

Software Upgrade Cost Justification – Making a Case

Azad I. Ali
Indiana University of Pennsylvania
Indiana, PA, USA

Azad.ali@iup.edu

Abstract

Software upgrades are frequent in organizations, including academic departments. As soon as a major software developer releases a new version of their software, the questions repeats itself among various organizations as to whether to upgrade the software or to wait for further releases. Since the beginning of the recent difficult economic times, organizations have been trying to ration their spending and are asking for justifications for most of their expenditures. Academic departments are among those hit with budget cuts and calls to ration spending. One of the areas that was hit by such calls for justification of expenses is in the area of software upgrades.

This paper illustrates a case of one academic department that is requesting to upgrade their software and the justifications it provided to make a case to purchase new upgrades for their software. The department of Technology Support and Training (TST) at Eberly College of Business and Information Technology (ECOBIT) - Indiana University of Pennsylvania (IUP) wants to upgrade their Adobe Creative Suite software to the most recent version of the market – Adobe Creative Suite 5 (CS5). The TST department was asked to justify the purchase of this software when they submitted their request for upgrade. The reasons for the upgrade and the factors of justifications listed by this department are illustrated in this paper.

Keywords: Software upgrade, Justifying software upgrade, budget software upgrade, software upgrade and economy.

Introduction

Software vendors frequently release new versions of their software into the market. As soon as new version of major software is released, the question that echoes among users of the software is whether to purchase the newly released version of the software or whether to wait until additional versions are released (Goldsborough, 2003). Rushing the purchase of new software risks adding cost but waiting too long to purchase the software may be counterproductive as well due to increased costs and lost productivity (Ngwenyama et al., 2007).

Material published as part of this publication, either on-line or in print, is copyrighted by the Informing Science Institute. Permission to make digital or paper copy of part or all of these works for personal or classroom use is granted without fee provided that the copies are not made or distributed for profit or commercial advantage AND that copies 1) bear this notice in full and 2) give the full citation on the first page. It is permissible to abstract these works so long as credit is given. To copy in all other cases or to republish or to post on a server or to redistribute to lists requires specific permission and payment of a fee. Contact Publisher@InformingScience.org to request redistribution permission.

The potential benefits from software upgrades are numerous but at the same time there are different drawbacks associated with upgrading existing software (Mukherji et al., 2006). During difficult economic times, organizations are often asked to justify their spending and to provide cost-benefit-analysis or similar economic or financial measures to jus-

tify their expenses. One of the issues that face Information Technology (IT) managers is that their expenditures cannot always be directly gauged through these economic measures (Wulf & Jarke, 2004). In other words, certain economic measures (such as return on investment) may not be easily applied in such cases as information technology upgrades (Kingma, 2001). Instead, IT managers may have to provide different indirect measures to explain the feasibility of their intended purchase (Scherer, 1999). Mukherji et al. (2006) noted the following about the timing information technology upgrade:

In an environment of continuous change, organizations are faced with the challenge of deciding when to invest in information technology upgrades. While investing frequently is costly and at times risky, waiting too long can lead to lost competitiveness. Further, investing at a given time can preclude a firm from taking advantage of better technologies in the future. (p. 1684)

This paper illustrates the experience of one academic department that requested upgrading their software and the case they made to justify their purchase. The department of Technology Support and Training (TST) at Eberly College of Business and Information Technology (ECOBIT) - Indiana University of Pennsylvania (IUP) is trying to upgrade their Adobe Creative Suite software. When the request for the new software was submitted, the department was asked to justify their intended purchase and to provide analysis of the use of the new software. The reasons for the upgrade and justification factors listed by the department are illustrated in this paper.

The remainder of this paper is divided into the following four sections: First, the paper discusses the benefits of software upgrades. Second, the drawbacks and the problems associated with software upgrades are explained. Third, insights into software upgrades in academic institutions are introduced. Fourth, the experience of the TST department in making the case for the software upgrade is illustrated. A summary and suggestion for future study is introduced at the end.

