UNIVERSITY OF WISCONSIN, PLATTEVILLE

UNITED STATES OF AMERICA

The attached educational project, by NICHOLAS FELVER, entitled SOFTWARE

PROCUREMENT AND THE DIGITAL SUPPLY CHAIN: HISTORY, OVERVIEW AND

BEST PRACTICES, when completed, is to be submitted to the Graduate Faculty of the

University of Wisconsin- Platteville in partial fulfillment of the requirements for the

(MASTER OF SCIENCE IN INTEGRATED SUPPLY CHAIN MANAGEMENT) degree.

Approved: *Mary R. Bartling*   Date: 5/10/19

Project Advisor

Professor Mary R. Bartling

Suggested content descriptor keywords:

SOFTWARE, PROCUREMENT

SOFTWARE PROCUREMENT AND THE DIGITAL SUPPLY CHAIN: HISTORY,
OVERVIEW AND BEST PRACTICES

A Paper

Submitted to the Graduate Faculty of

the

University of Wisconsin, Platteville By

NICHOLAS FELVER

in Partial Fulfillment for the Degree of

MASTER OF SCIENCE IN INTEGRATED SUPPLY CHAIN MANAGEMENT

Year of Graduation: Spring 2019

Abstract

The topic of this paper will focus on the history of software procurement and the changes in organizational strategy and tactics surrounding the software acquisition life cycle. This will include a range of software products from simple plug-and-play licenses to complex configurable software solutions requiring project teams for implementation. Best practices associated with sourcing and purchasing different software solutions and products are identified and described. Wise organizations will first seek to understand and define the problem they hope to solve with software before beginning the process to purchase it. They will project the long-term consequences of their purchase into the future and consider the pros and cons. Wise organizations will continue to evolve their change management practices to ensure employee buy-in to new technologies that make their organizations more productive and efficient. As electronic security risks proliferate in the modern world, organizations will continue to use software to manage risk and secure their reputations. They will stay curious and proactive as new technologies emerge to remain competitive in the global economy.

Keywords: Software, Procurement

TABLE OF CONTENTS

Introduction

Statement of the Problem

The topic of this paper will focus on the history of software procurement and the changes in organizational strategy and tactics surrounding the software acquisition life cycle. This will include a range of software products from simple plug-and-play licenses to complex configurable software solutions requiring project teams for implementation. The goal will be to identify and describe the best practices associated with sourcing and purchasing different software solutions and products.

A proliferation of software products and solutions developed in the 1990s to meet changing business needs and help organizations enter the internet age. As the enterprise software industry matured, it has experienced many changes as consolidation of vendors increased since 2000 (Kauffman & Tsai, 2009). Single-source vendors were able to provide a range of products to meet their customers' business requirements. This trend continues to the present day but, increasingly, new start-ups are providing innovative and alternative software solutions for business problems.

The internet, globalization and better access to developmental tools has created even more product offerings originating in multiple countries. This can create challenges for organizations looking to protect themselves from legal risks associated with software purchasing and use.

Procurement professionals face a diverse set of challenges when looking to purchase the software their organization needs. A skill set is needed such that they can negotiate deals with multi-million-dollar software providers while retaining the flexibility to work with smaller,

innovative start-ups at the cutting edge of technology. Different approaches are required for each and the procurement professional must navigate the sometimes-conflicting demands of their information technology (IT) and legal departments to find the best solutions for their business. Depending on the needs of their business, this flexibility and expertise can make the difference between success and failure for their organization.

Purpose of the Study

The primary purpose of this paper is to understand the history of software procurement and identify its challenges and opportunities. After defining the desired goals and outcomes of successful software procurement, this paper will provide examples of organizations that have done this well as well as some that have not. This will include a discussion of the legal implications to be considered when engaging in software procurement. Finally, a menu of best practices for software procurement professionals will be presented.

