Measuring the Maturity of Open Source Software for Digital Libraries: a Case Study of DSpace

A Thesis submitted to the
Cochin University of Science and Technology
for the award of the Degree of

DOCTOR OF PHILOSOPHY

In

Library and Information Science

Under the
Faculty of Technology
by

Surendran Cherukodan
(Reg.No. 3540)

Under the supervision of

Dr. Humayoon Kabir S.

Department of Computer Applications
Cochin University of Science and Technology
Cochin-22, Kerala, India.
April 2015

Certificate

Certified that the study presented in this thesis entitled **"Measuring the Maturity of Open Source Software for Digital Libraries: a Case Study of DSpace"** is a bonafide work done by Mr.Surendran Cherukodan, under my guidance in the Department of Computer Applications, Cochin University of Science and Technology and this work has not been included in any other thesis submitted previously for the award of any degree. Also certified that all the relevant corrections and modifications suggested by the audience during the pre-synopsis Seminar and recommended by the Doctoral Committee of the candidate have been incorporated in the thesis.

Kochi
April 7, 2015

Dr.Humayoon Kabir S.
(Supervising Guide)

Declaration

I, Surendran Cherukodan, hereby declare that the thesis entitled **"Measuring the Maturity of Open Source Software for Digital Libraries: a Case Study of DSpace"** is the outcome of the original work done by me under the guidance of Dr. Humayoon Kabir S., Associate Professor, Department of Library and Information Science, University of Kerala, and that the work did not form part of any dissertation submitted for the award of any degree, diploma, associateship, or any other title or recognition from any University/Institution.

Kochi Surendran Cherukodan

April 7, 2015

Acknowledgements

Doing research is great that is what I learned from my experience with this study. It is great in the sense that we encounter with many ideas, more resources and numerous scholars. Doing research instills a sense of obligation and contribution. I acknowledge the intellectual support of my research guide Dr. Humayoon Kabir, S. throughout the period. Dr. K V Pramod and Dr. B Kannan of the Department of Computer Applications and Dr. G.Santhosh Kumar of Department of Computer Science extended valuable insights for my study. I express my sincere thanks to them. Dr. M. Kailasnath of the International School of Photonics and Dr. V.Sivandandan Achari of School of Environmental Science and Dr. Sebastian Rupert Mampilly of School of Management Studies extended intellectual support. I express my gratitude to them. I thank for the support of my colleagues including Thanooja, Dr.Veeran Kutty, and Dr. Shibu Ray, Jinu Sudhakaran and Babitha. Mr. Ayyappan R and Anthru M. also helped me a lot. Thanks are due to the staff of Dept. of Computer Applications who offered excellent service.

The support of my parents and family in accomplishing great things in life is remarkable. I always received encouragement and enthusiasm for learning from my father Cherukodan Kunhikannan and mother Vellampurath Cheeru. My wife Sheeja, daughter Alaina and father and mother in-laws helped me to work long hours in the course of study. Dr. N.P Hafiz Muhammed, Dr.Yasin Ashraf, both from Farook College, M.N Mazumdar (late) and S.R Talukdar both from New Delhi, are very dear to me as they instilled the quest for higher learning. Finally, I thank all those who supported me one way or the other to accomplish this work.

Cochin-22

7-4-2015 Surendran Cherukodan

Contents

List of Figures

List of Tables

List of Abbreviations

API	Application Program Interface
BSD	Berkeley Software Distribution
CD	Compact Disc
CDDL	Common Development and Distribution License
CDS-Invenio	CERN Document Server – Invenio
CERN	European Organization for Nuclear Research
CMM	Capability Maturity Model
CMS	Content Management System
DoD	Department of Defence
DoKS	Document and Knowledge Sharing application
DL	Digital Library
ETD	Electronic Thesis & Dissertation
FEDORA	Flexible and Extensible Digital Object and Repository Architecture
FOSS	Free and Open Source Software
GCC	GNU Compiler Collection
GDB	GNU Symbolic Debugger

GPL	General Public License
GNU	GNU Not Unix
HP	Hewlett-Packard
IBM	International Business Machines
ICT	Information and Communication Technology
ILS	Integrated Library System
IR	Institutional Repository
ISRO HQ	Indian Space Research Organization Head Quarters
IT	Information Technology
JSP	Java Server Pages
MARC	Machine Readable Catalogue
MD5	Message Digest 5
METS	Metadata Encoding and Transmission Standards
MIT	Massachusetts Institute of Technology
MPL	Mozilla Public License
NMLIS	New Millennium LIS Professionals
NRCFOSS	National Resource Centre for Free/Open Source Software
OAI-PMH Harvesting	Open Access Initiative Protocol for Metadata
OCLC	Online Computer Library Center
OPAC	Online Public Access Catalogue
OPALS	Open-source Automated Library System

OPENDOAR	Open Directory of Open Access Repositories
OSI	Open Source Initiative
OSS	Open Source Software
OSS4LIB	Open Source Systems for Libraries
OSMM	Open Source Maturity Model
PMB	Php MySQL Bibli
QSOS	Qualification and Selection of Open Source Software
ROAR	Registry of Open Access Repositories
RSS	Rich Site Summary
SDK	Software Development Kit
SQL	Structured Query Language
SWCMM	Software Capability Maturity Model
TAM	Technology Acceptance Model
UK	United Kingdom
UNESCO	United Nations Educational, Scientific and Cultural Organization.
URL	Uniform Resource Locator
WG-OSMM	Woods and Guliani's Open Source Maturity Model-2005
XML	Extensible Markup Language

Chapter 1 Introduction

1.1 Open Source Software

Computer software is the single most important technology on the world stage (Pressman, 2005, p.1). Software plays a crucial role in access to information and knowledge (UNESCO, 2010). In the modern software development world, the two universally accepted and followed ways of software development are proprietary and open source (Isitan, 2011). Proprietary software is computer software that is the exclusive property of its owners/creators and bears limits against uses, such as modification, sharing, studying, redistribution, or reverse engineering. Generally, the source code of proprietary software is closed. Proprietary software may also be called closed source software or commercial software. Open Source Software (OSS) on the contrary, are owned by a community of users who belong to diverse locations in the world with provision of access to source code for ensuring freedoms to run, study, change and to redistribute copies with or without changes. The history of OSS began with the early stages of computer and software development.

In the 1960s, there were no commercial software solutions and computer scientists and researchers relied on free and openly shared software code for their work. Operating system software was separated as a product from hardware in the 1970s (Valimaki & Oksanen, 2005). In the 1980s, software companies began to control source code. The restriction of access to the source code was a theme of debate and deliberation. The movement for software freedom led by Richard Stallman and his Free Software Foundation in 1985 brought a radical change in the software landscape. Stallman coined the term “free software" to mean software that ensured four essential freedoms; the freedom to run, study, redistribute copies and distribute modified versions of software. Access to source code is a prerequisite of any free software. The term “free” does not necessarily imply free of cost. The

GNU operating system developed by Stallman was the first model of the free software. Stallman also contributed significantly to the widespread adoption of open source by authoring a number of well-known and highly used development tools, including the GNU Compiler Collection (GCC), the GNU symbolic debugger (GDB) and GNU Emacs (Ratib & Rosset, 2006). Until 1991, The GNU was not ready for production use. In 1991, Linus Torvalds developed a Unix-compatible kernel and called it Linux. Around 1992, Linux was combined with GNU system to form a complete free operating system called GNU/Linux.

The term Open Source Software (OSS) was coined by Eric Raymond in 1997 to help market and create an acceptable face to the movement within the commercial world (O'Neill, 2012). OSS may be defined as software whose source code is published and made available to the public, enabling anyone to copy, modify and redistribute the source code without paying royalties or other fees, depending on the exact form of rights given with the software (Forge, 2006). The United States Department of Defence (DoD) defines OSS as software for which the human-readable source code is available for use, study, re-use modification, enhancement, and re-distribution by the users of that software (DoD, 2009). In 1998, the Open Source Initiative (OSI) was established by Bruce Perens and Eric Raymond. The OSI has an extensive ten-point definition of OSS. They include;

- Free Redistribution

The license shall not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources. The license shall not require a royalty or other fee for such a sale.

- Source Code

The program must include source code, and must allow distribution in source code as well as compiled form. Where some form of a product is not distributed with source code, there must be a well-publicized means of obtaining the source code for no more than a reasonable reproduction cost preferably, downloading via the Internet without charge. The source code must be the preferred form in which a programmer would modify the program. Deliberately obfuscated source code is not allowed. Intermediate forms such as the output of a pre-processor or translator are not allowed.

- Derived Works

The license must allow modifications and derived works, and must allow them to be distributed under the same terms as the license of the original software.

- Integrity of the Author's Source Code

The license may restrict source-code from being distributed in modified form only if the license allows the distribution of "patch files" with the source code for the purpose of modifying the program at build time. The license must explicitly permit distribution of software built from modified source code. The license may require derived works to carry a different name or version number from the original software.

- No Discrimination against Persons or Groups

The license must not discriminate against any person or group of persons.

- No Discrimination against Fields of Endeavor

The license must not restrict anyone from making use of the program in a specific field of endeavor. For example, it may not restrict the program from being used in a business, or from being used for genetic research.

- Distribution of License

The rights attached to the program must apply to all to whom the program is redistributed without the need for execution of an additional license by those parties.

- License Must Not Be Specific to a Product

The rights attached to the program must not depend on the program's being part of a particular software distribution. If the program is extracted from that distribution and used or distributed within the terms of the program's license, all parties to whom the program is redistributed should have the same rights as those that are granted in conjunction with the original software distribution.

- License Must Not Restrict Other Software

The license must not place restrictions on other software that is distributed along with the licensed software. For example, the license must not insist that all other programs distributed on the same medium must be open-source software.

- License Must Be Technology-Neutral

No provision of the license may be predicated on any individual technology or style of interface.

It is also important to note that there are several things fiddled as OSS. It is not shareware, public domain software, freeware, or software viewers and readers made freely available without access to source code (Bretthauer, 2001). The term Free Software and OSS are used interchangeably and on many situations it will be considered synonymous (Stahl, 2005)

1.2 Open Source License

A license can be thought of as the permission of the owner of property to use that property (Classen, 2007, p.11). Software licensing is a legal matter. Since software being a unique technology and is a work of authorship that is fixed in a tangible medium of expression, it is entitled to protection under copyright law, patent law and trade secrete law (Classen, 2007, p.5). All OSS are released under a license that defines the terms and conditions of their use. The OSS licenses grant general access to the software and its source code, as well as the right to read, modify, improve, redistribute and use it. The reason behind the stronger positions of OSS is due to its open source licensing policy that guarantees these rights (Valimaki & Oksanen, 2005). The OSS are released under a variety of licenses. The OSI has a list of 83 OSS licenses under different categories. The following are the nine licenses under the popular and widely used category:

- Apache License, 2.0 (Apache-2.0)
- BSD 3-Clause "New" or "Revised" license (BSD-3-Clause)
- BSD 2-Clause "Simplified" or "FreeBSD" license (BSD-2-Clause)
- GNU General Public License (GPL)
- GNU Library or "Lesser" General Public License (LGPL)
- MIT license (MIT)
- Mozilla Public License 2.0 (MPL-2.0)
- Common Development and Distribution License (CDDL-1.0)
- Eclipse Public License (EPL-1.0)

The OSS licenses are generally described under two groups; copyleft and permissive (non-copyleft). Copyleft licenses are highly restrictive and insist that modified versions of the programme must be free software as well. GNU

General Public License is a best example of copyleft license. Permissive licenses are less restrictive and allow modifications to remain closed-source. Berkeley Software Distribution (BSD) is an example of permissive license type.