Software Upgrades - Benefits

Software vendors frequently provide new versions of their software and send them to the market. These new releases often hold promises of better features, improved functionality and enhanced appearance (Goldsborough, 2009; Irani et al., 1997; Paine, 2000). This kind of news puts pressure on different organizations to upgrade their software. Such upgrades are becoming increasingly necessary to the survival of many IT organizations. Notess (2008) explained the following about the necessity of software upgrade:

Upgrades are a part of computing life. Few programs from a decade ago run well on modern computers and new features are offered as an enticement to upgrade. The IT economy revolves around the frequent-update. Is your computer slowing down? Buy a new one. Is your monitor too bulky? Move to a flat screen. Do you need to do your taxes? Shell out a new program every year. In general, software updates come out every few years, promising new features that are designed to make the program easier to use – and to get consumers to buy the upgrades.For the new technophile, upgrades can be exciting for their potential new features, and the challenge of learning new tools is incentive to upgrade. (p. 42)

Goldsborough (2003) noted that adopting additional software can bring the following advantages to the organization: New features, enhanced speed, improvement of the organization image and increase productivity. Paine (2000) acknowledged the challenges associated with software upgrade but noted that such upgrades could improve business operations expenses and data throughput. Most studies acknowledge the benefits associated with upgrading software, but the details of these benefits remained unclear. The remainder of this section details some of the potential benefits that can be brought to the organization from upgrading software.

New Features

Software upgrades always bring new features (Goldsborough, 2009). These may include simple changes to existing features or add new features that extend the use of the software to other areas. For example, word processing software may include new features that allow for a wider range of image and media editing. This may allow the use of the same word processing software to other areas like image editing, brochure and media preparation.

Major software vendors like Microsoft or Adobe put together major new features in one package and release the software in a new version. Minor revisions are released more frequently in terms of releases or patches. Patches or fixes are usually free while new versions are released to the market and their users often have to pay for it in order to have access to it.

There is general agreement among analysts that new versions of software always bring notable new features that make the upgrade attractive enough to some users to consider paying for it (Albright, 2008; Goldsborough, 2003, 2009; Hicks & Nettles, 2006) . However, within organizations two questions come into play: First, the potential use of the new features, and second the ease at which these new features are learned and integrated.

A study conducted by Norman (1999) noted that computers may have too many features such that they go beyond the capacity of one person. In other words, one person cannot use all the features provided by personal computer. The same could be said regarding the new features of software upgrade. The new features provided in software upgrades are plenty that users may not immediately understand the extent of such all the new features. Time, experience and extended use may necessitate putting the new features into use. However, this puts into question the second factor about the ease of learning the new features. An added decision point for software upgrade is the ease of learning and integration of the new features of the upgrades.

Enhanced Speed

Speed of the software often gets faster with new versions. This is always a benefit in terms of functionality and performance. In other words, giving all other conditions equal, a person with a higher speed of the software is able to complete the same task in less time. This in turn leads to enhanced productivity; improved performance and reduced frustration over delays at work (Mehra & Seidman, 2006; Paine, 2000). In terms of work environment, speed of processing is more crucial to certain organizations than others. An educational institution that uses certain software for student registration may be more equipped to handle large registration data than other with slower software. Similarly, healthcare providers may get a better use of faster software when entering large volume of patient data. In these examples, enhanced processing speed may lead to increased productivity, reduce cost, or enhance work environment. In other cases, enhance processing speed of software is always positive as it often leads to reduce waiting time.

The enhanced processing speed of new version of software is contingent often upon the minimum hardware capacity that the new software requires (Wulf & Jarke, 2004). If the minimum requirement of the software does not necessitate the organization to purchase new hardware, then this will spare it from additional purchase of hardware and further installations.

Enhanced Image of the Organization

An organization with newer technology often gives the impression of superior image due to the use of technology. For example, an organization using Windows 7 may be better off in the eyes of the customers than another organization that uses windows XP.

The introduction of new version of software means having two things: It means having features of the old version available and it means additional features are available as well. The availability

of the added features in the new version may give the impression of the more superior technology.

