introduction to computers course should focus not only on course-specific skills, but on overall resources appropriate to improve students' study habits. It may be that students having problems in the course are juggling work, family, and school demands or taking too many courses. Short sessions on managing time, improving concentration, preparing to study, reading textbooks, setting goals, managing test anxiety, and improving study habits may be more important for these students. Short workshops emphasizing these skills are often offered on college campuses. Advising students with low GPAs to take one or more of these targeted workshops prior to enrolling in the introduction to computers class may be warranted.

While our study was exploratory in nature, further studies should better pinpoint the GPA range that indicates whether students will have difficulty in the course. If future analysis confirms GPA as a key predictor, then faculty may want to advise students with these lower GPAs to enroll in workshops or college success courses as a possible prerequisite. Surveys of prerequisites and remediation at other colleges and universities would be helpful in starting the dialogue.

One interesting note from the study is the negative coefficient for student-reported ability with MS Word™. This sign was unexpected but may be due to students' greater familiarity with Word™ than the other selected Microsoft Office™ applications (Excel™, PowerPoint™, and Access™). This familiarity with Word™ may lead students to overestimate their skills and abilities as compared to the other applications within the course based solely on the name recognition and general familiarity.

Further studies are needed on the design of the introduction to computers course. While beyond the scope of this study, there has been much discussion in schools of business that today's entering students may have enough experience in computer applications from high school and/or work experience to omit all or part of the course. The business community agrees students need less computer theory and more application in Windows™, Word™, Access™, Excel™ and PowerPoint™ (Spinuzzi, 2006; Wilkinson, 2006). However, the academic community continues to debate the appropriate balance of theory and application, as well as the appropriate format for the course and whether it should be continued (McDonald, 2004; Stephens, 2006). A comparative article that profiles the structure of the course at various institutions is also needed, along with further discussion in the academic community. Further study of how the course is taught and organized at other colleges and universities would also be helpful for academicians.

Further replication and extension of this study too could determine the GPA cut-off point for remediation and study skills instruction prior to attempting the introduction to computers course. This study found students with a GPA of 2.1 or lower would earn either a D or F in the introduction to computers course, with those students with a 1.6 or lower earning an F. Further studies should attempt to validate this scale.

This study found overall student GPA to be a better predictor of the final course grade than the variables in the self-reported skills inventory. GPA was also a better predictor of performance in each of the various computer skills and packages lab and lecture modules. Further research is needed with a larger sample size, across additional time periods, and with samples from a variety of institutions to confirm the findings. If the findings continue to point to GPA as a better predictor over time, then the current stream of research in self-efficacy will need to be amended, as will the focus on various individual and combinations of demographic variables as predictors of course performance. Further research is needed to determine if targeted interventions to improve overall GPA would help the overall grades in the introduction to computers course.

6. REFERENCES

Agarwal, R., Sambamurthy, V., & Stair, R. M. (2000). Research report: The evolving relationship between general and specific computer self-efficacy - an empirical assessment. *Information Systems Research, 11*(4), 418-430.

Albion, P. R. (2001). Some factors in the development of self-efficacy beliefs for computer use among teacher education students. *Journal of Technology and Teacher Education, 9*(3), 321-334.

Ballantine, J. A., Larres, P. M., & Oyelere, P. (2007). Computer usage and the validity of self-assessed computer competence among first-year business students. *Computers & Education*, *49*(4), 976-990.

Ballou, D. J. & Huguenard, B. R. (2008). The impact of students' perceived computer experience on behavior and performance in an introductory information systems course. *Journal of Information Systems Education, 19*(1), 87-97.

Bandura A. (1986). *Social foundations of thought & action: A social cognitive theory*. Englewood Cliffs, NJ: Prentice Hall.

Bandura, A. (1997). *Self-efficacy: The exercise of control.* New York, NY: W. H. Freeman.

Banister, S. & Vannatta, R. (2006). Beginning with a baseline: Insuring productive technology integration in teacher education. *Journal of Technology and Teacher Education*, *14*(1), 209-235.

Buche, M. W., Davis, L. R., & Vician, C. (2007). A longitudinal investigation of the effects of computer anxiety on performance in a computing-intensive environment. *Journal of Information Systems*, Winter, *18*(4), 415-423.

Busch, T. (1995). Gender differences in self-efficacy and attitudes toward computers. *Journal of Educational Computing Research, 12*(2), 147-159.

Busch, T. (1996). Gender, group composition, cooperation, and self-efficacy in computer studies. *Journal of Educational Computing Research, 15*(2), 125-138.

Cassidy, S. & Eachus, P. (2002). Developing the computer user self-efficacy (CUSE) scale: Investigating the relationship between computer self-efficacy, gender and experience with computers. *Journal of Educational Computing Research*, *26*(2), 133-53.

Christensen, T. E., Fogarty, T. J., & Wallace, W. A. (2002) The Association Between the Directional Accuracy of Self-Efficacy and Accounting Course Performance. *Issues in Accounting Education, 17*(1), 1-27.

Compeau, D. R. & Higgins, C. A. (1995). Computer self-efficacy: Development of a measure and initial test. *MIS Quarterly, 19*(2), 189-211.

Compeau, D. R., Higgins, C. A., & Huff, S. (1999). Social cognitive theory and individual reactions to computing technology: A longitudinal study. *MIS Quarterly, 23*(2), 145-158.

Creighton, W., Kilcoyne, M., Tarver, R., & Wright, S. (2006). Computer literacy levels of students enrolling in a post-secondary computer applications/information technology course. *Information Technology, Learning, and Performance Journal, 24*(1), 15-23.

Davis, J. L. & Davis, H. (2007). Perceptions of career and technology and training and development students regarding basic personal computer knowledge and skills. *College Student Journal, 41*(1), 69-78.

Divaris, K., Polychronopoulou, A., & Mattheos, N. (2007). An investigation of computer literacy and attitudes amongst Greek post-graduate dental students. *European Journal of Dental Education*, *11*(3), 144-147.

Gist, M. E. & Mitchell, T. R. (1992) Self-efficacy: A theoretical analysis of its determinants and malleability. *Academy of Management Review, 17*(2), 183-211.

Goh, D., Ogan, C., Ahuja, M., Herring, S. C., & Robinson, J. C. (2007). Being the same isn't enough: Impact of male and female mentors on computer self-efficacy of college students in IT-related fields. *Journal of Educational Computing Research, 37*(1), 19-40.

Hasan, B. (2003). The influence of specific computer experiences on computer self-efficacy beliefs. *Computers in Human Behavior, 19*, 443-450.

Hasan, B. (2006a). Delineating the effects of general and system-specific computer self-efficacy beliefs on IS acceptance. *Information & Management, 43*, 565-571.

Hasan, B. (2006b). Effectiveness of computer training: The role of multilevel computer self-efficacy. *Journal of Organizational and End User Computing*, *18*(1), 50-68.

Hasan, B. & Jafar, M. H. (2004). An empirical examination of a model of computer learning performance. *The Journal of Computer Information Systems*, *44*(4), 27-33.

Hawkins, B. L. & Oblinger, D. G. (2006). The myth about the digital divide. *EDUCAUSE Review*, *41*(4), 12-13.

Hindi, N. M., Miller, D., & Wenger, J. (2002). Computer literacy: Implications for teaching a college-level course. *Journal of Information Systems Education*, *13*(2), 143-151.

Hollister, K. K. & Koppell, N. B. (2008). Curricular changes in response to assurance of learning results in information technology.

Journal of American Academy of Business, *13*(1), 287-293.

Houle, P. A. (1996). Toward understanding student differences in a computer skills course. *Journal of Educational Computing Research, 15*(1), 25-39.

Hsu, W. K. & Huang, S. S. (2006). Determinants of computer self-efficacy – An examination of learning motivations and learning environments. *Journal of Educational Computing Research*, *35*(3), 245-65.

Justice, E. M. (2001). Metacognitive differences between traditional-age and nontraditional-age college students. *Adult Education Quarterly*, *51*(3), 236-249.

Karsten, R. & Roth, R. M. (1998a). Computer self-efficacy: A practical indicator of student computer competency in introductory IS courses. *Informing Science, 1*(3), 61-68.

Karsten, R. & Roth, R. M. (1998b). The relationship of computer experience and computer self-efficacy to performance in introductory computer literacy courses. *Journal of Research on Computing in Higher Education, 31*(1), 14-24.