Significance of the Study

Software has been and continues to be a determining factor to business success. This will only grow in the future as organizations continue to focus on digital customer experiences and real-time information on consumer demands through the Internet of Things (IOT).

Assumptions

Procurement of software carries varied amounts of risk including the data it stores, the customers it touches and the legal implications of misuse of the software provider's property.

There are also safety and security concerns as a purchased software solution can be an avenue for hacking efforts. According to Ladd (2006), reasonable procurement policies can effectively raise security awareness and add to the computing environment's overall safety. Any purchasing and integration into the organization's software environment must be carried out thoughtfully with risk mitigation in mind while still meeting the needs of the business.

Methodology

A literature review will be conducted encompassing the history, best practices and legal implications of software procurement. Sources will be reviewed and analyzed to provide support for the goals of the paper. In addition to peer-reviewed academic source materials, internet research will be conducted regarding organizational practices as well as best practices for software procurement professionals.

Review of Literature

History

So called "embodied" software was the model with which early software was patented in the 1960s according to Con Diaz (2015). A description of the physical machine that the software controlled was used to solidify understanding of the software algorithm in question. By filing for a patent for its "Autoflow" software, the company Applied Data Research (ADR) assisted in the process to separate software and hardware in computer purchases, promoting the use of software from multiple companies on a single machine. Software patenting predated the software industry and helped give rise to innovation outside of the world of government research and hardware giants (Con Diaz, 2015).

Selection Methodology

Adoption of Agile software development methodologies can improve outcomes for information technology (IT) procurement activities (Baker, 2019). By focusing on quick wins and short-term deliverables, procurement professionals can hold suppliers accountable in terms of the actual product rather than solely tracking hours worked. In terms of software acquisition and implementation, use of the agile methodology will continuously focus on the testing of specific functionality piece by piece rather than blindly paying for a final product.

Purchasing accounting software is akin to buying a house according to Jones (2016). Use of a consultant to help understand the marketplace and manage the implementation can be beneficial in the same way that a realtor can best help a family find the right house. Different requirements will prescribe the structure and capabilities needed for a successful solution. Software has ongoing costs for maintenance, a factor that must be taken into consideration when evaluating the return on investment (ROI) for the solution. Implementation of the new software is a multi-phased effort that must be well-planned and tested before going live (Jones, 2016).

According to Michler (2005), open source software can be a desirable option compared to traditional closed source software. There are considerable cost advantages and source code can be modified as needed. In addition, open sourced software is considered to be more stable and secure as more users have access to test and refine it. Open source is not without drawbacks however as there are legal and technical risks associated with its use. The owner of the open source code may decide to end its free use and restrict access, or it may not be compatible with existing proprietary systems used (Michler, 2005). These risks must be evaluated by organizations before taking advantage of open source solutions.

The relationship between the software buyer and supplier will vary based on the procurement paradigm according to Brereton (2004). Commercial off the shelf (COTS) products are purchased and used as is by the customer, resulting in a long-term partnership with the supplier. Component-based software engineering (CBSE) products are made up of different preexisting components and are designed and integrated by the customer. Software service engineering (SSE) products are essentially subscribed to by the customer and delivered over the internet. The risks of the SSE approach are higher, but the associated opportunities can be rewarding (Brereton, 2004). History since the article's writing in 2004 has shown that such opportunities were indeed great.

Krouse (1999) outlines eight steps for hardware and software procurement to be followed by healthcare organizations.

1. Understand the need.

2. Capitalize the project.

3. Understand the Market.

4. Structure the right team.

5. Gather vendor information.

6. Apply drivers and values.

7. Integrate process and system.

8. Negotiate wisely.

These steps must be followed in order to set the stage for successful implementation of the software solution. They will help to maximize the benefits of the system and enable future successes (Krouse, 1999).