Software Upgrades - Drawbacks

Ellison and Fudenberg (2000) explained some of the problems associated with software upgrades. They noted that major software industry release upgrade after upgrade and that these upgrades often have drawbacks and frustrate users:

In particular, users complain that in the Internet era, sales of new "backward-compatible" upgrades are an exploitation of network externalities designed to "force" people to buy upgrades they don't want and that are also inefficient because they leave others struggling with incompatibility. (p. 253)

Notess (2008) reported that introducing new software upgrade brings many challenges that included the additional cost that the organization is required to pay for, the challenge of learning/relearning the new version of the software, and additional administrative procedures associated with introducing the new software. Stocker and Dugan (2003) explained that the beginning phases of introducing new software are often accompanied by problems that minimize the working of the organization. The remainder of this section highlights some of the drawbacks or challenges from introducing new software to the organizations.

Cost

Reports of the waste cost on software purchases are numerous. Hopkins and Kessler (2002) reported that U.S. Companies threw away \$130 billion in the past two years on unneeded software and other technology. The same study found out that of 25 years of tech spending worldwide, companies waste as much as 20% of the of the \$2.7 billion spent annually on technological upgrades. A 2004 Standish Group study reports that 71% of these IT projects did not meet top management's expectations (The Standish Group, 2004).

The drawbacks of technological upgrade is not limited to purchase and installations. Instead, these upgrades require support and this in turn cost organizations significant amount of money. Hopkins and Kessler (2002) reported that in one year, companies spend \$5 in support fees for every \$1 in software cost. Having this much expensive support cost may lead organizations to think more often on the drawbacks of software upgrades. In time of budget cuts, upgrade cost becomes more of an issue than in other times. The cost issue is critical in IT because there may not be a direct way to analyze the benefits of such investment of software upgrade. Instead, IT managers may have to resort to more rigorous and indirect means of analysis to justify their spending on software upgrade.

Training/Retraining

Software purchase may require training new users of the software or it may require retraining existing users of the same software on the new features provided by the upgrade. The extent of training/retraining schedule is dependent on some factors including the following:

- Magnitude of the upgrade and the new features provided
- The extent at which the new features are used
- Whether the organization requires people to be self taught on attend training program.

A more intensive training requirement of the software may place put an extra burden on the organization. It may lead to the development of a training schedule, extra time and cost for the training and putting aside time and energy to fulfill the training requirement. At times, it may require putting replacements to work for the employees attending the training.

Compatibility with Existing Software/Hardware

In most instances, the introduction of new software requires purchasing new and more advanced hardware. In other cases, the new software does not require the organization to purchase new hardware. The decision of new hardware is dependent on the minimum requirements of the new software. If the existing hardware meets the minimum requirements of the new software, this will be less of an issue. On the other hand, even if the current hardware meets the minimum requirements, in some cases the new software may slow down or affect performance of the hardware.

In terms of compatibility with software, most of the new software is compatible with the older version. In other words, files saved under older version will work in the same way under the new version of the software. However, the new version of the software may affect other installed software as well and may render conflicts with other existing software, thus further complicating the installation issues.

Software Upgrades in Academia

There are several similarities and differences between the IT environment in industry and the teaching of IT at colleges and universities in terms of software upgrades. One similarity is the desire to introduce the latest technologies into the work environment which may be often accompanied by the rush to introduce the latest software into the classroom. While introducing the latest software is beneficial often but the rush to update technology may be counterproductive in other instances. Goldweber et al. (1997) noted about this rush of updating technology curriculum:

Rapid changes in computing often motivate educators to introduce innovations in the curriculum and the classroom. The haste to do something new or adopt some current fad can cause teachers to overlook possible adverse effects of these innovations on students and the profession. The deployment of curricular or pedagogic innovations such as new languages and technologies may seem appropriate, but mistakes are costly. (p. 94)

A notable particular to the work of academic institutions in terms of their technology installations is the abundance in hardware and software available for the different faculty departments (Arteaga & Lucas, 2005). This kind of abundance creates opportunities but at the same time has different associated problems (Ali & Murthy, 2010). The remainder of this section explains the particulars of academic institutions in terms of the benefits and drawbacks that software upgrades brings to the institutions which were discussed earlier.

New Features

In most cases, academic institutions are supposed to be training students for the job market. Although there are various theories on the purpose and goal of academia, but the preparedness for the job market is a prime and common goal among most. The new features of software upgrades bring opportunities to the marketplace, as some of these new features create opportunities for those who were trained on them. A department that teaches the new technology may be better able to respond to this kind of demand in the job market. Thus, most often academic organizations desire to introduce new technology sooner than later sooner than later to ensure up to date graduates of the programs.