1.3 Open Source Development Model

Since 1998, the OSS movement has become a revolution in software development (Bretthauer, 2001). The OSS development model differs from the closed source or proprietary model. OSS usually evolves through community developers composed of individual programmers who are widely spread geographically, often never meeting with each other. Communication is carried out by e-mail and through newsgroups. Thus, OSS allows engagement, interaction, feedback and sharing of content (software) at the user level and flourishes as a result of this (Sen, Singh & Borle, 2012). The OSS model provides interesting tools and processes with which people can create, exchange, share and exploit software and knowledge efficiently and effectively (UNESCO, 2010). It is considered more efficient method of software development because OSS avoids the inefficiencies of a strong intellectual property regime (Subramaniam, Sen & Nelson, 2009). The OSS model produces better software than the traditional closed model (Ratib & Rosset, 2006). Raymond (2001) compared OSS development to a bazaar model, where anyone has the right to join and contribute while commercial software followed a hierarchical cathedral style. OSS is described as product development without manufacturers (Von Hippel, 2001). Open source phenomenon represents a radical change in the software landscape (Fitzgerald, 2006). The more persuasive argument for open source is the ability to localize and customize (Brewer et al., 2005).

1.4 Adoption of OSS

Sourceforge.net, the web based source code repository for OSS claims the creation of over 430,000 OSS projects involving 3.7 million (37 Lakhs) registered users and over 480, 0000 downloads a day (Sourceforgenet, 2014). In 2000, the Sourceforge.net web site listed 2370 OSS projects and 15,060 OSS registered users, and, by 2007, it listed 100,000+ projects with 1 million registered users (Elliott & Scacchi, 2008). The widespread adoption of OSS has generated immense interest among academics, who want to understand and explain various aspects of this phenomenon (Subramaniam, Sen & Nelson, 2009).

There are OSS applications for all spheres of human life. The Sourceforge.net lists the OSS projects under ten major heads which include Audio and Video, Business, Communications, Development, Home and Education, Games, Graphics, Science and Engineering, Security and Utilities and System administration. Among the ten, the Development sector, that includes Software Development, Text Editors, Database, and Data formats, has more number of OSS. The open source market is large and growing for application domains such as Web server (such as Apache), server operating systems (such as Linux Server), database server (such as MySQL), electronic mail client (such as Sendmail), and Internet browser (such as Firefox) (Nagy, Yassin & Bhattacherjee, 2010). OSS have changed the computer industry in many ways and will undoubtedly continue to do so (O'Neill, 2012).

OSS is widely used and is becoming a significant and irreplaceable part of the software engineering community (Zhang, 2007). Many organizations have recently increased their adoption of OSS (Lundell, Lings & Siberfeldt, 2011). The shift to the use of OSS has been perceived as to reap the direct benefits of

lower software costs and further indirect benefits such as greater adherence to open standards, more choice of vendor and service supplier, and efforts to establish flexible incremental architectures (Shaikh & Conford, 2011) .

OSS has also gained interest at policy and managerial levels (Holck, Larsen & Pedersen, 2005, p.289). It gives customers control over the technologies they use, instead of enabling the vendors to control their customers through restricting access to the code behind the technologies (Young, 2001).

OSS has made a strong impact as it has been adopted by many businesses, educational institutions, government departments, and individual users (De, 2009). Both developed and developing nations of the world make use of OSS applications as Governments of these countries support and encourage it (Mutula & Kalaote, 2010). Developing regions have a default preference for open source software on the premise that it is free (Brewer et al., 2005) and that it may work well with limited computing or similar assets (Nash, 2010). Governments are increasingly motivated to adopt open-source products to reduce the expenditure of scarce taxpayer money. Some governments (e.g. Argentina) have experimented with moving entirely to an open source model (Krishnamurthy, 2003). The Government of United Kingdom recognized the potential benefits of OSS and is committed to increasing the adoption of open source solutions across government, where it offers best value for the taxpayer (Cabinet Office, UK, 2012). In Sweden, the cross-government team who performed a feasibility study on OSS stated that OSS in many cases are equivalent to, or better than, commercial products (Cabinet Office, UK, 2012). There are countries including China, India, Norway and Germany where OSS is more established and twenty four countries have passed or considering laws encouraging OSS use (Applewhite, 2003). As a part of the ADELE project aimed at computerizing much of the country's administration by 2007, France installed OSS on its desktops. Singapore is offering tax breaks to companies

that use the OSS. Germany has reached an agreement with IBM aimed at offering discounts on IBM machines with pre-installed Linux (Comino & Manenti, 2005) The Life Insurance Corporation (LIC), one of the largest insurers in India saved about Rs 420 million by adopting OSS (De, 2009). Hence, OSS is increasingly seen as a tool that can help governments achieve effective service delivery because of its low cost compared with commercial software (Mutula & Kalaote, 2010).

OSS development has been the basis for the creation of many of today's most innovative products and solutions (Ebert, 2008) and is seen as an exemplar of a successful approach for knowledge creation (Nash, 2010). OSS plays an increasingly important role in the IT sector (Aicherning, 2006). OSS can be found in digital video recorders (Tivo), telephones, personal digital assistants (PDAs), watches, networking hardware, MP3 players and automobiles (Stahl, 2005). OSS applications are first, second or third-rung products in terms of market share in several markets, including web servers, server operating systems, desktop operating systems, web browsers, databases, e-mail and other ICT infrastructure systems (Ghosh, 2006). Major Internet sites are built on an open source platform, and that much of the Internet's infrastructure is based on OSS (Hendrickson, OReilly & Magoulas, 2012, p.35). OSS powers many of the web sites on the Internet, corporate computer servers used for research and development, and a plethora of new gadgets that have broad appeal (Stahl, 2005). Nearly every network appliance and custom hardware box sold in the open market today is built mostly or entirely using OSS (MITRE, 2011). Successful OSS projects like the Linux operating system and the Apache web server have demonstrated the strength of the OSS development process, where self-motivated users and developers share knowledge via the Internet (Aicherning, 2006). Internet Explorer is an example of a notable Microsoft utility that is based heavily on OSS and all modern Apple products, from Macs

to iPods and iPhones, are built on OSS with a thin layer of customized software on top (MITRE, 2011). Examples of notable adoptions include Amazon and Yahoo's use of Perl, Orbitz' use of Linux and Apache and Google's usage of Linux (Krishnamurthy, 2003).

OSS offer useful savings in time, money, and resources (Barve & Dahibhate, 2012). Many organizations have caught on to open source software and realized significant cost savings in technology expenditure as a result (Nagy, Yassin & Bhattacherjee, 2010). As open source technology continues to gain status, more organizations are migrating from proprietary to OSS (Amollo, 2013).

Since OSS has achieved significant advances (Poulter, 2010), in various fields, several studies were attempted to understand the factors influencing the adoption of OSS. The cost associated with OSS is found to be the major factor of adoption followed by OSS's compatibility with organizational needs (Marsen & Pare, 2013). The other attractions of OSS include security, technical superiority, customization, audit and support from the community as well as from vendors. The OSS model leads to better quality assurance, more frequent releases, involvement of community in development, and the ability for customers to try before they buy (Boulanger, 2005). OSS ensures contracting freedom, greater independence from suppliers, the options of portability across a wide variety of platforms and its independence from any other related products. The promise of open source, according to the Open Source Initiative, is better quality, higher reliability, more flexibility, lower cost, and an end to predatory vendor lock-in.

1.5 Adoption of OSS by Libraries

Librarians began to recognize the value of OSS in 1990s. Daniel Chudnov, a librarian at Yale medical library whose article in the August 1999 issue of Library journal titled "Open source software: The Future of Library Systems?" invited the attention of the mainstream library community towards OSS. He noted that librarians were not ready to adopt OSS and by doing so they were missing an important part in shaping OSS culture (Chudnov, 1999). Chudnov listed many library related OSS projects and gave a list of URLs so that librarians interested in learning more could research the open source phenomenon for themselves. He maintained the OSS4LIB (Open Source Systems for Libraries) web site and email discussion list along with Gillian Goldsmith Mayman. The OSS4LIB web site (http://www.oss4lib.org) serves as a clearing house for information on open source development within the library system. Tennant (2000) also observed OSS better than proprietary software for libraries as it may be altered to meet their needs, and such alterations may benefit other libraries as well. Currently, several OSS are available for libraries to provide new value added services to end-users while handling large volumes of library data (Barve & Dahibhate, 2012). The process learning, selection and adoption of OSS in libraries is a modern trend throughout the world. The observation of Smith (2002) that "whenever a new information technology emerges, librarians invariably appropriate it and adapt it to the library setting" is quite appropriate to understand the trend towards OSS in libraries. Now, library and information science literature cover more articles on OSS. Conferences and workshops are being organized by professional associations and educational institutions for imparting training and education to library professionals. Many library professionals became resource persons on OSS for libraries. Funding

agencies supported many events that highlighted OSS in libraries. Research was also conducted on OSS in libraries.

The reason behind the adoption of OSS by large companies is due to their technical merits and their ability to meet stringent requirements (Krishnamurthy, 2003). Similarly, OSS was advocated for libraries for its quality of portability (Buchanan & Krasnoff, 2005), ability to solve problems for libraries of all types (Bisson, West & Eby,2007), zero maintenance features (Hasan, 2009), to save and preserve library data for future and participate in the movement of sharing information globally with open standards and open formats (Barve & Dahibhate, 2012) and to provide a cost effective automation solution regardless of location, size and budget (Amollo, 2013).

Categorization and listing of library-related OSS include applications for document delivery, Z39.50 clients and servers, systems to manage collections, MARC record readers and writers, integrated library system applications, digital library software, digital archiving software, next generation OPAC software, electronic journal archiving, etc. (Barve & Dahibhate, 2012).

1.6 Open Source Software for Digital Libraries

Digital libraries (DLs) have achieved a fundamental role in our knowledge society (Candela, Castelli & Pagano, 2011, p.1) by facilitating the creation, organization and management of multimedia digital content and collections, and providing search, retrieval and other information services over computer networks and other electronic media (Hoe-Lian Goh et al., 2006). DLs are systems that consist of specially designed hardware and software perfected by trained professionals with the mission of providing access to various kinds of digital objects to a community of users. The real value of digital libraries rests

on their ability to alter the way individuals, groups, organizations etc., behave, communicate, and conduct their affairs (Griffin, 1998).

DLs consist of several technology and standards to collect, organize, share and preserve digital materials over the long term. DLs have become a major part of the mainstream library landscape that provides an integrated set of services (Krishnamurthy, 2008). DLs have greatly evolved during the last few years (Tramboo, Shafi & Gul, 2012). Currently DL projects are very active at global, national and institutional level. The Universal Digital Library Project (www.ulib.org) has digitized lakhs of books that have immense educational, cultural and artistic value. The World Digital Library (www.wdl.org) project of the Library of Congress, United States with the support of UNESCO makes available materials from all countries and cultures. The Digital Library of India (www.dli.ernet.in) project holds more than four lakhs books from various parts of the country. DLs established at various universities across the globe add new dimensions and values to the concept of DLs as an integral part of modern society.

The OSS for DLs is a major portion of total OSS application in libraries. There are many OSS packages available for organizations and individuals to create DLs (Hoe-Lian Goh, et al., 2006). The prominent among them include; DSpace, (www.dspace.org), developed by MIT Libraries and Hewlett Packard, EPrints (www.eprints.org), developed by the University of Southampton, Fedora (http://fedorarepository.org) by Cornell University and Greenstone (www.greenstone.org) by the University of Waikato.

The statistics produced by the OpenDOAR (Directory of Open Access Repositories) showed that the vast majority of existing institutional repositories (IRs) were built upon OSS (Jones, Day & Ball, 2009). The investigator's examination of OpenDOAR after five years validated this claim.

As of December, 2014, ROAR (Registry of Open Access Repositories) and Open DOAR list 3292 and 2778 repositories respectively in the world. The repository map (maps.repository66.org) by Stuart Lewis provides a list of 3045 DLs in the world. The analysis of type of software being used for DLs revealed that the majority of DLs were built using OSS. The distribution of software among repositories has been given in the fourth chapter of this thesis.