According to Kauffman & Tsai (2009), advances in IT have led more organizations to move to a unified procurement approach. Unified procurement happens when the organization selects a single vendor from whom to purchase all applicable software products. By using a unified procurement approach, the organization can simplify integrations between products, transfer more risk for managing and maintaining products to the vendor, gain greater leverage for faster customer support and realize cost savings from bundling and consolidation. This shift toward unified procurement was found to be most likely in small and medium-sized businesses, with the reasoning that large organizations were already able to realize many of these benefits through the scale of their larger, separate multi-vendor partnerships (Kauffman & Tsai, 2009).

Procurement teams often do not have enough knowledge about the software products they are tasked with acquiring. As a result, much time and effort are expended to create and use metrics and criteria. According to Williams & Pollock (2007), comparisons are possible only with much effort, often it is not the properties of technology but the way they are weighted and evaluated in the comparison that makes the difference and the comparisons themselves are easily swayed subjectively by members of the procurement team. In order to make the best decision, procurement teams must learn as much about the requirements and product as possible, strive to recognize where any biases may lie in their comparisons and incorporate both rational, technical considerations with the implicit sociological factors involved.

Purchasing and Implementation

According to Samuelson (2011), copyright law states that owners of property (including intellectual property) are only able to control the first sale of a copy of that property. After the first sale, the buyer is then able to sell or transfer that copy as they see fit. Software licenses can

be different however. Depending on the terms of sale, a license is issued to a licensee and is often

not transferable to other parties. Consumers often see these terms as so called "click through"

agreements where acquiescence is required for access. The terms of sale and licensing model

must be understood to mitigate risk in software transactions.

Too often, large software acquisition projects miss the mark by not delivering the results

promised by the vendor. This is often due to the procurement team not understanding the large

number of associated requirements for the software. In addition, they may not understand all of

the options available, limiting their choices to off-the-shelf offerings standardized to meet a wide

range of customers. This can lead to the need for further customization to meet their specific

needs, resulting in higher costs for the changes and ongoing maintenance (McDonnell, 2013).

By definition, an open source software license will not restrict any party from selling or

giving it away. An open source program must include the source code and be sharable in both

source and compiled form (Bretthauer, 2002). In terms of software procurement, open source is

does not always mean free products, but freedom to use those products however the acquirer sees

fit.

Organizations are increasingly turning to software solutions to help them meet their

health and safety objectives according to Glass (2017). However, software alone will not solve

all problems. The culture of the organization must be considered in implementing a new solution.

Executive leadership support must drive any new software efforts and communication on use of

the tool and its importance is paramount. Internal marketing and incorporation of software

utilization metrics is necessary for a successful implementation (Glass, 2017).

Intellectual Property and Risk

For the United States Department of Defense (DoD), software acquisition purchases are governed by the Defense Federal Acquisition Regulations Supplement (DFARS). According to Dungan (2010), intense regulation of intellectual property (IP) rights by DFARS is inhibiting innovation. The encumbrance of DFARS discourages small and non-traditional software companies from engaging with the DoD which increases the possibility that the DoD will miss out on the emergent technology that comes outside of the traditional hegemony of the software industry. DoD must respect the IP rights of the makers of useful and innovative software if it is to remain on the cutting edge (Dungan, 2010).

Outsourcing business processes to third parties can open an organization to cyber security risks that it may not have anticipated according to Thomson (2018). Hackers are always trying to find new ways into a company's systems and the use of a wide range of vendors can create the same number of access points. This is especially true in the software purchased by a company. Organizations must include cyber security requirement standards in their contracts with third parties, holding them accountable for any breaches. Contractually obligating the vendor for any breaches will motivate them to maintain high standards in their own systems. The buying organization's own standards for cyber security practices should be used to inform the due diligence with which it vets any potential supplier partners (Thomson, 2018).