Enhanced Speed

Speed of processing in terms of software introductions is crucial to teaching in academic institution and their work. Speed of the technology is considered essential when taking into considera-

tion the number of students, faculty, staff and administrators working on the technology at the same time. Enhancing the speed of one piece of software on one computer may benefit multiple faculty, multiple students, and different departments. The same computer in one lab may be used by different faculty and is accessed by different students. This in turn creates a chained benefit to all involved.

Better Image for the Organization

The image of the academic department may be enhanced by the introduction of the latest technology. Moreover, the image of the organization in terms of the technologies used is often compared to their counterpart colleges and universities. A college or a university that had more updated technology may be considered more superior than other colleges with outdated technology if all other factors remain the same.

However, image of the department can be measured in terms of the population in which they draw their students from. If the surrounding high schools to the department teach more up-to-date technology than a department at the college or university, this may not be looked as a positive factor. A college or university with more up-to-date technology may be able to draw more students with more advanced technology.

Cost

Difficult economic times are hitting academic institutions as well. Thus, academic departments have to ration their expenses and have to justify their costs separately. The manner in which academic departments operate may be different than at other organizations.

Academic departments have numerous software installations. A particular software when purchased in a given college or university may need to be installed multiple times. Thus the cost of software is often not limited to one license or a few licenses; instead it has to be purchased to cover the labs and the faculty/staff that use it. Luckily, major software vendors issue volume licensing and often offer academic licenses at reduced prices. This may help in overcoming the cost issue.

The cost issue is often complicated when taking into consideration that the new software may need additional software/hardware requirement. In these cases, having more advanced hardware in one computer lab may create a chained effect from other departments and the call for renewing hardware in their labs as well.

Compatibility with Hardware/software

The issue of compatibility with existing hardware/software is crucial to different departments at colleges and universities. One particular issue is that at colleges and universities is that faculty teach at computer labs that are specific for teaching but there also general public labs that are available to the students. The teaching labs may often be limited to one department but the general public labs are open to students enrolled in all departments. Given the different departments involved in any given lab, this may create multitude of software installations and may cause conflicts among the installed software (Ali & Murthy, 2010).

The incompatibility among software may not be experienced immediately after the installations of the software as there may be delayed effect that may appear as a result of certain condition or certain requirement that may cause the software to malfunction. Thus, the result of this incompatibility may take an extended period to show up.

Training/Retraining

The training for new software in academic institutions is often a two-phased process:

- Learning the essentials of the new software
- Learning how to deliver the new software in the classroom

In most cases, learning is completed unofficially and often faculty will be self taught. However, to fulfill the requirement for both learning phases for the students, the faculty may need to design new lesson plans, select new textbooks and create new exams. Moreover, the faculty may need to develop new methodologies for the teaching and assessment of the new features to be learned. This puts extra burdens on the faculty while experiencing a learning curve with the new software.

Software Upgrades – Making the Case

The Technology Support and Training (TST) department in the Eberly College of Business and Information Technology (ECOBIT) at Indiana University of Pennsylvania (IUP) offer two bachelor degrees, and one associate degree at the undergraduate level. The first bachelor degree is in Business Technology Support while the second is in Business Education. The Associate degree is in Computer and Information Technology. Additionally, the department offers a master degree in Business Education (M.Ed.). The TST department teaches two different undergraduate web design courses and one graduate course in web development. Their main software of choice regarding the teaching of at least two web design courses is the Macromedia Suite. This suite has different software that helps with image editing, animation, and web design tools. Additionally, the TST department uses some other media software like Sony Vegas, DVD Architect, and Soundbooth for the same courses as well.

Recently, the TST department requested the purchase of a new upgrade to Adobe CS5 software. This request was thought to be a routine request in light of the length of time that has passed since the last request by this department for software upgrades. But recent budget cuts forced the department to study the features of this new version of the software to make the case and justify spending money for the new version of the software. The remainder of this section explains the justifications for the new software. The section starts by explaining the software that is currently in use. It then discusses the perceived benefits and drawbacks as it pertains to the Adobe CS5 software and the particular of the course being taught at the department. The section ends with an explanation of how the department made the case for purchasing the new software.