1.7 DSpace Software

DSpace is the major DL software in the OSS domain used by academic, non-profit, and commercial organizations in the world. The registries that list DLs in the world show that more DLs are created by DSpace. The preference for a particular OSS for DL has been subjected to study and research. The factors behind the adoption of DSpace over other DL software have been pointed out by various authors. Chapter 2 of this study provides detailed discussion on these aspects. The process of installation and customization of DSpace is easy and the software has a vibrant community of users and developers all over the world. The vendor support for DSpace adds more opportunities for organizations to deploy the software. The tradition of regular release, integration of additional features, support for standards, lengthy documentation, live email forums etc. make DSpace acceptable for more institutions.

1.8 Maturity of Open Source Software

The literature survey on OSS reveals that the adoption of OSS is intensive in several sectors of human life. Librarians are also in the forefront of applying OSS to libraries. There are many OSS for libraries. However, all OSS projects are not created equal (Woods & Guliani, 2005, p.45). Choosing suitable

software from a large pool of OSS is a time-consuming, challenging and confusing process (Zhang, 2007). Hence the selection of OSS in libraries is a matter of great concern. Survey of literature reveals that the adoption of OSS in libraries is made generally through the comparison of features of software. While features and facilities of an OSS offer solutions for library requirements, the question on the future of OSS was left unattended. Since libraries adopt OSS for long term use and the systems contain valuable data, the future of OSS being used has great prominence. The future of software cannot by understood by studying features alone. Literature on OSS in libraries don't provide much attention on the future of OSS. Here comes the importance of maturity of the software.

The concept of maturity is understood differently by various fields of knowledge. Generally, maturity is defined as "the state, fact, or period of being mature" (Oxford dictionary of English, 2010, p.1093). When applied to human beings maturity is related to the stage of physical as well as psychological development. In agronomy, physiological maturity is defined as a period of advancement in the cycle of a plant when in a state of full or complete growth development or ripeness (Fageria, 1992, p.69). In management it is related to a product and seen as one of the stages of product life cycle. The product life cycle consist of four stages namely introduction, growth, maturity and decline (Anandan, 2009, p.93). Maturity of software is a concept universally recognized throughout the IT industry (Golden, 2005, p.71).

In software world, the term maturity is used to denote the quality and longevity of software both as a process and a product. Maturity of the software process signifies the strategies adopted by an organization in producing software whereas product maturity implies quality, reliability and longevity of

the finished software product. The maturity of software is measured through various maturity models. The history of software maturity models goes to 1986 when the Software Engineering Institute (SEI), a research and development centre, United States, developed the Capability Maturity Model (CMM) to help organizations improve their software process. A software process is defined as a set of activities, methods, practices, and transformations that people use to develop and maintain software and the associated products (Paulk, Curtis, Chrissis & Weber, 1993). All process maturity models are built on the premise that the quality of a system or product is highly influenced by the quality of the process used to develop and maintain it. Process maturity models were generally designed for proprietary software.

Since the software development process of OSS is entirely different from that of proprietary software, traditional maturity models cannot be applied to OSS. Hence, based on the success of CMM, various maturity models were developed for OSS. They include Navica Open Source Maturity Model (OSMM), Open Business Readiness Rating (OpenBRR), Cap Gemini Open Source Maturity Model, Qualification and Selection of Open Source software (QSOS) etc. The OSS maturity models generally focus on software as a product and share the concept that software products go through predictable stages of maturity. OSS maturity models try to understand the features of the software as well as the software environment. The software features include the extent and scope of documentation, quality of packaging, quality of code and design, quality of architecture, testing practices, integration with other products, support for standards and license type. The software environment denotes the leadership behind a project, project culture, size of the community, quality of end-user support, frequency of release of new versions of the software, the contents and quality of project's website, potential for

commercial conflicts and corporate commitment. Maturity is a word that captures how "grown up" a product is (Golden, 2005, p.72). A software product will be said to be mature if it has a full feature set, high quality, longevity in market, good support and exhibits robust behavior in error situations. (NRCFOSS, 2014). The maturity of a software is key to understanding how well suited a product is for a particular use as production use requires very mature products (Golden, 2005, p.73). Maturity of software is an important area of study and research both in proprietary and OSS sector. OSS is getting more and more attentions and market shares today. However, there are so many OSS out there and there seems to be no way to tell which one is the best (Zhang, 2007).

1.9 Need and Significance of the Study

Understanding maturity of OSS is important for several reasons. Being a powerful technological opportunity (Meeker, 2008, p.1) OSS help libraries to overcome budgetary restrictions, bring more flexibility and evolution toward new generation of technologies and achieve better interoperability with other systems (Macan, Fernandez & Stojanovski, 2012). OSS helps to attain more support options than those using proprietary software (Breeding, 2009).

The absence of concern for the maturity of OSS affects libraries many ways. The perception of lack of maturity has been reported as one of the factors of low adoption of OSS. (Gurusamy, 2011). The other factors include lack of awareness of OSS products (Vimal Kumar & Jasimudeen, 2012; Al Zeheimi et al, 2014), lack of quality assurance and direction (Macan, Fernandez & Stojanovski, 2012), challenges of sustainability (Schneider, 2009) and lack of high quality end user documentation (Yeates, 2005). The selection of a

suitable OSS from a pool of software is also a challenging task for librarians. The adoption of OSS in libraries is made generally through the comparison of different features of software. Many institutions depend on feasibility studies on OSS conducted by others. These studies provide little attention on the aspects of maturity. While concentrating on the features of the software they neglect the software environment. The features of an OSS do not guarantee the sustainability of the software. Since libraries adopt software for long-term use the longevity of software is highly important. This aspect can be understood only through measuring maturity of the software using appropriate models. The present study is attempted towards this direction

The study is attempted based on the following theoretical backgrounds.

- The adoption of OSS is slow in the world mainly due to the perception of a lack of maturity of OSS solutions (Sharma & Adkins, 2005).
- Maturity of OSS products was an enabler for OSS adoption as organizations tended to adopt mature OSS products (Gurusamy, 2011).
- Information technology adoption and use in the workplace remains a central concern of information systems research and practice (Venketesh & Davis, 2000).
- OSS present many possibilities for libraries (Morgan, 2002)
- All OSS projects are not created equal (Woods & Guliani, 2005, p.45).
- Many OSS projects, after a certain long period of evolution, stop evolving, and in fact become inactive (Khondhu, Capiluppi & Stol, 2013, p.61).
- OSS can appear unattractive and risky to some because there is no central point of control from which advice about the software package

and its future development can be sought (Damsgaard & Karlsberg, 2010).

- Lack of good information keeps library information professionals from embracing OSS technologies (Muller, 2010).

The present study is significant as it addresses the aspects of maturity of OSS for libraries and fills the research gap on the area. In this sense the study opens new avenues to the field of library and information science by providing an additional tool for librarians in the selection and adoption of OSS. Measuring maturity brings in-depth knowledge on an OSS which will contribute towards the perceived usefulness and perceived ease of use as explained in the Technology Acceptance Model theory. (Venketesh & Davis, 2000).

1.10 Statement of the Problem

The problem of the present study is entitled as "Measuring the Maturity of Open Source Software for Digital Libraries: a Case Study of DSpace".

1.11 Definition of Key Terms

The following are the definitions of key terms used in the study.

Measuring: According to The new international Webster's comprehensive dictionary of English language, (1996) measuring means to take or ascertain the dimensions, quantity, capacity etc., of, especially by means of a measure (p.790).

The Oxford concise English dictionary (2011) defines the verb "measure" as; ascertain the size, amount of degree of (something) by comparison with a

standard unit or with an object of known object (p. 886). In this study measuring is used to denote the act of ascertaining the maturity status of an OSS product with an instrument designed for it.

Maturity: Maturity is defined as the state or condition of being mature (The new international Webster's comprehensive dictionary of English language, (1996, p.787). In this study the word maturity is used to denote the longevity of software over the years. Software that is mature is supposed to be stable and durable and it will have strong leadership, wider community base, well documentation, end-user support and frequent releases at regular intervals incorporating new features.

Open Source Software (OSS): Software that meets the terms of the Open Source Definition of Open Source Initiative (OSI). To be open source, the software must be distributed under a license that guarantees users the right to read, redistribute, modify, and use freely.

Digital Libraries: A digital library is fundamentally a resource that reconstructs the intellectual substance and services of a traditional library in digital form (Seadle & Greifeneder, 2007). Digital libraries are organizations that provide the resources, including the specialized staff, to select, structure, offer intellectual access to, interpret, distribute, preserve the integrity of, and ensure the persistence over time of collections of digital works so that they are readily and economically available for use by a defined community or set of communities". (Digital Library Federation, 1998). A single, simple, stand-alone web page is probably not a digital library in any meaningful sense, any more than a single page or a single book is a traditional library (Seadle & Greifeneder, 2007). The study has used digital library as an umbrella term to include institutional repositories, digital repositories, archives, electronic thesis and dissertation and similar systems.

Case Study: Case study is an empirical inquiry that investigates a contemporary phenomenon within its real life context, especially when the boundaries between phenomenon and context are not clearly evident (Yin, 1994 p.13.)

DSpace: DSpace is the name of the software selected for the study. DSpace is an open source software for digital libraries/repositories/ETDs created jointly by Massachusetts Institute of Technology (MIT) and Hewlett Packard (HP) Lab in 2002.

The concept of DLs is also manifested through terms such as 'virtual library,' 'institutional repository,' 'digital repository', 'digital archives' or 'electronic library'. A search conducted on Google Trends (www.google.com/trends/) as of December 2014, revealed that digital library is the most sought after term worldwide. The term institutional repository (IR) was popularized by the idea of a system established in a university or higher education institution to manage and disseminate the digitized and the born digital materials produced by the members of a university. IRs were construed as the organizational commitment towards the stewardship of digital materials including long term preservation as well as organization and access or distribution (Lynch, 2003). The different terms like DLs and IRs are used to denote the same entities. Hence, this study has used the word DL as an umbrella term to denote all systems mentioned above. DL also covers Electronic Theses and Dissertations (ETDs) as they are digital library of theses.

1.12 Objectives of the Study

1. To summarize OSS phenomena and status of the growth of OSS in various fields with special reference to libraries.
2. To conduct an in-depth study of DSpace Software

3. To understand the existing methods of OSS selection and adoption for digital libraries
4. To apply a framework for measuring maturity of OSS for digital libraries
5. To measure the maturity of DSpace software based on an OSS maturity model.

1.13 Methodology in Brief

This study has used web content analysis for the collection of research data. Web content analysis is the application of traditional content analysis techniques to the web (Herring, 2006, p.235). The data for the research was collected mainly from the official website of DSpace software, sourceforg.net and GitHub code repository supplemented by literature search. The data collected through web content analysis were analysed using multiple methods and tools.

1.14 Scope and Coverage

The study offers a general description on the nature and application of OSS in libraries with particular reference to software for digital libraries. An in-depth study of DSpace software is attempted from the perspectives of OSS maturity to understand both the features of software as well as the software environment. The study covers the DSpace version 1.0.x to 5.x involving a period starting from November 2002 to January 2015 and the entire documentation corresponding to all versions. The six DSpace mailing lists were covered from the beginning to December 2014. The study uses Woods and Guliani's Open Source Maturity Model-2005 as an instrument to measure

the maturity of DSpace which involves fifteen elements of maturity. The outcome of the study can be used by libraries and information centres for the selection and adoption of OSS for digital libraries.

1.15 Limitations of the Study

While there are many OSS for building digital libraries and repositories, the present study is limited to DSpace software. The investigator has selected DSpace software for the study since it is the top among the various OSS being used for digital libraries and repositories the world over. On the methodological part, the research is designed as a case study. The result cannot be generalized to other OSS. On the selection of instrument for measuring maturity, the study is limited to Open Source Software Maturity Model -2005 framed by Woods and Guliani. This Model is designed to measure software as a product and not the software process. This Model is originally designed for information technology departments. The work is not a comparative study of features of various OSS for digital libraries.

1.16 Organization of the Report

1.2 The report of the study is organised under six chapters as follows:

Chapter 1 – Introduction This chapter sketches the emergence of OSS, definition formulated by OSI, adoption of OSS in general, OSS adoption in libraries, Maturity of OSS. It also outlines the significance of the study, operational definitions, objectives, scope and limitations and organization of the study.