As software continues to grow in importance to the organization, the scope of software procurement must increasingly consider security when choosing a product and vendor. According to Ladd (2006), security considerations are often not a priority until after a breach has happened. Post-incident management results in reactive new policies that are minimally adhered to as departments seek to conform with the least amount of cost and effort. A proactive approach

is needed and it must begin with the acquisition process. By requiring secure standards and partnering with the vendor to maintain those standards, procurement teams can keep their organizations protected from external threats (Ladd, 2006).

Best Practices

Outsourcing for software solutions promises savings and innovation but does not always reach its goals according to Putnam (2018). Four best practices; pre-acquisition, request for proposal (RFP), award and post-award should be thoughtfully planned out and executed in order to achieve the desired outcomes. Pre-acquisition must quantify the scope of the project and independently determine cost and timeline estimates. The RFP process must issue clear and concise requirements for vendors to bid on, including plans for unexpected changes. The award must include a full assessment of the vendor to ensure they can fulfill all aspects of the project. Lastly, the result and performance must be measured and adjusted continuously in the post award phase (Putnam, 2018).

Tate (1999) argues that most organizations struggle and miss the mark when purchasing and implementing business software. While many fault the vendor for the failure, the ultimate decision for purchasing rests with the organization and hence the blame. The organization must gather information from and navigate the biases of different stakeholders including external salespeople and consultants and internal stakeholders such as IT staff and end-users. Alignment of stakeholder objectives with project success factors is crucial. In addition, incentivizing consultants and vendors for achievement of the objective by paying them for the result rather than the time spent can focus their efforts and bring success (Tate, 1999).

There are several legal concerns that are unique to software purchasing according to Brennan (2003). Specifying the jurisdiction in which disputes will be settled is important as different localities may have different laws regarding software licensing. This is especially important for products developed overseas and sold and distributed electronically over the internet. The buyer should ensure that the vendor has provided a warranty of some kind that the software can function as advertised and is free of defects. Limitation of liability is also important to understand as the seller will try to limit their exposure if the product causes a loss to the buyer (Brennan, 2003). This may happen through infringement of intellectual property or exposure of sensitive information within the software.

Evaluation and selection of software and service providers can be narrowed down to ten steps according to Pendergraft & Blakely (2010). These are...

1. Create a project team and action plan.

2. Identify and map sources of information and records.

3. Review current process for potential improvement.

4. Prioritize business needs.

5. Pre-qualify potential solutions.

6. Pre-evaluate potential service providers.

7. Validate service provider references.

8. Make the decision.

9. Establish an implementation plan.

10. Negotiate price and service level agreements.

Purchasers are cautioned against paying for extraneous features that do not add to the bottom line and needlessly complicate the solution. It may be unwise to purchase brand new, untested solutions or old solutions that are at the end of their supported life. It is also helpful to include your current providers in the search for new solutions as there may be unused functionality within your current offering that could solve the problem (Pendergraft & Blakely, 2010).

As IT needs and capabilities continue to evolve, governments need to seek out increasingly new and innovative ways of identifying and sourcing software. However, Sieverding (2008) argues that such activities should not be based on express or implicit premises, an approach that may be missed by emphasizing open source software solutions. While a preference for open source seems beneficial on the surface, such preference may exclude a proprietary product that may serve the need more effectively. Lower cost is not always the best approach. A better approach is a preference for interoperability with a variety of solutions in order to better react to changes in technology and ensure the best solution for each scenario. Governments should promote free choice (whether open source or proprietary) by their IT departments in order keep pace with innovation (Sieverding, 2008).

Discussion

History

Software has come a long way in a very short span of time. As code used to program and direct mechanical and electronic machinery, its use has expanded from a very narrow scientific, governmental and military focus to its present use in all fields and industries in a variety of applications. Discovery of new ways to create, simplify and distribute code has coincided with

great leaps forward in the capabilities and impact of all kinds of technologies. Software has grown exponentially in usefulness and complexity (Dungan, 2010).