The Current Software

At the time of writing this paper, the TST department uses two versions of the software that dates back to the days when Adobe and Macromedia used to be two separate companies and produced two different software tools. At the time of this writing, the TST department is using MacroMedia Suite 8 and Adobe Photoshop CS2. The TST department is also using Sony Vegas 6 for media purposes. All these software tools were first purchased in 2006. The latest technology in the market at the time of this writing is Adobe CS5 and Sony Vegas 9. The CS5 version includes a combination of the software that was developed by Adobe and Macromedia companies. They merged into one company and produced Adobe CS3, Adobe CS4 and most recently Adobe CS5 as different packages of the software combination in the suite. To put it in simple term, the current software being used at the TST department is at least two versions behind the market versions.

New Features in CS5

From the literature review, it appears that Adobe met the expectations in their release of CS5 versions (Lam, 2010; Stafford, 2010). This is evidenced by the increase of the company's profits

with their release of CS5 in April 2010 (Mike, 2010). Most analysts agree that Adobe CS4 did not meet expectations, thus the company tried to released a new version that overcome the repetition that the release of CS4 created (Lam, 2010; Mike, 2010; Stafford, 2010). From the literature review, it appears that some of the new features include the following:

- Creative licenses that satisfy different user demands (Parsons, 2010)
- Better menu selections and options
- Improved control over content
- Easier creation of web animation (Mike, 2010)
- Many other new features geared

In addition, faculty that teaches a few courses with this software intend to introduce the new features as soon as possible. They may have already adopted new textbooks for one web design course that uses several new features of Adobe Dreamweaver CS5 and Adobe Flash CS5 and intends on teaching a summer workshop that uses Adobe Photoshop CS5.

Enhanced Speed

One of the slow features of web design courses is the speed at which files are uploaded and downloaded. Adobe CS5 provides faster access to digital files as compared to their older versions (Lam, 2010, Stafford, 2010). It appears from at least one test that new software provides faster access without the need to upgrade hardware. As a general observation, it seems that Adobe CS5 offers faster editing features on comparable hardware (Stafford, 2010). In other words, users do not need to upgrade their hardware in order to take advantage of the higher speed that CS5 offers.

Image of the Department

Being two versions behind the market does not help with the image of a department that teaches three web design courses. Additionally, an unofficial survey of the school districts in the area as well as the universities in the same system revealed that the TST department falls short of both of them in terms of using Adobe CS software. All others that were included in the unofficial survey use a more recent version of the software than is being used by the TST department.

Cost Issue

Cost has become more of a difficult issue with the recent economic problems and budget cuts, although the college assesses technology fee which is a pool of money that doesn't come out of college's operating budget. Various departments submit requests for using the tech fee for their purchases and there is a technology committee, along with the dean, that decides on spending the money based on the priorities established by the committee.

Training/Retraining

For most part, although different faculty submitted requests for this software, training on the new software was up to the faculty members themselves. This required faculty members to learn the software, select new textbooks, create revised lesson plans, and establish new procedures for assessment

Compatibility with Existing Software/Hardware

The department has recently undergone an upgrade of the operating system from Windows XP to Windows 7. Although there was a minor reduction in speed of the computers but no major problems were reported from upgrading to the most recent technologies. Additionally, prior installa-

tions generated some problems but the technology support department at the college was able to contain these problems and install fixes for them without major distributions. Thus, it is not anticipated that the new version will generate major problems.

Making the Case to Upgrade to Adobe CS5

The TST department has been asking for an upgrade to the Adobe CS software for more than a year. Despite the numerous requests to upgrade, the department has not received they asked for. Instead, the request was delayed for financial and administrative reasons. Efforts of the department included an explanation that the current version of the software was behind the current market version and that it would better serve the students, and the university, to have the current version

Following memos and meetings to discuss the issue of purchasing the new software, the TST department was asked to align the software features with the goals and objectives listed in their program and in the courses that use the software. This led the department to search through their curriculum and through their course syllabi in order to review their course objectives and to find the objectives that use the intended software. The department developed a table listing the course number, the course objective that uses the intended software and the specific software that is used to meet the course objective. Table 1 below shows the document that the TST department submitted to support their request for purchasing Adobe CS5 software.