Karsten, R. & Schmidt, D. (2008). Business student computer self-efficacy: Ten years later. *Journal of Information Systems Education, 19*(4), 445-453.

Kuhlemeier, H. & Hemker, B. (2007). The impact of computer use at home on students' internet skills. *Computers & Education, 49*(2), 460-80.

Marakas, G. M., Johnson, R. J., & Clay, P. F. (2007). The evolving nature of the computer self-efficacy construct: An empirical investigation of measurement construction, validity, reliability, and stability over time. *Journal of the Association for Information Systems, 8*(1), 16-46.

Marakas, G. M., Yi, M. Y., & Johnson, R. D. (1998). The multilevel and multifaceted character of computer self-efficacy: Toward clarification of the construct and an integrative framework for research. *Information Systems Research, 9*(2), 126-163.

Mayall, H. J. (2008). Differences in gender based technology self-efficacy across academic levels. *International Journal of Instructional Media*, *35*(2), 145-155.

McDonald, D. S. (2004). Computer literacy skills for computer information systems majors: A case study. *Journal of Information Systems Education*, *15*(1), 19-33.

Messineo, M. & DeOllos, I. Y. (2005). Are we assuming too much? Exploring students' perceptions of their computer competence. *College Teaching*, *53*(2), 50-55.

Mykytyn, P. P. (2007). Educating our students in computer application concepts: A case for problem-based learning. *Journal of Organizational and End User Computing*, *19*(1), 51-61.

Oblinger, D. G. & Hawkins, B. L. (2006). The myth about student competency. *Educause Review, 41*(2), 12-13.

Qutami, Y. & Abu-Jaber, M. (1997). Students' self-efficacy in computer skills as a function of gender and cognitive learning style at Sultan Qaboos University. *International Journal of Instructional Media*, *24*(1), 63-74.

Reed, K. D., Doty, H., & May, D. R. (2005). The impact of aging on self-efficacy and computer skill acquisition. *Journal of Managerial Issue*s, *17*(2), 212-228.

Sharkey, J. (2006). Towards information fluency: Applying a different model to an information literacy credit course. *Reference Services Review*, *24*(1), 71-85.

Shiue, Y. (2003). The effects of cognitive learning style and prior computer experience on Taiwanese college students' computer self-efficacy in computer literacy courses. *Journal of Educational Technology Systems, 31*(4), 393-410.

Spinuzzi, C. (2006). Multiliteracies for a digital age. *Journal of Business and Technical Communication, 20*(2) 225-228.

Stephens, P. (2005). A decision support system for computer literacy training at universities. *The Journal of Computer Information Systems*, *46*(2), 33-44.

Stephens, P. (2006). Validation of the business computer self-efficacy scale: Assessment of the computer literacy of incoming business students. *Journal of Educational Computing Research*, *24*(1), 29-46.

Thatcher, J. B. & Perrewe, P. L. (2002). An empirical examination of individual traits as antecedents to computer anxiety and computer self-efficacy. *MIS Quarterly*, *26*(4), 381-396.

Tien, F. F. & Fu, T. (2008). The correlates of the digital divide and their impact on college student learning. *Computers & Education*, *50*(1), 421-36.

Trine, J. A. & Schellenger, M. H. (1999) Determinants of student performance in an upper level corporate finance course. In *Allied Academies International Conference Academy of Educational Leadership Proceedings, 4*(1), 91-100.

Varank, I. (2007). Effectiveness of quantitative skills, qualitative skills, and gender in determining computer skills and attitudes: A causal analysis. *The Clearing House*, *81*(2), 71-80.

Webster, L. D. (2004). Measuring change in computer self-efficacy and computer literacy of undergraduates in an introduction to computers course. UMI *Dissertation Service.* (UMI No. 3164548).

Wilkinson, K. (2006). Students computer literacy: Perception versus reality. *Delta Pi Epsilon Journal. 48*(2), 108-20.

Wilson, T. E., Jr., Ward, S. P., & Ward, D. R. (1997). Empirical evidence regarding the use of self-reported student data in accounting education research. *The Accounting Educators' Journal, 9*(1) 50-69.

APPENDIX A

TABLE 1

Selected Research – Variables Affecting Computer Self-Efficacy and/or Performance

	Gender	Gender of mentor	Ethnic minority	Age	Cognitive learning style	Computer access and/or experience	Email use	Prior computer	Software knowledge	Blue-collar or unemployed parents	ACT Score
Busch (1995)	X										
Qutami & Abu-Jaber (1997)	X				X						
Albion (2001)						X					
Cassidy & Eachus (2002)						X		X			
Shiue (2003)					X	X					
Webster (2004)						X	X	X			
Messineo & DeOllos (2005)	X		X								
Reed, Doty, & May (2005)				X							
Creighton, Kilcoyne, Tarver, & Wright (2006)								X		X	X
Hsu & Huang (2006)						X					
Wilkinson (2006)			X								
Davis & Davis (2007)	X			X							
Divaris, Polychronopoulou, & Mattheos (2007)	X						X				
Goh, Ogan, Ahuja, Herring, & Robinson (2007)		X									
Tien & Fu (2008)	X		X						X	X	
Ballou & Huguenard (2008)						X					

TABLE 2

Sample Demographics

Variable		Total	Percent		Total	Percent
Gender	Male	94	45.6%	Female	112	54.4%
Age	Under age 24	138	66.7%	24 and older	69	33.3%
High School Computer Training Experience	None	146	70.9%	Some	60	29.1
Prior College Computer Training Experience	None	54	26.2%	Some	152	73.8%
Major	Non-Bachelor's Degree Seeking	100	48.3%	Bachelor's Degree Seeking	107	51.7%
Previous Course Attempts	None	141	68.1%	One or More	66	31.9%

Overall GPA	**< 1.00**		**1.00 to < 2.00**		**2.00 to <3.00**		**3.00 to 4.00**	
Number|%-age	2	1.0%	18	8.7%	82	39.6%	105	50.7%

Course Grades	**A**		**B**		**C**		**D**		**F**	
Number|%-age	45	21.8%	99	48.1%	53	25.7%	4	1.9%	5	2.4%

TABLE 3

Variables Entered/Removed During Stepwise Regression[a]

Model	Variables Entered	Variables Removed	Method
1	GPA	.	Stepwise (Criteria: Probability-of-F-to-enter <= .050, Probability-of-F-to-remove >= .100).
2	HWSW	.	Stepwise (Criteria: Probability-of-F-to-enter <= .050, Probability-of-F-to-remove >= .100).
3	WORD	.	Stepwise (Criteria: Probability-of-F-to-enter <= .050, Probability-of-F-to-remove >= .100).

a Dependent Variable: GRADERCD

TABLE 4

Model Summary

					Change Statistics				
Model	**R**	**R Square**	**Adjusted R Square**	**Std. Error of the Estimate**	**R Square Change**	**F Change**	**df1**	**df2**	**Sig. F Change**
1	.607(a)	.368	.365	.713	.368	118.422	1	203	.000
2	.622(b)	.387	.381	.704	.019	6.221	1	202	.013
3	.634(c)	.402	.393	.697	.014	4.846	1	201	.029

a Predictors: (Constant), GPA
b Predictors: (Constant), GPA, HWSW
c Predictors: (Constant), GPA, HWSW, WORD

TABLE 5

Regression Coefficients[a]

Model		Unstandardized Coefficients B	Unstandardized Coefficients Std. Error	Standardized Coefficients Beta	t	Sig.
1	(Constant)	.482	.226		2.129	.034
	GPA	.827	.076	.607	10.882	.000
2	(Constant)	.133	.264		.504	.615
	GPA	.841	.075	.617	11.175	.000
	HWSW	.121	.049	.138	2.494	.013
3	(Constant)	.466	.302		1.543	.124
	GPA	.833	.075	.611	11.153	.000
	HWSW	.176	.054	.200	3.246	.001
	WORD	-.136	.062	-.135	-2.201	.029