The advent of the internet greatly changed the way in which software was developed, distributed and sold. Prior to the internet, software would have to be physically shared via some kind of electronic storage. Sharing meant the copying of a disc or a physical connection between machines. Companies sold software on physical discs (floppy or compact) and used encryption keys to prevent duplication and sharing after the sale. On an organizational level, companies buying software stored the purchased code on physical hard drives or servers within their walls. Any modifications or enhancements would have to be made at the physical level where the software was stored, either through manually altering the code or deleting it altogether and installing new code.

The internet provided new options for buying and modifying code. Code can now be transferred electronically to new machines and distributed and copied perpetually. Rather than mailing physical copies of code, software manufacturers sell licenses to copy and distribute their software. While many buyers still purchase software code that is stored on their own equipment, increasingly, a Software as a Service (SaaS) model of software sales is growing. SaaS allows an organization to purchase a license to access the software electronically on the selling organization's hardware without having to own the code or host it themselves (Jones, 2016). Otherwise known as "cloud" software, this model is becoming increasingly popular due to its cost savings and ease of modification and upgradability. Though a useful development, cloud software brings with it risks that must be researched and mitigated by the buyer in the contracting progress.

Selection Methodology

Selection of software products is an important decision for any organization. The choice of software is influenced by the problem it solves, the people that will be using it, the technical specifications and the cost.

Before choosing software, the organization needs to clearly define and articulate the problem that the software will address and envision the new state of capability or efficiency that the tool will provide. This process may be more complicated depending on the type of software needed and the scale, both technical and financial, of the project. The more clearly defined the problem and solution, the more clearly identifiable will be the best solution. In the case of a large-scale project, to best clarify the problem and future state, the organization must connect the subject matter experts (SMEs) involved with the problem, the information technology (IT) experts that will install and manage the software, procurement staff and management in a selection group.

This team will then work to identify the stakeholders that will be affected by the software. In the case of a specialized data research tool, there may only be a couple of end users, whereas an enterprise resource planning (ERP) system may have thousands of users within the company. The roles of these users within the new software solution must be well understood and defined. The people that will be using the tool every day must be given input as to how it will work best for them while meeting the goals of the organization. Ranking their priorities as critical, desirable, or nice-to-have will help focus the team on the most impactful solution (Pendergraft & Blakely, 2010).

Technical specifications of the new software tool must be understood and interoperability with the appropriate existing software tools within the company must be ensured. If an ERP

system needs to connect to an inventory management solution, integrations between the tools

will be needed. If these integrations do not exist between the two tools, a third-party developer

may be needed to build them.

The selection group must conduct market research to learn about options and the cost of

each option (Krouse, 1999). There may be different approaches depending on what is pre-

existing in the organization's software environment. The financial state of the organization may

dictate whether a best-in-class solution is chosen or simply one that meets a few crucial needs. A

Request for Information (RFI) may be issued by the group to prospective software vendors. An

RFI is a formal inquiry for information regarding capabilities around an organizational

requirement for a product or service (Krouse, 1999). Collecting information from multiple

vendors can help the organization better understand their own requirements in terms of what is

available in the marketplace. A previously unknown solution to the problem may present itself in

this stage. Sometimes another method than purchasing software can be uncovered to solve the

original problem.

Another consideration in the selection methodology is the long-term return on investment

of the project. The group needs to understand the length of time the software will be needed as

well the as costs in time and effort surrounding switching from the current system or process to

the new software (Sieverding, 2008). Is this a long-term investment or a short-term purchase to

achieve an immediate goal or advantage?

Purchasing

For large and complex purchases, once the decision is made to move forward with a

software tool, a Request for Proposal (RFP) can be issued to vendors that have acceptable

solutions. The RFP formally inquires about specific products that meet the need and their

associated pricing. The RFP document must clearly state the requirements decided upon by the

group. The more clearly the RFP details the different facets of the need, the more accurate

proposals and pricing will be received by the vendor. The RFP also allows for a vendor to answer

questions and provide information about themselves in a format that can be numerically scored

by the team. This allows for an empirical evaluation of the vendor capabilities in comparison to

each other and will help in the decision-making process. The team compares the responses from

the vendors and chooses the best one to meet their needs.