Table 1 – Aligning CS5 Features with TST Department Course Objectives

COURSE #	COURSE OBJECTIVE	SOFTWARE USED	
BTED411/511 BTED412/512 BTED470/570 BTED 609 BTST401 BTST670 BTST675	Identify elements of planning and designing successful web sites	Dreamweaver	
	Add text, hyperlinks and style sheets to web pages		
	Participate in the construction of and posting to a blog.		
	BTED470 BTED609 BTST402 BTST655 BTST675	Design shared components on web sites	Flash
		Apply an understanding of Web design principles to create effective, professional quality Web-based instructional materials	
		Integrate rich media elements into web sites	Photoshop
		Create dynamic web pages	
		Incorporate animation and layer tools into web pages	
BTED609 BTED470 BTED609 BTST480 BTST655 BTST650 BTST675 BTST680	Implement client and server side web development	Contribute	
	Identify elements of web server technologies		
	BTST655 BTST675	Connect web pages with data sources	Dreamweaver
		Manage web state information	
BTED609 BTED470 BTED609 BTST480 BTST655 BTST650 BTST675 BTST680	Examine recent trends in topics related to technology support	Premiere, Soundbooth Sony Vegas	
	Utilize technology for effective marketing of student' skills and knowledge		
	Assess various forms of software and media available to enhance e-learning		

COURSE #	COURSE OBJECTIVE	SOFTWARE USED
BTED411 BTED511 BTED470 BTED609 BTST655 BTST675	Use Acrobat files in the review cycle of documents Add signature and security to Acrobat document	Acrobat pro,
BTED470 BTED609 BTST401 BTST675 BTST680	Use software tools to create, enhance and modify contents in digital media Create e-learning content modules that is designed to meet the needs of all learning styles	Soundbooth
BTED470/570	Integrate the use of software applications into an interactive lesson presentation	Sony Vegas

Research Summary and Future Research

This paper explained about upgrading software from one version to another. It started by discussing how budget crises and difficult economic times force managers to further explain and justify their purchases. The factors that necessitate upgrades were reported, as well as the drawbacks that can accompany software upgrades. Then the potential problems with software upgrades in academic institutions were presented. Last the papers focused on the TST department at IUP when they requested upgrading their older software to newer version and the justifications they submitted for their purchase.

Although the TST department is using this commercial software and is paying for upgrades, faculty in the department is also using free and open software as well. However, the experience of the author suggests that free and open source software is not for every student. Instead, students with certain technical skills can work and learn from open source software while commercial software is more suitable for other students. Some open source software cannot be used for all the students in the department while at the same time the budget situation requires some rationing and cutting down on the purchase of new software. Thus some balance between the two types of software (commercial versus open source) may need to be practiced for teaching within technology departments. The balance of software and the issues associated issues with using both type of software will be the focus of discussion in the author’s next paper.

References

- Albright, B. (2008). Ready to upgrade? *Behavioral Healthcare*, 28(10) 28-29.
- Ali, A., & Murthy, R. (2010). The coordination between faculty and technical support staff in updating computer technology courses – A case example. *Issues in Informing Science and Information Technology*, 7, 459-469. Retrieved from <http://iisit.org/Vol7/IISITv7p459-469Ali827.pdf>
- Arteaga, C., & Lucas, T. (2005). So many labs, so little time. *Proceedings of the 33rd annual ACM-SIGUCCSuconference on User services*. 192-196. Retrieved April 20, 2009 from ACM Digital Library <http://www.acm.org/dl>
- Ellison, G., & Fudenberg, D. (2000). The neo-Luddite's lament: excessive upgrades in the software industry. *RAND Journal of Economics*, 31(2), 253-272.