Chapter 2 - Review of Related Literature This part contains the related studies done before. They are arranged chronologically under four categories.

Chapter 3 – Methodology This chapter describes the methodology adopted for the study

Chapter 4 – Open Source Software for Digital Libraries – The status of DSpace

This chapter provides the theoretical explanations and foundation of the study.

Chapter 5 – Analysis This chapter offers the analysis of' data obtained through web content analysis from official sources.

Chapter 6 - Summary, Conclusions and Suggestions The last chapter sketches major findings, conclusion, recommendations and suggestions for further research.

The thesis ends with a general bibliography listing books, articles and websites consulted and accessed by the researcher for the preparation of this study. The citation and bibliographic reference have been given according to APA style 6th edition.

1.17 Conclusion

This chapter has given an introduction to the meaning and definition of OSS with special reference to the software development model. The application of OSS in different sectors of social life and the adoption of OSS by governments, organizations and institutions has been described to get an overview of the impact of OSS. The application of OSS in libraries with special reference to OSS for digital libraries was attempted to comprehend the prospects of OSS for librarians in the digital world. The need and significance

of the study is described in the context of selection and adoption of OSS in libraries. The introductory part also contains statement of the problem, definition of key terms, objectives of the study, methodology in brief, scope and coverage and limitation of the study. A study of this kind is justified in the context of research gap in the areas of maturity of OSS for libraries.

References

1. Aichernig, B. K. (2006,) *Qualification of open source software for e-government. e-Maco report 2008, Version 1.0,* Retrieved from http://www.emacao.gov.mo/documents/20/report28.pdf (accessed June 5, 2014).
2. Al Zeheimi, S., Zeki, A. M., Razi, M., Jalaldeen, M., Zain, S. M., & Abubakar, A. (2014). Perceptions of library and information science community towards open source software adoption in libraries of Oman. In *Information and Communication Technology for The Muslim World (ICT4M), 2014 The 5th International Conference on* (pp. 1-6). IEEE. Retrieved from http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7020591 (accessed June 5, 2014).
3. Amollo, B. A. (2013). Feasibility of adaptation of open source ILS for libraries in Kenya: A practical evaluation. *The Electronic Library, 31*(5), 608-634.
4. Anandan, C. (2009). *Product management*. New Delhi: Tata McGraw-Hill Education.
5. Applewhite, A. (2003). Should governments go open source?*., IEEE Software*, *20*(4), 88-91.

6. Barve, S., & Dahibhate, N. B. (2012). Open source software for library services. *DESIDOC Journal of Library & Information Technology*, *32*(5), 401-408.
7. Bisson, C., West, J., & Eby, R. (2007, May/June). *Open source software for libraries*. Library Technology Report. American Library Association Tech Source. Retrieved from http://maisonbisson.com/oss4lib/ (accessed June 5, 2014).
8. Boulanger, A. (2005). Open-source versus proprietary software: Is one more reliable and secure than the other?. *IBM Systems Journal*, *44*(2), 239-248.
9. Breeding, M. (2009). The viability of open source ILS. *Bulletin of the American Society for Information Science and Technology*, *35*(2), 20-25.
10. Bretthauer, D. (2001, December 12). *Open source software: A history*. Retrieved from http://digitalcommons.uconn.edu/cgi/viewcontent.cgi?article=1009&context=libr_pubs (accessed June 6, 2014).
11. Brewer, E., Demmer, M., Du, B., Ho, M., Kam, M., Nedevschi, S., & Fall, K. (2005). The case for technology in developing regions. *Computer*, *38*(6), 25-38.
12. Buchanan, K. & Krasnoff, B. (2005). Can open source software save school libraries time and money? *Knowledge Quest, 33*(3), 32-34.
13. Cabinet Office, UK (2012, April 27). *All about open source: An introduction to open source software for Government IT, version 2.0*. Retrieved from https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/78959/All_About_Open_Source_v2_0.pdf (accessed June 6, 2014).

14. Candela, L., Castelli, D., & Pagano, P. (2011). History, evolution, and impact of digital libraries. In Iglezakis, I., Synodinou, T. & Kapidakis, S. (Eds.). *E-publishing and digital libraries: Legal and organizational issues*, (pp.1-30). Hershey : IGI Global.
15. Chudnov, D. (1999). Open source software: The future of library systems? *Library Journal, 124* (13), 40-43.
16. Classen, H. W. (2007). *A practical guide to software licensing for licensees and licensors: Analyses and model forms* (2nded.). Chicago, IL: American Bar Association.
17. Comino, S. & Manenti, F. M. (2005). Government policies supporting open source software for the mass market. *Review of Industrial Organization*, *26*(2), 217-240.
18. Damsgaard, J. & Karlsbjerg, J. (2010). Seven principles for selecting software packages. *Communications of the ACM*, *53*(8), 63-71.
19. De', Rahul. (2009) *Economic impact of free and open source software - A Study in India*. Retrieved from http://www.iimb.ernet.in/~rahulde/RD_FOSSRep2009.pdf (accessed June 7, 2014).
20. Digital Library Federation (2011, January 11). *A working definition of digital library*. Retrieved from http://old.diglib.org/about/dldefinition.htm (accessed June 5, 2014).
21. DoD (2009, October 16). *Clarifying guidance regarding open source software (OSS).* Retrieved from http://dodcio.defense.gov/Portals/0/Documents/FOSS/2009OSS.pdf (accessed June 7, 2014).
22. Ebert, C. (2008). Open source software in industry. *IEEE Software, 25*(3), 52-53.

23. Elliott, M. S., & Scacchi, W. (2008). Mobilization of software developers: The free software movement. *Information Technology & People*, *21*(1), 4-33.
24. Fageria, N. K. (1992). *Maximizing crop yields*. New York: CRC Press.
25. Fitzgerald, B. (2006). The transformation of open source software. *MIS Quarterly, 30* (3) 587–598.
26. Forge, S. (2006). The rain forest and the rock garden: The economic impacts of open source software. *Info*, *8*(3), 12-31.
27. Ghosh, R.A. (2006, November 20). *Economic impact of open source software on innovation and the competitiveness of the Information and Communication Technologies (ICT) sector in the EU*. Retrieved from http://ec.europa.eu/enterprise/sectors/ict/files/2006-11-20-flossimpact_en.pdf t (accessed June 7, 2014).
28. Golden, B. (2005). *Succeeding with open source*. Boston: Addison-Wesley Professional.
29. Griffin, S. M. (1998). NSF/DARPA/NASA digital libraries initiatives: A program manager's perspective. *D-Lib Magazine, 4* (7/8). Retrieved from http://www.dlib.org/dlib/july98/07griffin.html (accessed June 7, 2014).
30. Gurusamy, K. (2011, February 21). *Open source software adoption in the Australian public sector* (Doctoral dissertation). Retrieved from http://www.canberra.edu.au/researchrepository/file/12add23b-cbf0-eab2-1906-90d50c1017a2/1/full_text.pdf (accessed June 5, 2014).
31. Hasan, N. (2009, October). Issues and challenges in open source software environment with special reference to India. In *Proceedings of International Conference on Academic Libraries (ICAL-2009), University of Delhi, Delhi.* Retrieved from http://crl. du. ac. in/ical09/papers/index_files/ical-43_144_ 317_1_RV.pdf (accessed June 9, 2014).

32. Hendrickson, M., OReilly, T. & Magoulas, R. (2012). *Economic impact of open source on small business: A case study*. Sebastopol, CA: O'Reilly Media.
33. Herring, S. C. (2010). Web content analysis: Expanding the paradigm. In Jeremy Hunsinger, Lisbeth Klastrup and Matthew M. Allen (Eds.), *International handbook of Internet research* (pp. 233-249). Netherlands: Springer.
34. Hoe-LianGoh, D., Chua, A., AnqiKhoo, D., Boon-HuiKhoo, E., Bok-Tong Mak, E., & Wen-Min Ng, M. (2006). A checklist for evaluating open source digital library software. *Online Information Review*, *30*(4), 360-379.
35. Holck, J., Larsen, M.H., Pedersen, M.K. (2005). Managerial and technical barriers to the adoption of open source software. In X. Franch & D. Port (Eds.), *COTS-Based Software Systems: 4th International Conference, ICCBSS 2005, Bilbao, Spain, February 2005* (pp. 289–300). Heidelberg: Springer.
36. Isitan, K. (2011, May). *An approach to establish a software reliability model for different free and open source software development paradigms*. Retrieved from https://tampub.uta.fi/bitstream/handle/10024/82597/gradu05074.pdf?sequence=1 (accessed June 8, 2014).
37. Jones, P., Day, M., & Ball, A. (2009). Institutional repositories should be built on open source software. *Bulletin of the American Society for Information Science and Technology*, *35*(4), 22-26.
38. Khondhu, J., Capiluppi, A., & Stol, K. J. (2013). Is it all lost? A study of inactive open source projects. In Etiel, P., Giancarlo, S., Nabil E.I. &

Alberto, S. (Eds.), *Open source software: Quality verification* (pp. 61-79). Heidelberg: Springer.

39. Krishnamurthy, M. (2008). Open access, open source and digital libraries: a current trend in university libraries around the world. *Program: Electronic Library and Information Systems, 42*(1), 48-55.
40. Krishnamurthy, S. (2003, February). *An analysis of open source business models*. Retrieved from http://faculty.washington.edu/sandeep/d/bazaar.pdf (accessed June 6, 2014).
41. Lundell, B., Lings, B., & Syberfeldt, A. (2011). Practitioner perceptions of open source software in the embedded systems area. *Journal of Systems and Software*, *84*(9), 1540-1549.
42. Lynch, C. A. (2003). Institutional repositories: Essential infrastructure for scholarship in the digital age. *Portal: Libraries and the Academy*, *3*(2), 327-336.
43. Macan, B., Vanesa Fernández, G., & Stojanovski, J. (2013). Open source solutions for libraries: ABCD vs Koha. *Program*, *47*(2), 136-154.
44. Marsan, J., & Paré, G. (2013). Antecedents of open source software adoption in health care organizations: A qualitative survey of experts in Canada. *International Journal of Medical Informatics*, *82*(8), 731-741.
45. Meeker, H. J. (2008). *The open source alternative: Understanding risks and leveraging opportunities*. New Jersey: John Wiley & Sons.
46. MITRE (2011). *Open source software*, Systems engineering guide. Retrieved from: http://www.mitre.org/publications/systems-engineering-guide/enterprise-engineering/engineering-

informationintensive-enterprises/open-source-software (accessed June 8, 2014).

47. Morgan, E. L. (2002). Possibilities for open source software in libraries. *Information Technology and Libraries*, *21*(1), 12.
48. Müller, T. (2011). How to choose a free and open source integrated library system. *OCLC Systems & Services*, *27*(1), 57-78.
49. Mutula, S., & Kalaote, T. (2010). Open source software deployment in the public sector: A review of Botswana and South Africa. *Library Hi Tech*, *28*(1), 63-80.
50. Nagy, D., Yassin, A.M. & Bhattacherjee, A. (2010). Organizational adoption of open source software: Barriers and remedies. *Communications of the ACM, 53*(3), 148-151.
51. Nash, J. C. (2010). Directions for open source software over the next decade. *Futures*, *42*(4), 427-433.
52. NRCFOSS (2013). *Open source maturity model for a department/enterprise/institution*. Retrieved from http://www.nrcfoss.au-kbc.org.in/maturity/ (accessed June 7, 2014).
53. O'Neill, A. (2012), Open source software. In *Ruth Chadwick* (Ed.) Encyclopaedia of applied ethics. (2nd ed. pp.281-287) Retrieved from http://www.sciencedirect.com/science/article/pii/B978012373932200636 (accessed June 5, 2014).
54. Paulk, M C, Curtis M, Chrissis, B and & Weber, C. V. (1993). *Capability maturity model for software, version 1.1. Technical report*. Retrieved from https://resources.sei.cmu.edu/asset_files/TechnicalReport/1993_005_001_16211.pdf (accessed June 7, 2014).
55. Poulter, A. (2010). Open source in libraries: An introduction and overview, *Library Review*, *59*(9), 655-661.