The RFP process should help the organization uncover not only the best vendor, but the

best licensing model with which to purchase the software. Different vendors have different

pricing structures that can be confusing. The selection group must work with the vendors to

understand their approach to licensing and ensure it fits the needs of the organization. One

example of this is the difference between a named user license and a concurrent user license. A

named user license is assigned to a specific individual within the company. Only they can use

that software licenses. A concurrent user license is assigned to the organization and while anyone

can use it, only one person may use the license at a time. If the organization expects ten different

people to use a license but at different times, it is very important to understand this difference

when reviewing the vendor's proposal. Lack of compliance with the vendor's licensing

agreement can make the organization liable to purchase more licenses than it originally wanted.

Upon selection of a vendor, the organization can begin negotiating a contract agreement

for the software. Depending on the scale of the purchase, there may be some leverage to

negotiate discounted pricing or better terms concerning warranties, implementation provisions or

ongoing maintenance costs (Pendergraft & Blakely, 2010).

For smaller purchases, a purchase order or a credit card may be used to secure a single license of a low-cost product. Even in the case of small purchases, organizations need to obtain copies of the software license agreements and understand the terms of use to ensure compliance. Because of the minimal cost, there is often no leverage to negotiate these license terms to the advantage of the purchasing organization. In that case, an understanding must be documented that the benefit of purchasing the low-cost software is worth the risk to the organization of a potentially less than advantageous licensing arrangement.

Implementation

Depending on the complexity of the product and the organizational processes that it is automating or replacing, implementation of a software product can be a significant undertaking. For large and complex implementations, an organization may contract with an implementation consultant as an expert advisor to help them implement their new product and maximize the value of their investment. These consultants can act as project managers and handle the entire process for the organization, or they can be assigned small pieces of the task where the organization does not have expertise.

A big piece of any implementation is change management and patiently building a culture that will support the transition (Glass, 2017). In the case of a new software tool that is replacing one or more incumbent systems, the current group of users must be trained in the use of the new software. Often, those presenting the new solution will meet resistance from the user group as they do not want to change the way they do things and do not see value in the new tool. Hence, one of the most important pieces of implementation is helping the users see the value in the change and encouraging them to embrace the learning process of the new system. The

effectiveness of training improves when the trainees are motivated to learn and excited about the change.

Intellectual Property and Risk

As discussed in the selection methodology section, software licensing must be understood and adhered to in order to mitigate compliance risk. The purchasing organization must understand exactly what it has purchased and how those licenses may be allocated within their own environment or accessed externally in the case of SaaS licenses. This necessitates continual monitoring by the IT staff to add and remove users as needed.

The selection group will need to identify and understand the risks involved with a software purchasing approach. Software that is purchased and stored on premise will require more maintenance and security precautions as it is now owned and managed by the organization. By using a SaaS approach, the organization has less to worry about in terms of upgrades and security as the software is owned and hosted by the vendor. That said, the purchasing organization must contractually ensure the safety of its data through contract data. In addition, a warranty that the software is defect-free and will perform the agreed upon tasks should be included in any agreement (Brennan, 2003).

The organization is responsible for maintaining and securing the software that is physically present on its owned equipment. In the same way, organizational data must be secured when on owned equipment as well. Data breaches via hacking or careless use can be very costly to the organization (Thomson, 2018).