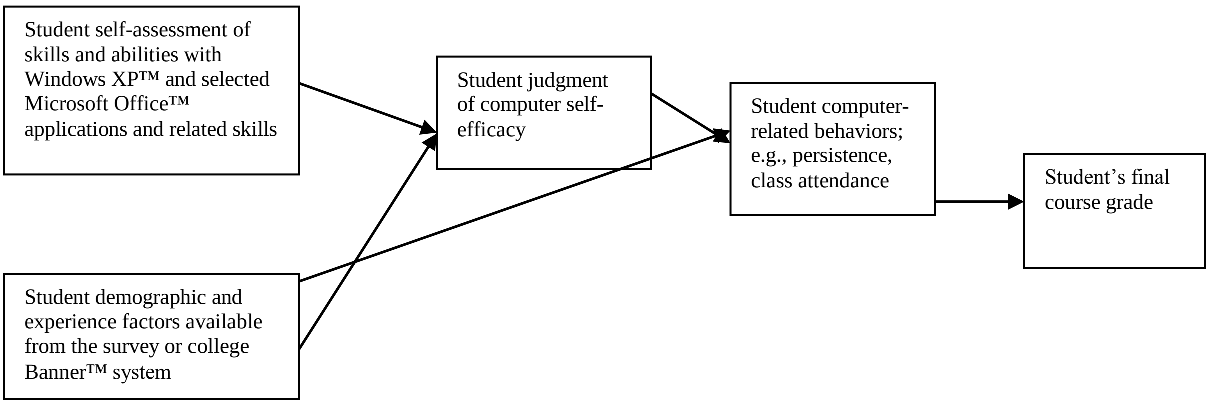
TABLE 6

R^2 Increase for Course Grade and Individual Computer Packages Due to GPA and Other Significant Factors

DEPENDENT VARIABLE	R^2 Increase GPA	R^2 Increase Sum of Other Sig. Factors	Order of GPA Entry Into Model
Course Grade	0.368	0.033	1
XP Lecture	0.151	0.175	1
XP Lab	0.049	0.000	1
Word Lecture	0.218	0.016	1
Word Lab	0.056	0.000	1
Excel Lecture	0.326	0.016	1
Excel Lab	0.158	0.000	1
Access Lecture	0.374	0.031	1
Access Lab	0.250	0.041	1

APPENDIX B

Figure 1



APPENDIX C

ABRIDGED SYLLABUS

CLASS TEXT:

Shelly, Gary B., T. J. Cashman, and M. E. Vermaat. 2008. *Microsoft Office 2007: Introductory Concepts and Techniques, Windows Vista Edition*. Thomson Course Technology: Boston.

COURSE DESCRIPTION:

Assures a basic level of computer applications literacy to include spreadsheet, database, word processing, LAN, e-mail, presentation software, and Internet utilizations. This course satisfies the computer literacy requirement.

STUDENT LEARNING OUTCOMES:

To successfully complete this course, the student should achieve the following objectives:

1. Understand how information technology aids business decision making.
2. Identify the components of a typical microcomputer system.
3. Identify and describe the most widely used general microcomputer software applications, the difference between application software and system software and understand the role of operating system software.
4. Demonstrate knowledge of computer hardware and software, including "multimedia" and be familiar with the legal, ethical, and privacy issues relating to the use of hardware and software in a business environment.
5. Be familiar with computer networks and know the basic components of a communications system to include e-mail, user interfaces, communications, and the Internet.
6. Effectively use a word processing software program, a spreadsheet program, a database management program, and develop a simple presentation using a presentation software program.

These objectives will be measured through written tests, laboratory assignments, and laboratory tests.

ASSESSMENT SCALE:

A = 90 – 100 B = 80 - 89 C = 70 - 79 D = 60 - 69 F = < 60

ASSESSMENT:

Component	Percent of total grade
Lecture tests (3-4 @100 points)	60%
Lab and Other Assignments	15%
Lab tests	25%
	100%

ATTENDANCE AND OTHER MATTERS:

Regular lecture and laboratory attendance is expected. If you miss a class, it is your responsibility to find out what you missed, including announcements of homework, lab assignments, test dates, etc. Exams are to be taken on designated test dates. No makeup tests for missed exams will be given, except in the case of extreme emergency and only with prior notification, if possible.

All assignments that have due dates are to be turned in at the beginning of the class meeting on the assigned due date. Late work will be accepted, but with a 10% penalty for each class day the work is late.

APPENDIX D

Survey Instrument

I have read, understand, and agree to abide by the policies established in this course.

Printed Name: ____________________ Signature: ______________________________

Date: __________

Please complete the following information:

Email address: ___________________________________

Phone number where you can be reached:

Day: _____________ Night: ________________ Cell Phone: ______________

Computer courses completed in:

High School: ____

College (or technical school) _________________________________

Other computer training, experience, etc.: ______________________________

Please rate your level of knowledge in each of the following:

Application	None	Novice	Intermediate	Advanced	Expert
Windows Vista	_____	_____	_____	_____	_____
MS Word	_____	_____	_____	_____	_____
MS Excel	_____	_____	_____	_____	_____
MS Access	_____	_____	_____	_____	_____
MS PowerPoint	_____	_____	_____	_____	_____
E-mail	_____	_____	_____	_____	_____
Internet Searching	_____	_____	_____	_____	_____
Computer Hardware/ Software concepts	_____	_____	_____	_____	_____

My personal goals as an outcome of this class:

expected grade: _______

what I hope to achieve: ____

knowledge I expect to gain: ____

I can perform the following activities (check all that apply):

Operating System:

____ Create text files
____ Create folders
____ Format disks with operating system (make a boot disk)
____ Format disks without operating system
____ Copy files
____ Move files
____ Create subfolders
____ Capture a screen image

Word Processing:

____ Create a document
____ Set margins
____ Set Tabs (left, right, center, dot leader)
____ Center text
____ Bold text
____ Underline text
____ Add borders
____ Add shading
____ Import graphics
____ Create tables
____ Add headers
____ Add footers
____ Create page breaks
____ Print a document
____ Show formatting marks
____ Show reveal formatting task pane
____ Customize word processor toolbars
____ Change font characteristics
____ Inserting dates (static and dynamic)
____ Create a hyperlink
____ Create a bulleted list
____ Save a document
____ Save a document as a web page
____ Create endnotes and footnotes
____ Find and replace text
____ Align text in a document
____ Align text in a table

Spreadsheets:

____ Create a new workbook
____ Select a cell
____ Enter text in a cell
____ Justify text in a cell
____ Enter numbers
____ Change font type, style, size and color
____ Save a workbook
____ Create formulas to add, subtract, multiply, and divide cell contents
Use built-in functions (e.g., SUM, MIN, MAX, AVERAGE)
____ Add shading and borders
____ Change column width and row height
____ Delete text in cells
____ Delete rows and/or columns
____ Copy cell contents
____ Move cell contents
____ Insert dates as text, as numbers, as system
____ Use relative, mixed and absolute addressing of cells

____ Create charts (both embedded and on a separate sheet)
____ Create X-axis, and Y-axis and Chart titles
____ Create an exploded pie chart
____ Name individual worksheets in a workbook
____ Make decisions using IF statements
____ Create static and dynamic web pages
____ Perform what-if analysis using
Goal-Seek

Databases:
____ Create a new database
____ Create tables
____ Create forms
____ Create reports
____ Create queries
____ Add, change, and/or delete data to/in/from a table
____ Create permanent relationships among tables
____ Enforce referential integrity
____ Specify cascade deletes and cascade updates
____ Create calculated fields
____ Specify validation rules for entering data
____ Apply filters to a query
____ Create a parameter query
____ Change the structure of a table
____ Save a database
____ Create an index
____ Create a primary key
____ Use wildcards in queries to search for certain records
____ Compact and repair a database
____ Backup a database
____ Use comparison operators to look up records
____ Use AND and/or OR operators in a query

Presentation Software:
____ Create a presentation file
____ Create slides in the presentation
____ Add graphics to the presentation
____ Create bullets
____ Change the background of the slides
____ Create animation effects
____ Add a new slide
____ Create a slide show
____ Check presentation for spelling errors
____ Create an outline
____ Print a presentation as slides, notes, and/or handouts

Please put your initials here: ____

Impact of pre-grading / resubmission of projects on test grades in an introductory computer literacy course

Thomas N. Janicki
janickit@uncw.edu

Judith Gebauer
gebauerj@uncw.edu

Ulku Yaylacicegi
yaylacicegiu@uncw.edu

Information Systems & Operations Management Department
University of North Carolina Wilmington
Wilmington NC 28403

Abstract

This research builds on the behavior learning theory that suggests a response from a student, followed by a quick feedback and another response from the student will increase student learning. An experiment was performed that allowed students to submit particular homework projects (response) early. The early submissions were graded promptly and returned to the students with comments for improvement (feedback). The students were then given the opportunity to resubmit the projects prior to the due date (another response) for final grading. Theory indicates that the students who took advantage of such a pre-grading option should do better on subsequent tests which would indicate increased learning as a result of the extra stimulus. The experimental results reported in the current paper provide partial support for the suggested increase in learning by those students who took advantage of the pre-grading option.