56. Pressman, R.S. (2005). *Software engineering: A practitioner's approach*. (6th Ed.). New York: McGraw-Hill.
57. Ratib, O. & Rosset, A. (2006). Open-source software in medical imaging: Development of OsiriX. *International Journal of Computer Assisted Radiology and Surgery*, *1*(4), 187-196.
58. Raymond, E.S. (2001) *Cathedral and the bazar: Musings on Linux and open source by an accidental revolutionary*. (Rev. Ed.) CA: OReilly Media, Inc.
59. Schneider, K. G. (2009). The thick of the fray: Open source software in libraries in the first decade of this century. *Bulletin of the American Society for Information Science and Technology, 35*(2), 15-19.
60. Seadle, M., & Greifeneder, E. (2007). Defining a digital library. *Library Hi Tech*, *25*(2), 169-173.
61. Sen, R., Singh, S. S., & Borle, S. (2012). Open source software success: Measures and analysis. *Decision Support Systems*, *52*(2), 364-372.
62. Shaikh, M. & Cornford, T. (2011, November). *Total cost of ownership of open source software: A report for the UK Cabinet Office supported by open forum Europe*. Retrieved from http://eprints.lse.ac.uk/39826/1/Total_cost_of_ownership_of_ open_source_software_%28LSERO%29.pdf. (Accessed June 7, 2014).
63. Sharma, A., & Adkins, R. (2005). OSS in India. In Chris DiBona, Mark Stone, Danese Cooper (Eds.), *Open Sources 2.0: The Continuing Evolution* (pp.189-198), CA: OReilly Media.
64. Smith, R. (2002). Adapting a new technology to the academic medical library: Personal digital assistants. *Journal of the Medical Library Association*, *90*(1), 93-94.

65. Sourceforgnet (2014) Retrieved from http://sourceforge.net/ (accessed June 1, 2014).
66. Stahl, M. T. (2005). Open-source software: Not quite endsville. *Drug Discovery Today*, *10*(3), 219-222.
67. Subramaniam, C., Sen, R., & Nelson, M. L. (2009). Determinants of open source software project success: A longitudinal study. *Decision Support Systems*, *46*(2), 576-585.
68. Tennant, R. (2000, January). The role of open source software. *Library Journal*, 125(1), 36. Retrieved from http://search.proquest.com/abicomplete/docview/196834140/fulltextPDF/B6DB42E755B44E0PQ/5?accountid=39226 (accessed June 7, 2014).
69. *The concise Oxford English Dictionary* (2011). New Delhi: Oxford University Press.
70. *The Oxford Dictionary of English* (3rd Ed.) (2010). New Delhi: Oxford University Press.
71. *The New International Webster's Comprehensive Dictionary of the English Language*. (1996) Naples: Trident Press International.
72. Tramboo, S., Shafi, S. M., & Gul, S. (2012). A study on the open source digital library software's: Special reference to DSpace, EPrints and Greenstone. *International Journal of Computer Applications 59(16), 1-9. Retrieved from*

    http://arxiv.org/ftp/arxiv/papers/1212/1212.4935.pdf (accessed June 9, 2014).

73. UNESCO (2010). *Free and open source software*. Retrieved from http://www.unesco.org/new/en/communication-and-information/access-to-knowledge/free-and-open-source-software-foss/browse/5/ (accessed May 2, 2014)

74. Välimäki, M., & Oksanen, V. (2005). The impact of free and open source licensing on operating system software markets. *Telematics and Informatics*, *22*(1), 97-110.
75. Venkatesh, V., & Davis, F. D. (2000). A theoretical extension of the technology acceptance model: Four longitudinal field studies. *Management science, 46*(2), 186-204.
76. Vimal Kumar, V., & Jasimudeen, S. (2012). Adoption and user perceptions of KOHA library management system in India. *Annals of Library and Information Studies*, *59*(4), 223-230.
77. Von Hippel, E. (2001). Learning from open-source software. *MIT Sloan Management Review*, *42*(4), 82-86.
78. Woods, D., & Guliani, G. (2005). *Open source for the enterprise: Managing risks, reaping rewards*. New Delhi: O'Reilly Media.
79. Yeates, S. (2005, November 1). Documentation issues in open source, OSS Watch. Retrieved from http://oss-watch.ac.uk/resources/archived/documentation (accessed June 7, 2014).
80. Yin, R. K. (1994). *Case study research: Design and methods*. (2nd ed.). CA: Sage publications.
81. Young, B. (2001). Forward. In E.S. Raymond, *Cathedral and the bazar: Musings on Linux and open source by an accidental revolutionary*, (Rev.ed.) CA: OReilly Media, Inc.
82. Zhang, D. (2007, August). *Open source software maturity model based on linear regression and bayesian analysis*. (Doctoral dissertation) Retrieved from http://repository.tamu.edu/bitstream/handle/1969.1/ETD-TAMU-1454/ZHANG-DISSERTATION.pdf?sequence=1(accessed June 8, 2014).

Chapter 2 Review of Related Literature

2.1 Introduction

This chapter presents a critical analysis of the related literature to identify and record existing knowledge on the topic of research. The total reviews are brought under four main categories. The first category includes studies on the adoption of DSpace software for the creation of DLs by libraries across the globe. The second category comprises the comparative studies of different OSS for DLs. The third category contains studies that explore various factors influencing OSS adoption in all types of organizations. The studies listed under the fourth category provide information on the measurement of OSS maturity. These reviews are presented as they provide various factors that influence the adoption of OSS by libraries. Moreover, these studies offer

valuable insights to understand whether the maturity of OSS is one of the factors of adoption considered by organizations. The reviews are arranged chronologically under each category

2.2 Adoption of DSpace for DLs

Today's digital environment demands that the development of DLs is no longer an option for libraries but rather an obligation to control the flow of knowledge through various forms of digital materials. A number of articles discussed the adoption of DSpace software for developing various digital systems and the experience of investigators in developing and maintaining these systems in their institutions.

Cervone (2006) illustrated the aspects of system software selection giving particular emphasis to those points where a digital library project differs from a traditional enterprise-level software selection process.

Digital library system software selection differs in some significant ways from traditional software implementations. In particular, security and authentication issues, long-term cost and maintenance considerations, vendor viability, as well as training and documentation are areas where the software selection team needs to devote greater attention if the project is to be successful.

Sutradhar (2006) described the adoption of DSpace for the establishment of an IR at the Indian Institute of Technology (IIT), Kharagpur. The benefits of maintaining an IR were listed that include the archiving of research papers of the members of the institution for online access as well as digital preservation. The resources in the IR include instructional materials, records, data sets, electronic theses, dissertations, annual reports, as well as published papers. The investigator observed that IRs change many of the current practices of

scholarly communication and publishing. The study has also provided the reason behind the adoption of DSpace for the IR. DSpace, being an open source technology platform, can be customised to meet future needs. DSpace was found to be a suitable platform for building an IR as it is OAI-PMH compliant, uses Lucene searching (supporting fuzzy search logic) and it has the "handle" system (for global unique ID of documents).

Barwick (2007) shared the experiences of setting up an IR at Loughborough University, UK. The study outlined the various decision processes involved during the 12-month pilot phase. The selection of the suitable OSS was one of the main tasks. The University selected DSpace above EPrints and Fedora because DSpace had a good web interface and the ability to manage various file formats. The University was looking at developing a "blended" repository and DSpace was found quite suitable. The IR was set up in May 2005. They were able to customize the software. Moreover, they found the structure of DSpace flexible to organize their repository collections according to the University's faculty/departmental structure. The license embedded into the DSpace software, caused some problems for submitters and this was settled by a minor change of the license's wording.

A case study presented by Hulse, Cheverie and Dygert (2007) outlined the process through which the Washington Research Library Consortium selected and implemented the DSpace in a shared information technology environment. The issues confronted in dealing with a multi-institutional implementation were examined through both a detailed description of the implementation and a generalized description of the challenges the consortium faced. They revealed that the collaborative approach presented significant benefits in drawing on the breadth of expertise available among the Consortium and utilizing a shared information technology infrastructure. OSS designed for IRs like EPrints, DSpace and Digital Commons were selected and evaluated by the

consortium. The evaluators found the functionality across these platforms to be generally similar. Initial testing of the DSpace product indicated that it would not be a significant challenge to existing technical support and expertise. As the DSpace platform was designed primarily for a single institution, some customization for a consortium implementation was made. Since DSpace was an out-of-the box platform, the consortium could customize every implementation.

A study on the problems associated with the creation of an IR for a consortium of institutions on one software platform was presented by Joki (2007). DSpace was selected for the PEPIA (Project for Electronic Publications and Institutional Archives) which is a Norwegian government-sponsored effort, to provide institutional repositories to multiple Norwegian universities, university colleges and other research institutions through a consortium. Because of the highly specific requirements and very limited resources (time, money, personnel), a well extensible software platform was required. DSpace was chosen as it was the system which had the most functionality of the ones evaluated. Further, DSpace had a large user group, which would come in handy if help was needed during the development process. Another important requirement was that the system could be hosted on the already existing server platform, preferably without having to acquire new competence in programming languages. The study found that it is possible for multiple organizations to join forces and create a consortium to develop an IR on one software platform.

A study to understand the OSS based DL development in India was conducted by Jose (2007) using online questionnaire. He recorded that DSpace, Eprints, Fedora and Greenstone were the most popular OSS packages used in India.

The result of his study revealed that DSpace was the most popular among the OSS DL solutions in India.

Lam and Chan (2007) documented Hong Kong University of Science and Technology's (HKUST's) experiences in developing its IR. The study highlighted the reasons for adopting DSpace over other OSS packages. The task force decided to focus on OSS that supported OAI-PMH (Open Access Initiatives – Protocol for Metadata Harvesting). Two such IR software programs were evaluated, namely EPrints and DSpace. EPrints was widely used by IR implementers in 2002. DSpace was developed with experience gained from EPrints, but with a clever move from the Perl programming language to Java and Servlet. And at that time, it also had better Unicode support, which was essential to the repository that would contain Chinese materials. With the above consideration, the library decided to adopt DSpace.

Laxminarsaiah, and Rajgoli (2007), described the adoption DSpace for the establishment of an IR for the ISRO HQ library for enabling online access to the various resources on satellite applications, remote sensing, tele-medicine and tele-education and other allied topics. The IR accommodated newspaper clippings, research papers, speeches/lectures, office orders/memorandums, videos, annual reports and the in-house publications. The selection of software for the IR was made by detailed study of OSS like Archimede, CERN (CDSware), DSpace, E-prints and Greenstone. DSpace was found suitable on grounds of technical support and training in using the software. It was also observed that the majority libraries in Bangalore city used DSpace which would facilitate interacting with them for technical support. DSpace was selected as it had tremendous potential and can support numerous forms and formats.

Devakos and Toth-Waddell (2008) in their study described a project to increase access and longevity of electronic government documents in the Ontario Legislative Library. They found that digital repository software, such as DSpace, could be used to extend access to, and longevity of, special collections. DSpace has a number of preservation features including the ability for libraries to set preservation support by file type, checksums to ensure file authenticity, and persistent identifiers. DSpace uses open standards to facilitate interoperability and hence makes it easy to re-use metadata and for search services, such as Google, to crawl content. DSpace is organized by the concept of communities often corresponding to administrative units within an organization.

Reporting the IR development at ICFAI Business School (IBSA), Ahmedabad, Doctor and Ramachandran (2008) observed that developing an IR provided a means for the institution to create archives and make available their wealth of knowledge, increase visibility and prestige through exposure to its digital scholarship. IRs are emerging technologies for capturing intellectual capital, knowledge sharing and management in academic and research institutions especially in developing countries like India. They conducted a survey to identify the various OSS being used for IR in India. It was found that out of 20 IRs, 13 used DSpace. IBSA also chose DSpace for its IR. They realized that the installation of the IR was complex, requiring technical know-how of different software. Creation of communities and collections, archiving of documents into the repository, enriching them with metadata were essential for efficient retrieval of information.