Conclusion

Software can be a complex and confusing product to source, understand and purchase. It is available as something to be owned, licensed or rented as a service. Its nature has evolved throughout its relatively short history and will continue to do so as technical advances continue to grow in leaps and bounds. Wise organizations will first seek to understand and define the problem they hope to solve with software before beginning the process to purchase it. They will project the long-term consequences of their purchase into the future and consider the pros and cons. Since licensing models will continue to evolve as software companies discover new ways to monetize their products and services, so must purchasing organizations continue to learn how to leverage licenses to meet their needs while remaining compliant with intellectual property rights. Wise organizations will continue to evolve their change management practices to ensure employee buy-in to new technologies that make their organizations more productive and efficient. As electronic security risks proliferate in the modern world, organizations will continue to use software to manage risk and secure their reputations. They will stay curious and proactive as new technologies emerge in order to remain competitive in the global economy.

References

Baker, B. (2019). Digital IT Acquisition Professional: The Federal Standard for Digital Services Procurement. *Contract Management*, 59(1), 60-68.

Brennan, T. (2003). Purchasing Software: What Corporate Counsel Need to Know. *Intellectual Property & Technology Law Journal*, 15(8), 6-9.

Brereton, P. (2004). The Software Customer/Supplier Relationship. *Communications of the ACM*, 47(2), 77-81.

Bretthauer, D. (2002). Open Source Software: A History. *Information Technology and Libraries*, 21(1), 3-10.

Con Diaz, G. (2015). Embodied Software: Patents and the History of Software Development, 1946-1970. *IEEE Annals of the History of Computing*, 37(3), 8-19.

Dungan, C. Peter. (2010). Less is More: Encouraging Greater Competition in Computer Software Procurement by Simplifying the DFARS Licensing Scheme. (Defense Federal Acquisition Regulations Supplement). *Public Contract Law Journal*, 39(3), 465-482.

Glass, E. (2017). You Can't Just Flip a Switch. *ISHN*, 51(9), 28-29.

Jones, R. (2016). Select Your Next Accounting Software Without Vendor Bias. *Contract Management*, 56(4), 34-37,39,41-45.

Kauffman, R., & Tsai, J. (2009). The Unified Procurement Strategy for Enterprise Software: A Test of the "Move to the Middle" Hypothesis. *Journal of Management Information Systems*, 26(2), 177-204.

Krouse, Michael. (1999). Eight Steps to Successful Hardware and Software Procurement. *Healthcare Financial Management*, 53(6), 60-4.

Ladd, D. (2006). A Software Procurement and Security Primer. Security & Privacy, *IEEE*, 4(6), 71-73.

Mcdonnell, Evan. (2013). What Crime Scene Investigators Would Find if They Evaluated Major Software Acquisitions. *Contract Management*, 53(3), 54-59,61.

Michler, Carla. (2005). The Procurement Decision - "Open" or "Closed" Source Software? *Deakin Law Review*, 10(1), 261-269.

Pendergraft, Lee, & Blakely, Alan F. (2010). Ten Steps for Evaluating and Selecting Software

and Service Providers: Investing Resources in Preparation for Purchasing Software and Services is the Best Way to Ensure That Your Organization Gets the Best Service and a Quality Product. (Business Matters). *Information Management Journal*, 44(1), 40.

Putnam, D. (2018). Selecting the Right Vendor for Outsourced Software Projects. *Contract Managemen*t, 58(8), 12-19.

Samuelson, P. (2011). Legally Speaking, Do You Own the Software You Buy? *Communications of the ACM*, 54(3), 26-28.

Sieverding, M. (2008). Choice in Government Software Procurement: A Winning Strategy. *Journal of Public Procurement*, 8(1), 70-97.

Tate, J. (1999). How to Select New Business Software. *Management Accounting*, 77(4), 52-54.

Thomson, L. (2018). Third-Party Vendors Can Be a Weak Link: ABA Vendor Contracting Cybersecurity Checklist Focuses on the Procurement Process to Strengthen Security Protections. *Scitech Lawyer*, 14(3), 36-37.

Williams, R., & Pollock, N. (2007). Technology Choice and its Performance: Towards a Sociology of Software Package Procurement, *Information and Organization* 131-161.