Keywords: pedagogy, learning theory, feedback, computer literacy

1. INTRODUCTION

In many computer literacy courses, students are required to complete homework projects, in particular projects teaching how to use office productivity software, such as spreadsheets and databases. However, students often only receive a grade and some minor comments as feedback, and no option is given to correct the errors and actually learn from the mistakes. Learning theory suggests that increased learning will occur with additional stimuli and responses (Gagne, Briggs and Wager, 1992). Even though a number of studies in the research disciplines of pedagogy and psychology have addressed the argument of increased learning through added stimuli, the validity of this theory for the teaching of basic computer skills has not yet been established. In the current paper, we investigate the impact of pre-grading/resubmission of skill teaching projects on student learning.

Over the past two years, we have used an automated grading tool in an introductory computer literacy course that is offered to business students at a regional university. The use of the automated grading tool has significantly decreased the amount of time an instructor must dedicate to grading.

The current paper emphasizes the capability of automated grading tools to support pre-grading of particular projects. First, a review of behavior learning theory is provided. It follows a review of automated grading tools and a description of the experiment with the statistical results. The paper concludes with a discussion of the results and potential extensions of the research.

2. BACKGROUND: BEHAVIORAL LEARNING THEORY

A student's success is influenced by the ability of the educator to present new information and to evaluate the student's understanding of the information. This process requires the student to learn the material covered by the educator.

Based on the behavioral learning theory, Gagne et al. (1992) proposed several principles for the effective design of instructional courses, including contiguity, repetition, and feedback. Contiguity is the concept that the **feedback** should follow the **response** without delay. The longer the delay of the feedback to a learning stimulus the less is the likelihood of correct answers to future similar questions. The principle of repetition states that practice strengthens learning and improves a learner's retention. By combining the design principles Gagne et al. (1992) developed a conceptual framework of cognitive learning that includes nine "conditions for learning":

- Gaining attention ("reception")
- Informing learners of the objective ("expectancy")
- Stimulating recall of prior learning ("retrieval")
- Presenting the stimulus ("selective perception")
- Providing learning guidance ("semantic encoding")
- Eliciting performance ("responding")
- Providing feedback ("reinforcement")
- Assessing performance ("retrieval")
- Enhancing retention and transfer (generalization")

The results of subsequent research studies suggest that of the nine conditions, eliciting performance and practice from the student ("responding") and providing adequate feedback ("reinforcement") are the conditions most directly connected to student success (Martin, Klein & Sullivan, 2007).

Similarly, Murray (1998) encouraged a teaching style based on drill/rote learning and memorization. Modules should be built with many exercises that are example driven. The principle of feedback requires that instructors inform the learner about whether the answer was correct or incorrect. In the case of an incorrect answer, feedback should include a new path to solve the problem. This new path could be a hint at the correct answer, a restatement of a prior fact, or even a new example that is less complicated (Uden & Beaumont, 2006). For instance, for an incorrect if-statement, a feedback explaining the binary nature of if-statements (true vs. false) would be appropriate. In addition, feedback that indicates that an answer is correct is just as important as feedback on incorrect answers. Confirming the correct answers would remove any doubt students might have on their newly learned skills and thus help retain the knowledge.

Responding is required from learners after they have been given sufficient material to comprehend an objective (Tomei, 2008). A related process called orientation and recall is defined as a process where learning involves the synthesis of prior information that must be recalled to short term memory (Uden & Beaumont, 2006). Similarly, there is a school of thought that learners construct knowledge by making sense of experiences in terms of what is already known (Eugenia, 2010).

When practice is included in a lesson, performance implies an active response by the student to the material provided. For example, in a database lesson, responding might require a student to create a query that counts the number of records in a table in order to demonstrate his/her comprehension of the newly introduced concept.

Responding enables the student to reinforce his/her understanding. Effective practice should parallel the assessments that will be used to test skills and the knowledge reflected in an objective (Reiser & Dick, 1996).

Building on on Gagne et al.'s (1992) results that response and reinforcement are key learning components, the current study investigates whether a focus on these key components can be helpful in teaching hands-on skills more effectively.

3. AUTOMATED GRADING TOOLS

Automated grading systems are provided by a number of textbook publishers, among others. Key advantages of automated grading include:

- Reduced lag time between submission of a project by a student, and response in the form of a grade or other feedback to the student
- Application of a grading rubric for a project that is consistent for all students
- Capability to add assignments as the grading time per project has been reduced.

Indeed, the results of previous research studies suggest that the use of automated rubrics can result in faster and increased feedback, and that systems may be of advantage to instructors (Tan 2009; Anglin, Anglin, Schumman and Kalinski 2008; and Debuse, Lawley and Shibl 2007). Similarly, Janicki and Steinberg (2003) suggested the need for increased computerized support for learning. Heinrich, Milne, Ramsay and Morrison (2009) demonstrated how e-tools can be used to increase the efficiency and quality of assignment making.

Examples of automated grading systems include case-based auto graders and procedural-based graders.

Case-based Auto Graders

An example of a case-based auto grader is CASEGRADER by Thomson Course Technology (Crews and Murphy 2008). Instructors are provided with a set of cases that can be instantly graded. This type of system offers challenging, multi-step, realistic problems that students may submit to be automatically graded. Feedback is instantaneous and based on incorrect responses. Students are informed of their grade and provided feedback immediately following their submission of an assignment. One major limitation of this system includes the inability of instructors to create their own cases (Crews & Murphy, 2008). For example, for the Office 2007 release, CASEGRADER offers a total of twelve different cases. If multiple sections of a course use the same limited set of cases, an increase in student plagiarism could occur.

Procedural-based Graders

Procedural-based graders include systems such as SAMS2007 (2007) by Thomson Course Technology and SNAP by EMC Paradigm Publishing (2007). These alternative systems are applications that grade student responses based on the procedure used to reach the answer. The application may either be a web system or a software application that simulates the environment of Microsoft Office programs in order to provide a hands-on experience for the students. These systems usually incorporate smaller problems that attempt to reinforce a procedure to be remembered. Few complex problems exist in the database of questions for these graders.

4. PRE-GRADING WITH A CUSTOM-BUILT AUTOMATED GRADER

Adaptive Grading and Learning System

In order to meet the specific needs of students and instructors at a regional public university, a customized grader was developed and implemented in the fall of 2008. Known, as the Adaptive Grading/Learning System (AGLS), the system consists of modules that provide automated grading of Microsoft Excel 2007 and Access 2007 assignments with personalized and rapid feedback, assignment libraries that can be shared among participating instructors, and plagiarism detection. In addition, the system allows increasing the complexity of exercises without much additional effort by the instructor. This increase in complexity serves to challenge students and increases the likelihood of learning success.

One result of the availability of the AGLS to instructors was a notable increase in the number of assignments that are given in class due to a significant decrease in grading time. For example, the instructor of one section of the computer literacy course now requires twelve different assignments, versus five projects that were required prior to the implementation of the AGLS four semesters ago. According to behavior learning theory, more responses from students should be associated with more learning.

Pre-grading

Following the introduction of the AGLS, some instructors gave students the opportunity to submit their projects in advance of the due date for one (or even several) round(s) of pre-grading. After a project was graded and specific comments were posted to the student's grade book on the web, the student could resubmit the project for final grading.

It should be noted that the comments provided to the students did not give them the solution but rather pointed to what needed to be corrected. Examples include:

- Excel: Incorrect formula in B17
- Excel: Missing IF in C24
- Excel: Absolute reference in D22
- Excel: Incorrect use of the SUM function
- Access: Primary key incorrect in table 'Customers'
- Access: Field type incorrect for zip code
- Access: Query Invoices, criteria for past due invalid

The practice avoided students turning in a project basically blank and the automated system giving them the correct formulas or criteria.

Methodology, Data Gathering and Analysis

For the current study, experimental data was gathered from one section of eighty-seven students in an introductory information systems course. By selecting only one section taught by the same instructor the experiment avoided differences due to different instructor content, teaching styles and assignments.