An evaluative study of Indian digital libraries and repositories was done by Mittal and Mahesh (2008) by applying literature review and content analysis. They reported the phenomenal increase in the number of digital libraries and

repositories the world over. They also noted that India was following the global trend and a number of digital libraries and repositories have been developed in the country. The use of OSS for the creation of IR/DL was found to be common. Among the OSS, DSpace was increasingly adopted for the creation of IRs/DLs in India.

DSpace is also adopted for creating ETDs across the globe. Ghosh (2009) in her case study on nine ETD digital libraries in India observed that "ETDs are a new generation of theses and dissertations that can include colour diagrams, colour images, hypertext links, audio, video, animations, spreadsheets, databases, simulations, and virtual reality worlds". She collected data using questionnaire survey, face to face or telephone interviews, and content analysis of ETD web sites and bibliographical databases. Out of 9 ETDs 8 (89%) were using DSpace software. However, the study did not provide information on the process of selection of the software for these ETDs.

Winter and Bowen-Chang (2010) reported the adoption of DSpace for building an IR at the main library of the University of the West Indies, St Augustine, Trinidad and Tobago. They observed that an IR could increase the visibility of an institution's scholarship while paving the way for greater collaboration among researchers outside the institution. The research draws upon the DSpace experience of the University of the West Indies, St Augustine and serves as a model for future projects in the implementation of the DSpace software, particularly in developing countries. One of the main challenges of implementing the IR was choosing the right software that would adequately suit the library's needs and at the same time be cost-effective. Four OSS platforms were evaluated: DSpace, EPrints, Fedora and Greenstone. The evaluation exercise was conducted over a one-month period and included contacting various universities to determine their IR experience and visiting websites of institutions that employed the OSS. The selection process

eventually guided the digitization team to recognize that criteria such as purpose, cost, features and functions, and support were paramount to the internal needs of the University. At the end of the evaluation exercise, the DSpace platform was selected because a community of users existed who could be consulted, it has the ability to accept information in all formats and the software is designed to accommodate long-term preservation.

An IR was built using DSpace software at the Independent University, Bangladesh (IUB), for the systematic storing and access to the research output of the university (Shoeb, 2010). DSpace software was chosen for the IR after reviewing literature on it and observing the comparison of the different OSS packages done by other institutions. In addition, 40 students were given the task of analysing the features of different OSS for IRs. DSpace received highest scores over other software on many aspects. The customization of DSpace was the most challenging task at IUB.

Müller (2011) in his study provided methods to choose a Free and Open Source Integrated Library System (ILS) based on objective criteria. The methodology applied involved three broad steps. The first step consisted of evaluating all the available ILS and keeping only those that qualify as truly open source or freely licensed software. The second step involved evaluating the community behind each open source or free ILS project, according to a set of 40 criteria in order to determine the attractiveness and sustainability of each project. The third step entailed subjecting the remaining ILS to an analysis of almost 800 functions and features to determine which ILS were most suited to the needs of libraries. The final score was used to identify strengths, weaknesses and differentiating or similar features of each ILS. More than 20 open source ILS's were submitted to this methodology but only 3 passed all

the steps: Evergreen, Koha, and PMB. The study used "maturity" as one of the evaluation criteria.

The survey conducted by Sawant (2011) using web based questionnaire revealed the existence of 16 online IRs in India. DSpace was the software used by majority (11) of the IRs. The author tried to understand the preference of IR developers for DSpace. It was reported that DSpace could be easily customised to meet local needs. Moreover, the large communities of DSpace developers were working on improvements and innovation of the software. More workshops were offered on DSpace by various institutions in India to train library professionals. Documentation Research and Training Centre (DRTC) provided a shell script for installation of DSpace to ease the installation process. The results of the study showed that the respondents considered end-user interface to be the top ranking IR-system feature.

Alayon, Nemiz, Superio, de la Peña and Pacino (2012) presented the experiences of adoption of DSpace for developing an IR at the Southeast Asian Fisheries Development Center (SEAFDEC), Philippines. The IR was developed mainly to provide a reliable means for its researchers to store, preserve, share their research outputs, enable easy access to and increase the visibility of its scientific publications. Prior to DSpace, the library developed an in-house digital library using the Greenstone Software in 2009. They were not satisfied with some of the features of Greenstone. They evaluated Eprints and DSpace as these were quite common in Southeast Asia and selected DSpace focusing on its leadership and largest community of users and developers worldwide. Initially, DSpace version 1.7.x was used and later updated to 1.8.x XML Manakin using the Mirage themes.

An overview of IR, developments that are taking place in the Arabian Gulf Region was given by Ahmed and Al-Baridi (2012) in their study. The data

were mainly drawn from various sources on the Internet and by e-mail contact with the authors' acquaintances in several universities and institutions of the Arabian Gulf region. The study covered educational institutions of higher learning and research of the Gulf Cooperation countries only, which included Bahrain, Kuwait, Qatar, Saudi Arabia, Sultanate of Oman and the United Arab Emirates. They reported that open access and IR developments were at the early stages in the Arabian Gulf region. There was an imperative need to spur the developments in these areas in order to derive utmost benefits to both researchers/stakeholders and institutions. DSpace was the most popular OSS for IRs in the region. Out of three IRs in the region, two used DSpace.

Chen, Chen, Hong, Liao and Huang (2012) highlighted the importance of DLs that are focused on creating, organizing, and managing multimedia digital content and collections, and providing search, retrieval, and other information services over computer network. They also put value on OSS for DLs as they were increasingly considered a beneficial alternative to commercial DL software. The increasing budget cuts in libraries were another reason for searching alternatives. Additionally, the costs of maintenance and producing software were very high. Free access and a good level of functionality were the main reasons accounting for the usage and interest in open source DL software. The authors reported the adoption of DSpace 1.4.1 with modifications for a DL that preserved Taiwan library history.

Adewumi, Omoregbe, Misra and Fernandez (2013) conducted a study on three repository software- DSpace, EPrints and Greenstone (DEG). Considering their increasing adoption and usage by universities, the study applied a model that could compare the qualities of repository software. The proposed model was used to measure quality in DEG. The model was validated through real data and the results indicated that DSpace was the better option.

Al Zeheimi, Zeki, Razi, Jalaldeen, Zain, and Abubakar (2014) conducted an in-depth field study to explore the perceptions of library and information science community towards open source software adoption in libraries of Oman and to understand factors affecting OSS adoption at libraries in Oman from library professionals' opinion. The study was undertaken by quantitative survey and case study research method. The study population included participants from seven institutions of Oman who belonged to library and information science field. The study found that lack of awareness on OSS products was the main reason behind the absence of adoption of OSS in Oman.

Khode and Chandel (2015) conducted a survey to assess the status of application of OSS in India. They used various sources to find out the users of OSS in India. The methodology included visiting users' list available on the website of respective OSS, consulting case studies and research papers on OSS, searching on Internet and communication with library professionals through various mailing lists such as NMLIS, LIS-Forum, etc. The study revealed that Koha OSS was the major Integrated Library System (ILS) used by Indian libraries followed by NewGenlib and Open Biblio. The study also discovered that there were 96 open access repositories in India created by using OSS which are accessible in public domain. Out of 96 repositories, 67 were created by using DSpace, 26 by E-print and 3 repositories by Greenstone.

2.3 Comparative studies of OSS packages in libraries

For the purpose of building a digital library of Library and Information Science, Madalli (2003) compared the features such as operating system, web server, language, database, resource identifier, Dublin Core, METS, OAI-PMH, submission and supported file formats of Greenstone, Eprints, DSpace

and Fedora OSS. The comparative analysis found that DSpace was a powerful OSS suitable for the proposed digital library. Its features like allowing submission of digital documents by different members from different locations were found most convenient. However, the software lacked METS standard, the study reported.

Jones (2004) revealed the result of a comparative study of DSpace with ETD-db, software specifically designed for E-theses. The purpose of the study was to identify and select suitable ETD software. The investigator observed that the overall methodology employed by DSpace was superior to that of ETD-db. The study also identified that DSpace was far more functional with regards to essential features such as security and administration which was an important infrastructure for any piece of software of this nature.

Han (2004) revealed the implementation of a Content Management System (CMS) at the University of Arizona Library as a way to manage the electronic contents effectively in terms of preservation, organization and dissemination. The CMS was perceived to be supporting improved information accuracy, increased flexibility, enhanced system management, and reduced maintenance cost for both locally developed documents and external e-contents. The selection of the software for the CMS was an important step. Preservation of the contents, metadata and access were the key areas considered for the success of a digital content management system. Search for a CMS through commercial and open source products were made which resulted in the identification of 17 systems. Out of 17, three candidates (Fedora, Greenstone and DSpace) were selected on the basis of broad criteria such as digital preservation strategies, metadata standards, and access policies. The three software were further evaluated based on four major criteria. Operational; Technical; Scheduled; and Economical. DSpace received the highest marks in

operational analysis, schedule analysis and economic analysis, while Fedora received the highest score in technical analysis. The overall scores showed that DSpace ranked first among these systems. DSpace keeps a file's original name, size and created date. In addition, its built-in data integrity check by using MD5 (a "message digest" algorithm for security applications) to ensure the correctness of each file was noteworthy. More importantly, it defined a migration strategy including introducing the concept of file formats as a hierarchy of "unknown", "known", and "supported". DSpace's consideration for scalable storage allowed the system to use multiple hard drives, which is particularly useful for an IR.

Kumar (2009) evaluated some of the most popular digital library packages. The evaluation was done by using a checklist consisted of 12 categories of items, each with varying degrees of importance: content management, content acquisition, metadata, search, access control and security, report and inquiry, preservation, interoperability, user interface, standard compliance, automatic tools and support. The weights were assigned on the basis of a modified Delphi technique. Three OSS viz. DSpace, Fedora and Greenstone were taken for evaluation. The consolidated score showed that DSpace emerged as a good option having best search and browsing support as well as good support for metadata and provides more power to administrator to put access restrictions at collection level.

A comparative study of DSpace and Eprints was carried out by Karmakar, Das and Thakuria (2010). They identified the following features specific to DSpace; DSpace indexes digital content, so users can search and retrieve results quickly. DSpace distributes digital content over the World Wide Web and also searchable through search engines. DSpace is easy to upgrade. DSpace preserves digital materials over the long term. DSpace has a persistent network identifier for work that never changes or breaks. DSpace has a

number of preservation features including the ability for libraries to set preservation support by file type, checksums to ensure file authenticity, and persistent identifiers. DSpace uses open standards to facilitate interoperability and hence makes it easy to re- use metadata and for search services, such as Google, to crawl content.

Randhawa (2012) attempted a comparative study of Greenstone and DSpace software for building digital libraries. The comparison was based on features like the availability, version, developer prestige, operating system, system requirement, license, language, technical and training support, examples, security, browse and search and other relevant points. The purpose of the study was aimed at supporting professionals to select a suitable OSS for building digital library. The study concluded that both Greenstone and DSpace were appropriate software for creating digital libraries with minor variation in the features and work flows.

Madalli, Barve and Amin (2012) presented an analytical study along with observations regarding digital preservation support available in existing open-source digital library software (OSS-DL) based on test beds created for that purpose. They had set up a test bed environment and installed major OSS-DL. All of the selected software were available under open source license terms and conditions. The OSS-DK included CDS-Invenio, DSpace, EPrints, Fedora, Greenstone, DoKS, and MyCoRe. These software programs were used specially for creating digital archives/digital libraries/institutional repositories. They compared metadata format, persistent identification, audit logs, details of files, actual file storage, checksum and versioning support. The study found that to a large extent Fedora supported more features that were essential from a digital preservation point of view, but it lacked a user-friendly interface; hence, there were not many installations of Fedora. DSpace and EPrints are

now used heavily all over the world to build digital repositories/institutional repositories. To some extent, both of these software programs supported digital preservation. There were large number of repositories available with DSpace. In India, many institutes have taken steps to build digital archives using DSpace.

