Chapter 16
Advances in Technology Project Management:
Review of Open Source Software Integration

Maurice Dawson
University of Missouri – St. Louis, USA

Brian Leonard
Alabama A&M University, USA

Emad Rahim
Oklahoma State University, USA

ABSTRACT

As organizations must continually drive down costs of software-driven projects, they need to evaluate the Systems Development Life Cycle (SDLC) and other software-based design methodologies. These methodologies include looking at software-based alternatives that could save a significant amount of money by reducing the amount of proprietary software. This chapter explores the use and integration of Open Source Software (OSS) in software-driven projects to include in enterprise organizations. Additionally, the legalities of the GNU General Public License (GPL), Lesser General Public License (LGPL), Berkeley Software Distribution (BSD), and Creative Commons are explored with the integration of these OSS solutions into organizations. Lastly, the chapter covers the software assurance and cyber security controls to associate with OSS to deploy a hardened product that meets the needs of today's dynamically evolving global business enterprise.

DOI: 10.4018/978-1-4666-6473-9.ch016

APPROACH

The authors reviewed multiple Linux distributions and their uses. Reviewed in depth were the copyrights and open sourcing legal implications.

IMPRESSION

As indicated through legal case reviews, there are some very valuable benefits to open source software, in that it allows for collaboration in the development of new software and technology that can undoubtedly spur innovation and improve many processes and functions that individuals and businesses in our society rely on every day. Consequently one purpose of the GNU GPL is to protect and preserve individual rights and the creativity of others while at the same time providing a benefit and contributing to society at large. OSS must be considered in the development process as it is essential in overall license cost reduction with the ability to reuse already constructed software.

PROJECT MANAGEMENT

There are numerous perspectives regarding the concept of project management as this is a field with many employment opportunities in various industries such as defense or aerospace (Dawson & Rahim, 2011). Thus, the definitions generated by these perspectives also vary, according to the context in which it is discussed. However, the purpose of most project management activities is generally similar. Project management is a way of managing and organizing corporate resources so the available resources can generate the completion of a project within given scope, time, and resource constraints (Wideman, 2001).

The understanding behind project management also accounts for the definition of a project. A project is a unique endeavor performed to create certain products, services, or results (Project Management Institute, 2009). This definition is dissimilar to the definitions of process and operation due to several factors. The easiest to define is the time-constraint factor. A project performs the work necessary to complete activities within a limited amount of time, while processes and operations generally account for on-going continuous effort. A project aims to produce a single or a group of products, services, or results and the chain of activities are terminated once these are produced. Thus it is important to understand the acquisition of Information Technology (IT) and Information Systems (IS) in project management (Rahim & Dawson, 2010).

Software Design Methodologies

The SDLC is a process for planning, creating, testing and deploying ISs (Avison & Fitzgerald, 2003). Requirements are an impact factor as they feed the development and serve as an important prerequisite to development. The SDLC is a modified waterfall method as when objectives are not met then the process is to move backward but the goal is to continually move forward into the next process steps such as system deployment. Another design methodology is agile software development. Agile is based on iterative and incremental development, in which requirements and solution evolve through collaborating teams (Cockburn, 2002). In agile it is essential to understand the people factor to ensure success (Cockburn & Highsmith, 2001). A modified agile methodology is Scrum which is an interactive and incremental software development framework (Rising & Janoff, 2000). All methodologies described allow for code reuse and the integration of OSS. As design methodologies continue to grow so does the need for quicker development. To do this effectively one would need to consider using the option of code reuse.