Assignment	Number of student who submitted early
Access (basic table and query design)	76 out of 87
Excel (basic IF's)	66 out of 87
Excel (Solver)	61 out of 87

Table 1: Pre-grading submissions

Over one semester, students were given the opportunity to submit three assignments for pre-grading. Students only had to submit the assignment a reasonable time prior to the due date to get feedback and an opportunity to resubmit. Table 1 details the number of students who took advantage of the pre-grading opportunity. The order of the assignments in Table 1 is the order in which they were assigned during the semester.

Two observations may be derived from Table 1. First, the number of students who submitted projects early was rather high. It was a very favorable observation that 85% of students submitted the first project early; and even at the end of the semester 70% of students submitted for pre-grading. The instructor expected the pre-grading rate to be lower.

The second observation is less surprising: the number of students who submitted early decreased over the semester. The decrease can be interpreted such that as more work in other courses became due, students tended to complete their assignments closer to the due date.

To test if additional learning occurred for those students who took advantage of the pre-grading opportunity, the following hypothesis was developed:

H_0 – Pre-grading will not be associated with higher student scores

H_1 – Pre-grading will be associated with higher student scores

In addition to the homework projects in Access and Excel (Table 1), four tests were administered during the semester. Each of the tests had two components: a multiple choice/short answer component and a hands-on component that tested the literacy skills covered in the previous weeks (i.e., Access and Excel). Pre-grading opportunities were available prior to three out of the four tests.

For each test, student data was divided into two groups based on whether a student had taken advantage of the pre-grading option or not:

- Experiment Group: took advantage of pre-grading prior to the test
- Control Group: no pre-grading prior to the test

For each test the population of the groups differed, based on who had taken advantage of the related pre-grading opportunity. Reflected in the numbers is, thus, the decreasing number of students who took advantage of pre-grading over the semester (Table 1).

To eliminate any bias due to the differences of a student's prior knowledge or motivation, relative instead of absolute test scores were used. This measure also eliminates the potential that those students who submitted projects early were more motivated or more intelligent. Specifically, the difference between the scores of the multiple choice component and the hands-on component of the test for each student was used as the data-basis. For example:

Experiment Group Student 1:
Multiple Choice Test Score: 85
Hands on Test Score 91
Difference: 6

Control Group Student 1:
Multiple Choice Test Score: 85
Hands on Test Score 87
Difference: 2

Thus, in this example the experiment student scored 6 points higher on the hands-on component of the test while the control student scored 2 points higher. An analysis of the means was performed to determine whether the differences between the two groups were statistically significant.

Results and Discussion

For all cases a two tailed t-test was run assuming normality of the data. For two of the cases the variances test yielded unequal variances and thus a modified t-test was run (Table 2).

Cases	p-value	Variance
Access Hands On	.114	Unequal
Excel IF's	.524	Unequal
Excel Solver	.012	Equal

Table 2: p values from t-tests

As is summarized in Table 2, the results of the t-test analysis suggest that for two out of three assignments, the difference in learning as a result of pre-grading is NOT significant at the 5%-level. For the third assignment (Excel Solver), however, a difference is found that is highly significant at 1.2%. In all, the support for learning theory as a result of our experiment appears to be mixed.

A closer look at the data provides some additional insights and support for our hypothesis H_1. The first test (Access Hands-On) has a p-value of .114 which indicates acceptance of the null hypothesis; however, this result is close to a .10 p-value that can in fact be accepted for experimental research. One explanation for the highly insignificant t-test in the case of the second assignment/test may be that, while pre-grading concentrated on IF-statements, the test was actually on Excel Scenarios. IF-statements can be used in Scenarios, but are not necessarily included in the building of scenario cases that students often find difficult. Thus this hands-on test did not fully match the pre-grading assignment.

Table 3 compares the average results of the control and experiment groups and computes the difference between the multiple choice and hands-on components for the entire section. The results support our H_1, as they suggest that there might in fact be a gain in learning from pre-grading: For the experiment group, the difference between the multiple-choice and the hands-on components of the test is larger (5.72) than for the control group (4.22). Students who took advantage of pre-grading performed particularly well in the hands-on component of the test when compared with the multiple-choice component. However, the difference was, again, not statistically significant the 5%-level.

	Control	Experiment
Avg Multiple Choice	82.04	86.06
Avg Hands On	86.27	91.8
Difference	4.22	5.72

Table 3: Test score means and differences, both groups, all assignments combined

An unanswered question is whether the scores on the multiple choice tests (which are higher for the experimental group) are not just a sign of more motivated students, but also a sign of learning of additional concepts from re-doing projects that then helped in answering questions on the multiple choice portion of the test. The effects of concept learning may thus have had an additional impact on the relative differences between the results of projects and tests and the levels of significance (or lack thereof).

6. LIMITATIONS AND FUTURE RESEARCH

One limitation of the current research setup resulted from the fact that the hands-on portion of the second test did not exactly match the concepts that were included in the pre-grading assignment. More specifically, pre-grading focused primarily on IF-statements while the test included scenario management skills in addition to IF-statements. In addition, there may have been an impact of concept learning from re-doing assignments.

To overcome these limitations, the following research is currently in progress:

A) Matching the concepts on the pre-graded assignments with the hands-on portions of the multiple choice tests; and

B) A re-examination of the multiple choice tests to eliminate the impact of concept questions about Excel or Access on the final scores. This will permit a less biased analysis of the data.

7. SUMMARY AND CONCLUSIONS

In summary, the results of the experiments partially supported the research hypothesis, and, thus provided limited support for the behavior learning theory that a response solicited from a student, followed by rapid feedback and then another response would increase student learning. Rapid feedback was assumed since all projects were graded within 48 hours of submission. In particular in the case in which the test concepts matched closely the hands-on concepts of the pre-grading project (Excel Solver), the results were highly significant, suggesting that learning did occur as a result of the pre-grading option.

As a side-effect of the experiment, it was encouraging to see how many students took advantage of the pre-grading option, thus increasing their chance for learning. Another positive result of the pre-grading experiment was a noticeable reduction in the 'arguments' from students on grading. Since students were given the option to re-submit their projects, they did not argue over small grading questions. Whereas if the first grading had been final they might argue that =SUM(B3, B4, B5, B6) was a valid answer since it resulted in the correct value on the spreadsheet. The pre-grading option permits students to fix formulas that might have yielded the correct value, but were not considered the correct answer according to the learning objectives.

The knowledge gained from this study provides valuable insights for instructors, particularly those teaching online web-based courses as such environment lacks the direct observation of learning during physical lab meetings.

8. REFERENCES

Anglin, L., Anglin K., Schumman, P., & Kalinski, J. (2008). Improving the efficiency and effectivement of grading through the use of computer assisted grading rubrics. *Decision Sciences Journal of Innovative Education*, 6 (1), 51-73.

Crews, T., & Murphy, C. (2008). *CASEGRADER: Microsoft Office Excel 2007 Casebook with Autograding Technology*. Boston, MA: Thomson Course Technology.

Debuse, J., Lawley, M., & Shibl, R. (2007). The implementation of an automated assessment feedback and quality assurance system for ICT courses. *Journal of Information Systems Education*, 18 (4), 491-502.

Eugenia, M. (2010). *Comparative Blended Learning Practices and Environments*. Hershey, PA. IGI Global.

Gagne, R., Briggs, L. & Wager, W. (1992). *Principles of Instructional Design* (4th Ed.). Fort Worth, TX. HBJ College Publishers.

Heinrich, E., Milne, J., Ramsay, A., & Morrison, D. (2009). Recommendations for the Use of E-tools for Improvements around Assignment Marking Quality. *Assessment & Evaluation in Higher Education*, 34 (4), 469 - 479.

Janicki T., & Steinberg, G. (1999). "Improving student satisfaction in large-sized classes through a computer-mediated communications system," *Journal of Information Systems Education,* 10(2), 8-13.

Martin, F., Klein, J. & Sullivan, H. (2007). The Impact of Instructional Elements in Computer-Based Instruction. *British Journal of Educational Technology, 38* (4), 623 - 636.

Murray, T., (1998). Authoring Knowledge Based Tutors: Tools for Content, Instructional Strategy, Student Model and Interface Design. *Journal of the Learning Sciences*, 7 (1), 5-64.