Masrek and Hakimjavadi (2012) appraised 59 features of three widely utilized open source IR solutions (DSpace, EPrints, Fedora) from the perspective of managing ETDs. For this purpose, all applications were installed and the features were tested in a test-bed environment (a benchmark machine) with a predefined set of ETD collections and registered users. They adopted evaluation criteria set suggested by Gibbons for the study because this criteria set considers the ETD-specific aspects of software solutions. The findings of the study revealed that, although all three solutions are capable of managing ETD systems, in most of the comparative areas that are vital for an ETD repository DSpace was ahead of EPrints and Fedora.

Lihitkar and Lihitkar (2012) compared the features, functions and usability of ten OSS, i.e., Greenstone, DSpace, E-Prints, Fedora, Ganesha, Invenio, XTS, Dienst, VuDL, and NewGenlib. Investigative and evaluative research methodologies were used for the study. Data were collected by surfing Internet and downloading the ten OSS under study. A worksheet was prepared using different criteria for comparative study. It included license, new version, downloaded site, size of the software, bundle of associated software, operating system, language support, facilities such as searching and browsing, multimedia, metadata etc. The study concluded that the compared OSS packages were flexible, and can be customized and modified at many different levels–including the programming level. The authors recommended Greenstone and DSpace for building digital libraries.

Tramboo, Shafi, and Gul (2012) attempted a comparative study of DSpace, Greenstone and Eprints. The study compared licence cost, product type, update cost, resource identifier, OAI PMH, supported item types, metadata formats, user interface functions, thumbnail preview, searching capabilities, browsing options, syndication, user authentication, statistical reporting, software platforms, databases, programming languages, web server, associated software, machine to machine interoperability, licence and services. The study did not propose any specific DL systems. This study can be used as a reference guide by any organization or institute to decide which one will be ideal for creating and showcasing their digital collection.

Choi (2014) evaluated the application profiles and development characteristics of library open source projects. The author evaluated 594 library OSS projects from Sourceforge and Foss4lib with a number of criteria like development status, license type, sponsorship etc. The study found that while various types of library OSS applications were found to be under development and in use, there has been a steady decrease in the number of projects initiated since 2009. Although sponsorship was significantly positively associated with several indicators of OSS project success, the proportion of sponsored projects was relatively small compared to the proportions reported in some other contexts. In total, 71 per cent of the projects have a restrictive license scheme, suggesting that the OSS ideology is valued among library OSS projects. The results also indicated that library OSS projects exhibit several characteristics that differ from the traditional developer-oriented OSS projects in terms of their technical environment.

2.4 OSS adoption factors in various organizations

Mtsweni and Biermann (2008) studied the implementation of open source software within the South African government. The study reviewed that OSS were increasingly becoming an alternative for proprietary software particularly in the government sector globally. The adoption and implementation of OSS by the government sector were cited as one of the enablers for the adoption of OSS by the private sector. It is also apparent that in the government sector internationally, OSS is seen as a viable technology for reasons such as lowering software costs, growing local software development industry, and bridging the digital divide.

Yuan (2009) in his dissertation tried to investigate the factors leading to the adoption of OSS by Singaporean companies. The study highlighted that the adoption of OSS was driven by the perception of a cost advantage. The organizations that adopted OSS agreed cost as their biggest concerns and top priorities. The next significant findingwas the need for increased OSS skills. The organizations that succeeded in adopting OSS possessed pre-existing skills in OSS use. This was instrumental to better mitigate risks and to lower their training costs. The final principal finding was that OSS appeared to be used mainly in systems infrastructure applications.

Mutula and Kalaote (2010) investigated OSS adoption in the public sector in Botswana and South Africa. The study explored different aspects of OSS deployment in the public sector. The study found that even though IT managers in government of Botswana had positive attitude towards OSS, there was limited use of OSS. Compared to Botswana South African Government provided support for harnessing OSS.

Gurusamy (2011) in her research on "Open source software adoption in the Australian Public Sector" investigated various factors that may enable or inhibit OSS adoption in public sector organizations. The study used two major technology adoption theories: Diffusion of Innovation (DOI) theory and the Technology Acceptance Model (TAM). The study observed that maturity of OSS products was an enabler for OSS adoption as organizations tended to adopt mature OSS products. The respondents agreed that most of the open source software they use was very mature and well documented. Characteristics of the software were identified as an enabler and were represented by the ability to add new features, prompt fixing of software bugs, availability of source code, and product maturity.

Midha and Palvia (2012) examined the factors that lead to OSS success longitudinally over a period of time with two measures of project success: project popularity and developer activity. They examined 283 OSS projects over a span of 3 years to understand the impact of various factors, categorized as intrinsic and extrinsic factors, on OSS project success over the first three years of its life. A longitudinal analysis of these factors was conducted at various stages in the OSS life cycle to reach unique insights into various project management decisions. The study had the following hypotheses that formed the factors of OSS success.

1. OSS projects with higher technical success are more popular
2. The cumulative existing developer base of the previous versions of an OSS project is positively associated with its current version's market success.
3. The cumulative existing developer base of the previous versions of an OSS project is positively associated with its current version's technical success

4. OSS projects that use a non-restrictive license exhibit higher market success than those that use a restrictive license.
5. OSS projects using a non-restrictive license exhibit lower technical success than those using a restrictive license.
6. The cumulative existing user base of the previous versions of an OSS project is positively associated with its current version's market success.
7. The number of language translations of an OSS project is positively related to its market success.
8. Complexity of the OSS project is negatively related to its technical success.
9. OSS projects that delegate responsibility exhibit higher technical success.
10. Modularity of the OSS project is positively related to its technical success.

The study was concluded with the testing of hypothesis. The hypothesis 1 was not supported while 2 and 3 received mixed results. The rest of the hypotheses were supported.

Rossi, Russo and Succi (2012) investigated the importance of factors for the adoption of free/libre open source software (FLOSS) in the public sector. Based on the methodological approach on two exploratory case studies with contrasting result logic, they built a multi-level framework grounded both on literature review, and feedback from stakeholders. The study considered phases of adoption (initiation, implementation) and the levels of adoption (technological, organizational, environmental and individual).The study found the importance of a strong and decision-centric management board to give the impulse for the initiation phase of the process. As perceived by the stakeholders, a strong governmental support is of paramount importance to increase the adoption at the public level, although in the case studies examined

the initiation stage started from the impulse of a championing management. Both case studies passed the initiation phase successfully. Continuous employees' training, organizational objectives consensus, and business process reengineering were found important for the implementation phase.

Spinellis and Giannikas (2012) listed the research around benefits and significant factors driving OSS adoption, and concluded that the most important reason of choosing open source was purchasing cost and the total cost of ownership. Although other benefits like stability and performance, flexibility, and control, external support, and security were also stressed in the advantages listed by open source adopters, it seemed that total cost of ownership and lower acquisition cost were the most significant ones. On the other hand, there were also many factors that operated as barriers toward the organizational adoption of OSS.

Li, Tan and Yang (2013) analysed whether human capital, that is, knowledge, skills, experience, abilities, and capacities possessed by employees, played a vital role in the adoption of open source software (OSS) by organizations. The study conducted by a survey among 104 OSS-adopting organizations and 111 non-adopting organizations in China. The result supported the argument that OSS-adopting organizations could be clearly distinguished from their non-adopting counterparts in terms of their availability of internal OSS human capital, accessibility to external OSS human capital, organizational size, IT department size, and criticality of IT operation. Theoretical and practical implications are discussed in the study.

Marsan and Pare (2013) conducted a qualitative survey to find antecedents of OSS adoption in health care organisations in Canada. They conducted 18 semi structured interviews with IT experts from all levels of the Province of Quebec's health and social services sector in Canada. The study found that

eight factors associated with three distinct theoretical perspectives influenced OSS adoption.

2.5 Measuring the maturity of open source software

Clark (1997) in his study on "the effects of software process maturity on software development effort" observed that controlling and improving the processes used to develop software was a primary remedy to the problems of time lag, over budget, non-conforming to requirements and of poor quality. The Software Engineering Institute at Carnegie Mellon University had published the Software Capability Maturity Model (SW-CMM) for use as a set of criteria to evaluate an organization's process maturity. The model is also used as a roadmap to improve a software development process's maturity. The premise of the SW-CMM is that mature development processes deliver products on time, within budget, within requirements, and of high quality.

Mockus, Fielding and Herbsleb (2002) in their research on "Two case studies of open source software development: Apache and Mozilla" addressed key questions about the development process of Apache and Mozilla. They observed that OSS development had the capacity to compete successfully, and perhaps in many cases displace, traditional commercial development methods. In order to begin investigating such claims, they examined data from two major OSS projects, the Apache web server and the Mozilla browser. By using email archives of source code change history and problem reports, they quantified aspects of developer participation, core team size, code ownership, productivity, defect density, and problem resolution intervals for these OSS projects.

Zhang (2007) in his research on "Open source software maturity model based on linear regression and Bayesian analysis'' introduced an OSS maturity model that facilitated the software assessment and helped users to make a decision in choosing an OSS from a large pool of OSS candidates in the same category. Though a few maturity models had been proposed in the past, the parameters in the model were assigned not based on experimental data but on human experiences, feelings and judgments. These models were subjective and can provide only limited guidance for the users at the best. The study has proposed a quantitative and objective model which was built from the statistical perspective.

Spiro (2009) in her report on "Archival management software" explored ten archival management systems belonging to commercial and OSS domain. Archival management systems are a kind of software that typically provide integrated support for the archival workflow, including appraisal, accessioning, description, arrangement, publication of finding aids, collection management, and preservation. The study brought out the maturity status of six commercial software and four OSS. The commercial software includes Adlib Archive, Calm for Archives, Cuadra Star, Eloquent Archives, Minisis M2A and Past Perfect. The OSS includes Archivist' Toolkit, Archon, ICA-AToM and Collective Access. The study collected data from interviews with users as well as on previous studies of archival software and information provided by the developers and vendors. The study offered features matrices including software maturity for selected archival management systems so that archivists can make quick comparisons of different software. The maturity of software included the year of software release, number of installations, release of different versions of the software, and the smooth progression of the software over the years.

Raza, Capretz and Ahmed (2012) presented a usability maturity model specifically aimed at usability-related issues for open source projects. The model examined the degree of coordination between open source projects and their usability aspects. The measuring instrument of the model contained factors selected from four of their empirical studies, which examined the perspectives of OSS users, developers, contributors and the industry. The model was questionnaire based and incorporates five maturity levels and eleven usability factors. The study was a first of its kind and contributed a methodology to evaluate the usability maturity of OSS.

Akbari and Peikar (2014) explored the trend of OSS development in the field of Geospatial Information Systems (GIS) and the maturity of Web Geospatial Information Systems (WebGIS). This paper applied Open Source Maturity Model (OSMM) to the most significant WebGIS software in GIS field to measure their maturity. The study found that OSMM was a mechanism for evaluating open source products to help professionals choosing the most suitable one. The study selected MapGuide OS, UMN MapServer and PostGIS for evaluation. Under longevity assessment the study examined life span, version number and the total number of downloads. Under product team assessment it examined size of the project team and the number of commits from the top ten contributors for the last year. Under support option assessment, community support and paid support were examined. The study observed developer creator documentation, web postings and commercially published documents under documentation assessment and operating system, web server, standards, database integration under product integration assessment. The training option assessment sought the availability of web based mini tutorials, developer created tutorials and commercial tutorials. The final maturity scores were calculated on the basis of points obtained for the six elements of maturity- software, support, documentation, training, integration

and professional services. UMN Mapserver got the highest score (83.5) followed by PostGIS (78.5%) and by MapGuideOS (56%). The results indicated that UMN MapServer is fully mature open source software compatible with other similar commercial products.

2.6 Conclusion

The investigator has reviewed 48 studies under four categories. The studies that describe the adoption of DSpace for DLs were attempted to understand the factors influencing the choice of DSpace. Since this is a case study of DSpace, more previous studies would strengthen the understanding of the issue. The factors that influenced the selection of DSpace include adherence to standards, use of Lucene search engine, handle system, the community structure of DSpace, matching to exiting support and expertise, unicode support, tremendous potential for growth, having forums, community of users, cost, facility for customization, more workshops on the area, leadership and largest community. The studies have touched some aspects of maturity.