- Goldweber, M., Impagliazzo, J., Bogoiavlenski, L., & Clear, A. (1997). Historical perspectives on the computing curriculum. *ITCSE97 Working Group*. Retrieved October 10, 2006 from ACM Digital Library <http://www.acm.org/dl>
- Goldsborough, R. (2003). Computer savvy. *Community College Week*, 15(18), 19-22.
- Goldsborough, R. (2009). Windows 7 can wait until it's time to buy a new computer. *Community College Week*, 22(7), 12-13. Retrieved from Academic Search Complete through EBSCOHost, 11/21/2010.
- Hicks, M., & Nettles, S. (2006). Dynamic software updating. *ACM Transactions on Programming Languages and Systems*, 27(6), 1049–1096.
- Hopkins, J., & Kessler, M. (2002, May 20). Companies squander billions on tech. *USA Today*. Retrieved from EBSCOHost 11/18/2010.
- Irani, Z., Ezingard, N., & Grieve, R. J. (1997). Integrating the costs of a manufacturing IT/IS infrastructure into the investment decision-making process. *Technovation*, 17(11/12), 695-706.
- Kingma, B (2001). *The economics of information: A guide to economic and cost-benefit analysis for information professionals*. Englewood, CO: Libraries Unlimited.
- Kotamraju, N. (1999). The birth of web site design skills: Making the present history. *American Behavioral Scientist*, 43(3), 466-477.
- Lam, S. (2010). Benchmarking Adobe Premiere Pro CS5. *EventDV*, 23(8), 38-43. Retrieved from EBSCOHost, 11/29/2010.
- McCrea, B. (2007). 5th annual software survey: Investment time. *Logistics Management*, 46(4), 43-47.
- Mehra, A., & Seidmann, A. (2006). The economics of software upgrades throughout the product life cycle. *Proceedings of 39th Hawaii International Conference on Systems Sciences*. Retrieved from IEEE 11/18/2010.
- Mike (2010). Adobe CS5 is meeting expectations. Retrieved December 5, 2010 from <http://cs5.org/?p=1354>.
- Mukherji, N. A., Rajagopalan, B. B., & Tanniru, M. (2006). A decision support model for optimal timing of investments in information technology upgrades. *Decision Support Systems*, 42(3), 1684 –1696.
- Ngwenyama, O., Guergachi, A., & McLaren, T. (2007). Using the learning curve to maximize IT productivity: A decision analysis model for timing software upgrades. *International Journal of Production Economics*, 105, 524-535. Retrieved 11/18/2010 from ScienceDirect <http://www.sciencedirect.com>.
- Norman, D. (1999). *The invisible computer. Why good products can fail: The personal computer is so complex and information appliances are the solution*. Cambridge, Massachusetts: MIT Press.
- Notess, G. R. (2008). The downside of upgrades. *Online* (Weston, Conn.), 32(6), 42-44. Retrieved from Wilson Web 11/06/2010.
- Paine, M. (2000). Making software upgrades a first-class experience: Issues, opportunities and speed bumps to consider for a smooth ride. *Health Management Technology*, 21(4), 22-23.
- Parsons, J. (2010). Creative license. *American Printer*; 127(8), 12-13. Retrieved from EBSCOHost 11/29/2010.
- Scherer, F. M. (1999). *New perspectives on economic growth and technological innovation*. Washington DC: Brooking Institute Press.
- Stafford, A. (2010). Premiere Pro CS5: Faster editing, higher demand. *PC World*; 28(6) 53-53. Retrieved from EBSCOHost 11/29/2010.
- Stocker, E., & Dugan, J. B. (2003). When does it pay to make software more reliable? *Proceedings of the 14th International Symposium on Software Reliability Engineering*. 321-333. Retrieved November 11, 2010 from IEEE Computer Society.

Software Upgrade Cost Justification

The Standish Group. (2010). *Trends in IT value*. Retrieved from http://www.standishgroup.com/sample_research/index.php/trends_itValue.pdf

Wagner, S. (2007). Using economics as basis for modeling and evaluating software quality. *Proceedings of the First International Workshop on the Economics of Software and Computation (ESC-1)*. IEEE Computer Society Press.

Wulf, V., & Jarke, M. (2004). The economics of end-user development. *Communication of the ACM*, 47(5), 41-42.

Biography



Azad Ali, D.Sc., Associate Professor of Technology Support and Training at Eberly College of Business – Indiana University of Pennsylvania has more than 20 years of combined experience in the area of information systems. He holds a bachelor degree in Business Administration from the University of Baghdad, an M.B. A. from Indiana University of Pennsylvania, an M.P.A. from the University of Pittsburgh, and a Doctorate of Science in Communications and Information Systems form Robert Morris University.