Reiser, R. & Dick, W. (1996). *Instructional Planning: A guide for teachers* (2nd Ed.). Allyn and Bacon Publication, Columbus OH.

SAMS2007 (2007). *Course Technology - SAM 2007 Training V1.0*. Available (last checked 3/17/2010): http://www.course.com/catalog/product.cfm?isbn=978-1-4239-1305-4

Tan, C. (2009), Assessment via WebCT Quizzes: Offline Grading Process with Customized Feedback, *Decision Sciences Journal of Innovative Education, 1* 321-326.

Tomei, L.A. (2008). *Encyclopedia of Information Technology Curriculum Integration*. Hershey, PA. IGI Publishing.

Uden, L. & Beaumont, C. (2006). *Technology and Problem-Based Learning*. Hershey, PA. IGI Publishing.

Design, The "Straw" Missing From the "Bricks" of IS Curricula

Leslie J. Waguespack
lwaguespack@bentley.edu
Computer Information Systems Department, Bentley University
Waltham, Massachusetts 02154-4705, USA

Abstract

As punishment in the biblical story of Moses the slaves were told they had to make bricks without straw. This was impossible because bricks made without straw had the appearance of strength and function but could not withstand the proof of actual use. The slaves' punishment was therefore not only to make bricks, but also to find the straw on their own with which to make them. In this day and age it would seem that many of our Information Systems curricula ask students to learn to make systems without teaching them about design. We are good at teaching students how to make software systems that do things but not so good at teaching students how one way of doing things in a system design is better than another. In this essay I consider the role of teaching systems design in preparing an IS professional and the forces that have come into play over the history of computing that have, in many cases, frozen out the study of design from the IS curricula.

Keywords: design, IS discipline, IS curricula

1. INTRODUCTION

As computing education embarks on its eighth decade of preparing the professionals who will build information systems supporting every facet of humankind's culture and commerce, the specialization of computing curricula has subdivided and compartmentalized the principles, science, and practice of computing into five general categories: computer science, computer engineering, software engineering, information technology and information systems (Shackelford, Cross, Davies, Impagliazzo, Kamali, LeBlanc, Lunt, McGettrick, Sloan & Topi, 2005). Without question the breadth of all the knowledge encompassing computing today is too large to be addressed to any significant depth in a computing student's undergraduate education. Reason and practicality dictate that the knowledge of computing be subdivided (aka. specialized) in practice-focused curricula. This essay explores the proposition that one *practice* essential to any form of computing, *design*, has been sidelined (if not virtually forgotten) in computing's curricular subdivision. This paper examines the disciplinary evolution of computing and the most recently published guidelines for computing curricula. I consider whether design education is sufficiently represented in their prescriptions and focus specifically on information systems education.

2. THE EVOLUTION OF COMPUTING FROM PROGRAMS TO SYSTEMS

In the early years of computing (1938 - 1958) computer systems (analog computers particularly) were capable of working on the solution of only a single problem at a time. This single-mindedness of function meant that computers were indivisible resources that could not be shared except through sequenced allocation (Green, 2010). Digital computing eventually revealed the opportunity to use the natural differential between the processing speeds of various computing components (i.e. I/O vs. computation usually resulted in idle time for the computation units) to multiplex tasks and recover time otherwise lost waiting for slower operations.

In that era the primary design challenge was bridging the conceptual distance between human requirements and computing functionality. Success most often depended upon the ability of designers to reshape their problems to accommodate the computer's capabilities.

The transition from running a single stream of sequential "work" through a computing resource into the coordination of multiple (seemingly) concurrent streams of "work" more closely approximated the real world of organizations and life but also introduced the challenges of workflow management (coordination, prioritization, dependency, and planning). What here-to-fore may have been challenges of resource utilization optimization for individual programs became optimization for application systems.

Although the dramatic growth of computing power and resources (e.g. virtual memory, parallel processing, multiprogramming, and multitasking: 1958-1975 (Blaauw & Brooks, 1997)) may have obviated detailed study of operating systems principles for application programmers, the same principles of problem solving remain critical because they (coordination, prioritization, dependency, and planning) had become the critical resource management issues at the service oriented application level of systems!

3. THE WIDENING BREADTH OF TECHNICAL INFORMATION IN COMPUTING CURRICULA

For the first generation of information system builders in the digital age (1956 - 1968) the patterns and recognition of software design quality in programming were learned / developed through countless repetitions of programming exercises across three or more programming languages (i.e. assembler, FORTRAN, COBOL). This included problems from the trivial (to learn syntax) to the more complex approaching application system complexity.

The paucity of pattern enforcing mechanisms in programming tools (languages, editors, compilers, debuggers, etc.) required successful developers to be vigilant as they wrote software: crafting modularity, transparency, traceability, and maintainability – the selfsame characteristics that in concert condition a holistic mindset on the design quality of systems. In particular traceability testified to the conceptual integrity of a design's pertinence as a "solution" to the problem.

Underlying structural software concepts received individual focus in coursework that isolated data structures, control structures, communications, module, and systems architecture (at various levels) more or less independent of any particular modeling or programming dialect.

Structured programming was the first overarching model to organize basic design principles of coding into a paradigm of do's and don't 's that focused on achieving qualities of clarity, reliability and transparency in code (Dijkstra, 1968).

In these first couple decades of computing removed from the research laboratories into the university classroom, the breadth of concepts and practice in computer science and computer engineering did not yet outstretch the capacity of an individual's awareness of issues and topics across the entire field.

4. THE EXPANSION OF COMPUTING'S APPLICATION SPACE FROM SCIENTIFIC TO COMMERCIAL

In the advent of digital computing (1950-1965) only a handful of organizations had access to any form of problem solving using "mechanical computation." Those organizations were resource-privileged either because of their governmental or financial power. As a result, the professionals involved in learning and employing these tools were recruited from the same ranks as those who were sought for research in mathematics, engineering and the sciences. Academia's response to the resource requirement for education of these professionals followed the same pattern as that found in mathematics, engineering and the sciences with heavy doses of foundational coursework including broad coverage of basic theory followed by extensive review of the current research in digital computation, electronic circuitry, hardware and software architecture (which usually meant reviewing the dozen or so contemporarily predominant computer designs).

As it became commercially feasible to offer computing systems within the financial means of more and more commercial customers, the demand for information systems development exploded. Professionals were needed to develop and manage computers in more far-flung application domains (business, medicine, applied engineering, etc.) in which computing's primary purpose was augmenting the existing culture of systems and problem representation in a domain. This prompted academic programs in those domains to introduce application domain-based computing education. Those programs naturally treated computing as an addendum to their "core" disciplinary foci. The subset of computing knowledge that was incorporated narrowed down to a treatment of application development. In most cases these applications were seen as generally isolated solutions to individual and separate applications of problem solving.

5. THE GROWTH OF FACADE-BASED APPLICATION DEVELOPMENT ENVIRONMENTS

Marked by a steady increase in connectivity and the coming of the Internet over the last three decades, the breadth of applying computing to more and more commercial opportunities for problem solving has swelled. Tools for application development have evolved to insulate developers more and more from the details and intricacies of the computing platforms and environment. At the same time application development has expanded to an ever-broadening population of "developers" less and less versed in the core fundamentals of computing theory and practice. Indeed business computing as confined to the collection, organization and reporting of data has evolved into more of a clerical activity as opposed to one of problem solving. Quite reasonably, as a proportion of ongoing business computing activities, "data processing" predominates.

Because of this dominance, technical education in computing activities has migrated from departments of mathematics and engineering to departments applying computing to their domain-based interests. And to the extent the academic programs focus on teaching best practice <u>using</u> applications of known solutions to domain-based problems, they serve their students well. But, the ever-increasing interconnectedness of information and processes has levied a new layer of complexity upon collaboration and adaptability. More than ever computing capabilities are changing "the existing culture of systems and problem representation in a domain." The challenges arise at the frontier of known solutions where either the reshaping of the domain-based problem or the creation of innovative applications of computing require more than the mastery of off-the-shelf solutions – they require creative design. They require systems that integrate the people, policies, information, hardware, software, networks, and quality management in the design of complete, holistic solutions. They require systems that accomplish a conceptual integrity and enlightened design (Brooks, 2010).

6. THE CONFINING PEDAGOGICAL RESOURCE - CURRICULAR TURF

When we consider domain-based education (business, medicine, applied engineering, etc.) combined with the fundamentals of computing and systems, the inventory of prospective, relevant coursework quickly exceeds the course credit hour "budget" of any undergraduate curriculum. Under this pressure the balance of emphasis and the share of the curricular coursework naturally tilts in the favor of the domain-based disciplines and away from the depth of fundamental computing theory and practice needed to fuel innovation and enlightened design. This has clearly been the case in computing programs contained in schools of business naturally preoccupied with certifying their "business" credentials [AACSB 2010, EQUIS 2010]. The footprint of coursework assigned to a business computing major is seldom more than 24 course credit hours dedicated to computing.

7. WHAT DESIGN IS ABOUT

The New Oxford American dictionary defines *design* (noun) as a plan representing the form and function of something before it is built or made. Design engenders the purpose, planning or intention that exists or is thought to exist behind an action, fact or material object.

Over the last 50 years Fred Brooks has been one of the most ardent and influential advocates of design as essential to the pursuit of information system quality.

"Whereas the difference between poor conceptual designs and good ones may lie in the soundness of design-method, the difference between good designs and great ones surely does not. Great designs come from great designers. Software construction is a creative process. Sound methodology can empower and liberate the creative mind; it cannot inflame or inspire the drudge.

The differences are not minor – they are rather like the differences between Salieri and Mozart. Study after study shows that the very best designers produce structures that are faster, smaller, simpler, cleaner and produced with less effort. [...] The differences between the great and the average approach an order of magnitude." (Brooks, 1995)

In his most recent reflection on the professional practice of creating information systems that support organizational goals he comments on the central role of design in this way.

"The essentials of [design] are *plan*, *in the mind*, and *later execution*. Thus a design (noun) is a created object, preliminary to and related to the thing being designed, but distinct from it."

"A book, in this conception, or a computer, or a program, comes into existence first as an ideal construct, built outside time and space, but complete in essence in the mind of the author. It is implemented in time and space, by pen, ink, and paper; or by silicon and metal. The creation is complete when someone reads the book, uses the computer, or runs the program, thereby interacting with the mind of the maker." (Brooks, 2010)

Brooks clearly distinguishes the act of system *design* from the *implementation*. The cycle of system creation differentiates *design*, *implementation* and *use*, but it does not segregate them! Indeed their interdependency is core to understanding each aspect as declared in the agile development concept. (Beck, 2010) Although distinct, these elements of system creation fuse as they conceive, develop and judge the design qualities that mark the degree of satisfaction (success) the stakeholders experience during a system's lifetime.

This distinction between design and implementation has faded from the structure of computing education. To ignore the conceptual distinction between the *design* and an *implementation* is tantamount to accepting any "solution" without even considering whether (as Brooks declares compared to the "average") there is a solution out there that is an order of magnitude "faster, smaller, simpler, cleaner and produced with less effort."

8. CURRICULUM GUIDELINES – IN SEARCH OF DESIGN

Finding the latest focus on design in computing curricula starts with The Overview Volume on Undergraduate Degree Programs in Computing. The CC2005 report is the de facto definition of subdivisions of computing education (see Figure 1 in the appendix).

As the report declares "We have created this report to explain the character of the various undergraduate degree programs in computing and to help you determine which of the programs are most suited to particular goals and circumstances." (Shackelford et. al., 2005)

The CC2005 report explains the general evolution of computing curricula depicted in Figure 2 (see the appendix).

Among the 39 Knowledge Areas of computing identified in CC2005 only 7 reference design as a specific professional competency in any form. Among those the area definitions in the glossary do not distinguish between *design* and *implement*. To some extent this is not surprising since the CC2005 effort was primarily conceived to contrast the foci of the 5 computing subdivisions rather than explain them in detail. To get detail we must explore each of the five subdivision curriculum guideline documents: CE, CS, SE, IT and IS. (Soldan, Hughes, Impagliazzo, McGettrick, Nelson, Srimani & Theys 2004; Cassel, Clements, Davies, Guzdial, McCauley, McGettrick, Sloan, Snyder, Tymann & Weide, 2008; Diaz-Herrara & Hilburn, 2004; Lunt, Ekstrom, Gorka, Hislop, Kamali, Lawson, LeBlanc, Miller & Reichgelt, 2008; Topi, Valacich, Wright, Kaiser, Nunamaker, Sipior & de Vreede, 2010)

All 5 curriculum guideline documents liberally refer to *design* in various applications of technology to systems development. However, only the software engineering curriculum guidelines address specific aspects of design quality or design principles in its knowledge area content (Diaz-Herrara et al, 2004). Indeed only the software engineering guidelines imply to any degree that design is a separate conceptual or practical activity distinct from implementation. There are no learning unit designations in the IS 2010 curriculum guidelines addressing aspects of design distinct from a technology.

This is the case because current practice in IS curricula has assumed that teaching any form of *implementation* suffices for teaching *design*. When *implementation* was taught across several courses and languages in earlier days of computing curricula, extensive *implementation* may indeed have sufficed for *design*-focused pedagogy. In an IS curriculum today, when it is almost impossible to find room for more than two or three courses in any systems development technology or more than a single course in any particular technology, teaching *implementation* cannot suffice for teaching *design*. If these current challenges weren't severe enough, IS 2010 no longer lists *implementation* (application development) as a core requirement. With that "juridical" justification removed IS and CIS programs may find it even harder to maintain any semblance of practical system life cycle pedagogy.

9. CONSEQUENCES OF TEACHING "BRICKS WITHOUT STRAW"

De-emphasizing *design* in IS curricula results in the narrowing of the learning experience toward *talking about* systems rather than *forming* systems. Here the term "forming systems" is not limited to "writing program code," but includes developing requirements, modeling information, processes and transactions, as well as building application software. *Design* permeates the forming of systems – even if only to describe them (Waguespack 2010). *Design* is the fundamental problem-solving aspect of systems. *Design* is the foundation and justification of systems and is essential to understanding them.

The most prominent consequence of de-emphasizing *design* in IS curricula is the effect it has on IS graduates' employment opportunities. Graduates of an "about-IS" focused academic program are increasingly challenged to justify to themselves and to employers their value over graduates in the business domain without an IS degree. It is increasingly difficult for an employer to distinguish the hiring advantage of a business student with an IS major over that of an IS minor or general business graduate. Where IS programs share a college with accountancy, marketing, management, finance, etc., these programs have successfully co-opted interest in IS to their programs by offering courses focused exclusively on the use of discipline-based, extant application systems - avoiding systems development completely. As a result, unable to clearly promote the career advantages of an IS degree over "general business," IS programs find it increasingly difficult to recruit IS majors.

10. WHAT THE FUTURE MAY HOLD

Whether Information Systems is or is not a discipline has long been the subject of debate in the field of computing. This can be evidenced by the search for labels in the field: DP, IS, MIS, CIS, and IT. Clearly IS first emerged at the intersection of computer science, business, management and (many would say) engineering. Over the past two or three decades many IS programs have devolved by de-emphasizing the construction aspects of their curricula; effectively jettisoning merged content from computer science and engineering in the process.

This essay contends that the primary loss in this devolution has not been "coding skill" in some particular programming language. The loss is the aspect of design as a holistic mindset and the tools it provides in shaping IS problem representation and problem solving – applying computing in the information and organizational contexts (Denning 2004) and reinforcing "systems think" (Waguespack 2010). This loss negatively impacts the students' ability to understand requirements and formulate models of software, models of business, and models of business process. In IS, design is the act of fusing technological opportunity with business opportunity often reshaping or reinventing both. Absent design, computing assumes the status of a contraption that one might take off the shelf as-is, surrendering the solution quality to the purposes of others – basically surrendering innovation to the appliance manufacturers. If the trajectory of this evolution continues I believe the debate will be over and IS as a discipline will indeed be no more.

The challenge is no simple one. If Information Systems is to maintain its valid role as the bridge between computing and the effective / efficient application of technology to information and process problems, IS curriculum architects must find a way to re-energize the teaching of *design* in their programs. In many institutions business programs are limited to prerequisite chains no longer than two courses. That makes it unlikely that renewed emphasis can be gained simply by adding courses to existing program structures. Some renewed energy may be gained through creative pedagogy by introducing systems building activities into more theoretical IS study (e.g. computer organization, networking, project management or policy). Such a creative reorganization of learning activities will surely require extensive investment in textbook and laboratory redirection. In some cases this will require the reversal of the IS-diffusion among business departments. In other cases it may require the inventive re-structuring of curricula that bridge departments of IS and computer science to take broader advantage of arts and sciences elective opportunities across the university.

In any case, the time is relatively short for reversing the decline of IS's relevance as an academic discipline. Remarkably, as few as the number of graduates from most IS programs there are, they are highly sought-after, and the employment market for them has weathered major storms of off-shoring and economic downturn. These are indications that society (particularly business) still needs practically educated professionals who understand both the application domain and computing, and combine that knowledge and skills to deliver tomorrow's quality, innovative information systems. How will IS programs and higher education respond?

11. ACKNOWLEDGEMENTS

Thanks to helpful referees. Special thanks are due my colleagues in Computer Information Systems at Bentley University for their insightful discussions and comments on these ideas.

12. REFERENCES

AACSB (2010). *Eligibility Procedures and Accreditation Standard for Business Accreditation*. Retrieved July 16, 2010 from http://www.aacsb.edu/accreditation/AAACSB-STANDARDS-2010.pdf

Beck K., Beedle M., van Bennekum A., Cockburn A., Cunningham W., Fowler M., Grenning J., Highsmith J., Hunt A., Jeffries R., Kern J., Marick B., Martin R.C., Mellor S., Schwaber K., Sutherland J., & Thomas D. (2010). *Manifesto for Agile Software Development*. Retrieved July 12, 2010 from agilemanifesto.org

Blaauw, G.A., & Brooks, F.P. (1997). Computer Architecture: Concepts and Evolution. Addison-Wesley, Reading, Massachusetts.

Brooks, Frederick P. (1995). The Mythical Man-Month: Essays on Software Engineering (2ed). Addison-Wesley, Boston, MA.

Brooks, Frederick P. (2010). The Design of Design: Essays from as Computer Scientist. Addison-Wesley, Pearson Education, Inc., Boston, MA.

Cassel L., Clements A., Davies G., Guzdial M., McCauley R., McGettrick A., Sloan B., Snyder L, Tymann P., & Weide B.W., (2008). Computer Science Curriculum 2008 An Interim Revision of CS2001. Association of Computing Machinery (ACM), & IEEE Computing Society (IEEE-CS).

Dijkstra, E. (1968). GOTO Statement Considered Harmful. *Communications of the ACM*, 11(3), 147-148.

Diaz-Herrara, J.L., & Hilburn, Thomas B. (eds.) (2004). Software Engineering 2004: Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering, IEEE Computing Society (IEEE-CS), Association of Computing Machinery (ACM).

Denning, P. J. (2004). The Great Principles of Computing, *Ubiquity*, 4(48), 4–10.

EQUIS (2010). *EQUIS Standards and Criteria*. Retrieved July 16, 2010 from http://www.efmd.org/attachments/tmpl_1_art_041027xvpa_att_080404qois.pdf

Green, T. (2010). Bright Boys. A.K. Peters, Ltd., Natick, Massachusetts.

Lunt, B.M., Ekstrom, J.J., Gorka, S., Hislop, G., Kamali, R., Lawson, E., LeBlanc, R., Miller, J., & Reichgelt, H. (eds.) (2008). Information Technology 2008: Curriculum Guidelines for Undergraduate Degree Programs in Information Technology, Association of Computing Machinery (ACM), IEEE Computing Society (IEEE-CS).

Shackelford, R., Cross, J.H., Davies, G., Impagliazzo, J., Kamali, R., LeBlanc, R., Lunt, B., McGettrick, A., Sloan, R., & Topi, H., (2005). Computing Curricula 2005: The Overview Report, Association for Computing Machiner (ACM), The Association of Information Systems (AIS), The Computer Society (IEEE-CS).

Soldan, D., Hughes, J.L.A., Impagliazzo, J., McGettrick, A., Nelson, V.P., Srimani, K., & Theys, M.D. (eds.) (2004). Computer Engineering 2004: Curriculum Guidelines for Undergraduate Degree programs in Computer Engineering, IEEE Computer Society (IEEE-CS), Association for Computing Machinery (ACM).

Topi, H., Valacich, J.S., Wright, R.T., Kaiser, K.M., Nunamaker, J.F. Jr., Sipior, J.C., & de Vreede, G.J. (eds.) (2010). IS2010: Curriculum Guidelines for Undergraduate Degree Programs in Information Systems, Association for Computing Machinery (ACM), Association for Information Systems (AIS).

Waguespack, L. J. (2010). Thriving Systems Theory and Metaphor-Driven Modeling. Springer, London, U.K.

Editor's Note:

This paper was selected for inclusion in the journal as an ISECON 2010 Meritorious Paper. The acceptance rate is typically 15% for this category of paper based on blind reviews from six or more peers including three or more former best papers authors who did not submit a paper in 2010.

Appendix

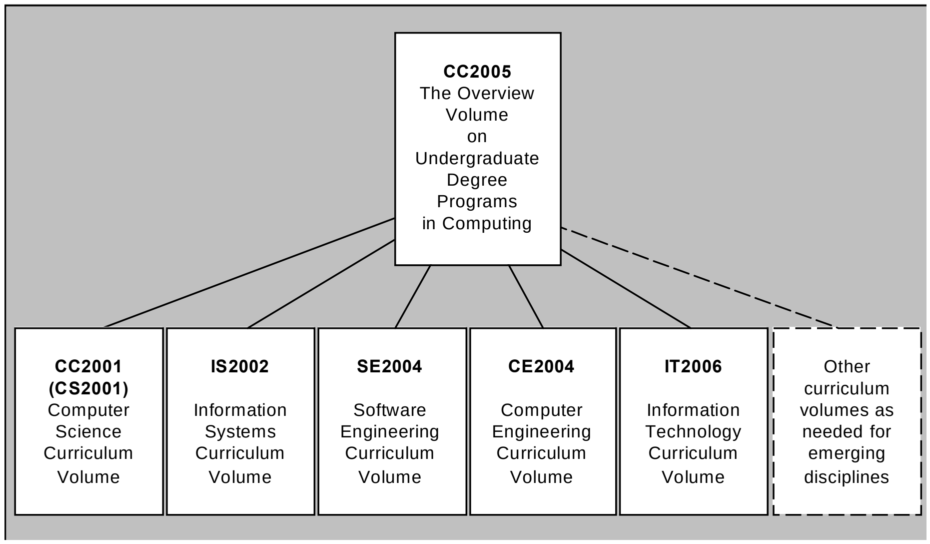


Figure 1 - Computing Curricula Guidelines

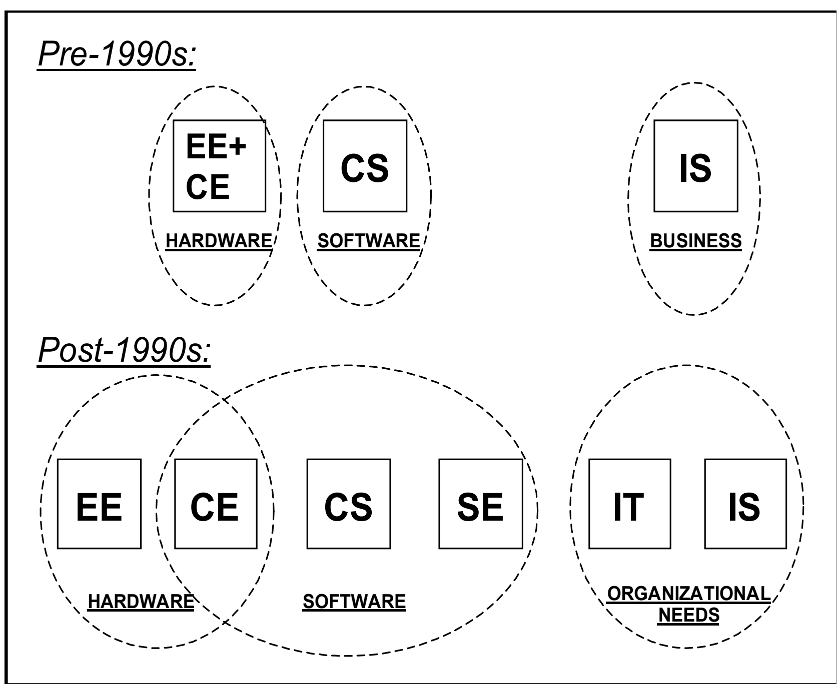


Figure 2 - The Outward Appearance of Computing Curricula Evolution