The studies that attempted to compare the features of different OSS for digital libraries provide us insights on the way libraries selected particular software from among several OSS. The features compared include operating system, web server, language, database, resource identifier, Dublin Core, METS, OAI-PMH, submission and supported file formats, security and administration, preservation, metadata and access, migration strategy, version, developer prestige, system requirement, license, language, technical and training support, user interface functions, thumbnail preview searching capabilities, browsing options, syndication, user authentication, statistical reporting, software platforms, databases, associated software, machine to machine

interoperability. Though maturity of OSS was not checked, studies compared some aspects that come under the subject of maturity of software.

The studies that highlighted the organisational factors of OSS adoption include government support, cost advantage, pre-existing skills in OSS use, positive attitude towards OSS, ability to add new features, prompt fixing of software bugs, availability of source code, and product maturity.

Studies that focus on measuring the maturity of OSS offer insights on Capability Maturity Model, organization's process maturity, development process of Apache and Mozilla projects, source code change history and problem reports, developer participation, core team size, code ownership, productivity, defect density, and problem resolution intervals for these OSS projects, year of software release, number of installations, release of different versions of the software, and the smooth progression of the software over the years. One study attempted to measure the maturity of WebGIS software.

The review of literature shows that libraries and organisations adopt various criteria for the selection of OSS. Though several factors considered by libraries come under maturity, there is lack of studies on the exploration of measuring maturity of OSS for libraries. This area is left unattended by researchers and the present study tries to fill this research gap.

References

1. Adewumi, A., Omoregbe, N., Misra, S., & Fernandez, L. (2013,). *Quantitative quality model for evaluating open source web applications: Case study of repository software.* In IEEE 16th International Conference on Computational Science and Engineering (CSE), 2013(pp. 1207-1213). Retrieved from http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6755361 (accessed March 5, 2014).
2. Ahmed, S. S., & Al-Baridi, S. (2012). An overview of institutional repository developments in the Arabian Gulf Region. *OCLC Systems & Services*, *28*(2), 79-89.
3. Akbari, M., & Peikar, S. R. H. (2014, September). Evaluation of free/open source software using OSMM model case study: WEBGIS and spatial database. *Advances in Computer Science: an International Journal3*(5), 34-43 Retrieved from http://www.acsij.org/documents/v3i5/ACSIJ-2014-3-5-521.pdf (Accessed October 5, 2014).
4. Alayon, S. B., Nemiz, E. S., Superio, D. L., de la Peña, J. G., & Pacino, L. G. (2013). The development of an institutional repository at the Aquaculture Department of the Southeast Asian Fisheries Development Center, Philippines. *Program: electronic library and information systems*, *47*(4), 384-398.

5. Al Zeheimi, S., Zeki, A. M., Razi, M., Jalaldeen, M., Zain, S. M., & Abubakar, A. (2014, November). *Perceptions of library and information science community towards open source software adoption in libraries of Oman*. In *Information and Communication Technology for The Muslim World (ICT4M), 2014 The 5th International Conference on* (pp. 1-6). IEEE. Retrieved from http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7020591 (accessed June 5, 2014).
6. Barwick, J. (2007). Building an institutional repository at Loughborough University: some experiences. *Program: Electronic Library and Information Systems*, *41*(2), 113-123.
7. Cervone, H. F. (2006). Some considerations when selecting digital library software. *OCLC Systems & Services: International digital library perspectives*, *22*(2), 107-110.
8. Chen, C. M., Chen, Y. T., Hong, C. M., Liao, C. W., & Huang, C. M. (2012). Developing a Taiwan library history digital library with reader knowledge archiving and sharing mechanisms based on the DSpace platform. *The Electronic Library,30*(3), 426-442.
9. Choi, N. (2014). The application profiles and development characteristics of library Open Source Software projects. *Library Hi Tech*, *32*(2), 260-275.
10. Clark, B. K. (1997). The effects of software process maturity on software development effort.
11. Devakos, R., & Toth-Waddell, A. (2008). Ontario government documents repository D-space pilot project. *OCLC Systems & Services*, *24*(1), 40-47.
12. Doctor, G., & Ramachandran, S. (2008). DSpace@ IBSA: knowledge sharing in a management institute. *VINE*, *38*(1), 42-52.
13. Ghosh, M. (2009). E-theses and Indian academia: A case study of nine

ETD digital libraries and formulation of policies for a national service. *The International Information & Library Review*, *41*(1), 21-33.

14. Gurusamy, K. (2011). *Open Source Software Adoption in the Australian Public Sector* (Doctoral dissertation). Retrieved from http://www.canberra.edu.au/researchrepository/file/12add23b-cbf0-eab2-1906-90d50c1017a2/1/full_text.pdf (accessed June 8, 2012).
15. Han, Y. (2004). Digital content management: the search for a content management system. *Library Hi Tech*, *22*(4), 355-365.
16. Hulse, B., Cheverie, J. F., & Dygert, C. T. (2007). ALADIN research commons: a Consortial institutional repository. *OCLC Systems &Services*, *23*(2), 158-169.
17. Joki, S. M. E. (2007). PEPIA: a Norwegian collaborative effort for institutional repositories. *OCLC Systems & Services*, *23*(2), 204-209.
18. Jones, R. D. (2004). DSpace vs. ETD-db: Choosing software to manage electronic theses and dissertations. Retrieved from: http://www.ariadne.ac.uk/issue38/jones (accessed March 2, 2012).
19. Jose, S. (2007, February 14). *Adoption of open source digital library software packages: A survey*. Retrieved fromhttp://eprints.rclis.org/8976/1/Sanjojose.pdf (accessed March 12, 2012).
20. Karmakar, G. S., Das, R., &Thakuria, J. (2010, February 18). *Institutional repositories in India: a comparative study of Dspace and Eprints*. Retrieved fromhttp://ir.inflibnet.ac.in/bitstream/1944/983/1/43.pdf (accessed July 5, 2012).
21. Khode, S., & Chandel, S. S. (2015). Adoption of open source software in India. *DESIDOC Journal of Library & Information*

Technology, *35*(1) p. 30-40.

22. Kumar, V. (2009). *Comparative evaluation of open source digital library packages.* Retrieved from https://drtc.isibang.ac.in/bitstream/handle/1849/441/comparative_evaluation_DL_vinit.pdf?sequence =1(accessed May 6, 2010).
23. Lam, K. T., & Chan, D. L. (2007). Building an institutional repository: Sharing experiences at the HKUST Library. *OCLC systems & services*, *23*(3), 310-323.
24. Laxminarsaiah, A., & Rajgoli, I. U. (2007). Building institutional repository: An overview. *OCLC Systems & Services*, *23*(3), 278-286.
25. Li, Y., Tan, C. H., & Yang, X. (2013). It is all about what we have: A discriminant analysis of organizations' decision to adopt open source software. *Decision Support Systems*, *56*, 56-62.
26. Lihitkar, S. R., & Lihitkar, R. S. (2012). Open source software for developing digital library: Comparative study. *DESIDOC Journal of Library & Information Technology*, *32*(5), 393-400.
27. Madalli, D. P. (2003). A digital library of library and information science using DSpace, paper presented at the DRTC Workshop on the Semantic Web, March 23-25, Bangalore Retrieved from https://drtc.isibang.ac.in/handle/1849/56 (accessed November 2, 2004).
28. Madalli, D. P., Barve, S. and Amin, S. (2012). Digital Preservation in Open-Source Digital Library Software. *The Journal of Academic Librarianship* 38(3), P. 161–164.
29. Marsan, J., & Paré, G. (2013). Antecedents of open source software adoption in health care organizations: A qualitative survey of experts in Canada. *International Journal of Medical Informatics*, *82*(8), 731-741.
30. Masrek, M. N., & Hakimjavadi, H. (2012). Evaluation of three open

source software in terms of managing repositories of electronic theses and dissertations: A comparison study. *Journal of Basic and Applied Scientific Research*, *2*(11), 10843-10852.

31. Midha, V., & Palvia, P. (2012). Factors affecting the success of open source software. *Journal of Systems and Software*, *85*(4), 895-905.
32. Mittal, R., & Mahesh, G. (2008). Digital libraries and repositories in India: an evaluative study. *Program: Electronic Library and Information Systems*, 42(3), 286-302.
33. Mockus, A., Fielding, R. T., & Herbsleb, J. D. (2002). Two case studies of open source software development: Apache and Mozilla. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, *11*(3), 309-346.
34. Mutula, S., & Kalaote, T. (2010). Open source software deployment in the public sector: a review of Botswana and South Africa. *Library Hi tech*, *28*(1), 63-80.
35. Mtsweni, J., & Biermann, E. (2008, October). An investigation into the implementation of open source software within the SA government: An emerging expansion model. In *Proceedings of the 2008 annual research conference of the South African Institute of Computer Scientists and Information Technologists on IT research in developing countries: riding the wave of technology* (pp. 148-158). ACM. Retrieved from
36. Müller, T. (2011). How to choose a free and open source integrated library system. *OCLC Systems & Services*, *27*(1), 57-78.
37. Randhawa, S. (2012). Open source software for creation of digital library: A comparative study of Greenstone digital library software &DSpace. *Indian Journal of Library and Information Science, 6*(3), 42-52.

38. Raza, A., Capretz, L. F., & Ahmed, F. (2012). An open source usability maturity model (OS-UMM). *Computers in Human Behavior*, *28*(4), 1109-1121.
39. Rossi, B., Russo, B., & Succi, G. (2012). Adoption of free/libre open source software in public organizations: factors of impact. *Information Technology & People*, *25*(2), 156-187.
40. Sawant, S. (2011). IR system and features: study of Indian scenario. *Library Hi Tech*, *29*(1), 161-172.
41. Shoeb, M. Z. H. (2010). Developing an institutional repository at a private university in Bangladesh. *OCLC Systems & Services*, *26*(3), 198-213.
42. Spinellis, D., & Giannikas, V. (2012). Organizational adoption of open source software. *Journal of Systems and Software*, *85*(3), 666-682.
43. Spiro, L. (2009, January). *Archival Management Software*. Retrieved fromhttp://www.clir.org/pubs/reports/spiro/spiro_Jan13.pdf (accessed March 5, 2012).
44. Sutradhar, B. (2006). Design and development of an institutional repository at the Indian Institute of Technology Kharagpur. *Program: Electronic Library and Information Systems*, *40*(3), 244-255.
45. Tramboo, S., Shafi, S. M., & Gul, S. (2012). *A study on the open source digital library software's: Special reference to DSpace, EPrints and Greenstone*. Retrieved from http://arxiv.org/ftp/arxiv/papers/1212/1212.4935.pdf (accessed March 12, 2013).
46. Winter, M., & Bowen-Chang, P. (2010). Dealing with DSpace: The experience at the University of the West Indies, St Augustine. *New Library World*, *111*(7/8), 320-332.
47. Yuan, E.K.K. (2009, January). *The adoption of open source software by Singaporean companies*. Retrieved from

http://eprints.qut.edu.au/20496/1/Edmund_Koh_Ker_Yuan_Thesis.pdf (accessed May 8, 2010).

48. Zhang, D. (2007, August). *Open source software maturity model based on linear regression and Bayesian analysis* (Doctoral dissertation). Retrieved from http://repository.tamu.edu/bitstream/handle/1969.1/ETD-TAMU-1454/ZHANG-DISSERTATION.pdf?sequence=1 (accessed March 9, 2011)

Chapter 3 Methodology

3.1 Introduction

The previous chapters have given an introduction to the topic of research and the related work done so far on this area. It was found that the problem of research taken by the investigator is an area where little studies exist. Hence it is quite significant to attempt a study on the maturity of Open Source Software (OSS) being used in libraries. This chapter provides information on the methodological basis of the study. Methodology forms the key part of any research and it gives scientific background to the chosen research. Research is best conceived as the process of arriving at dependable solutions to problems through the planned and systematic collection, analysis and interpretation of data (Mouly, 1978, p.12). Merriam-Webster online dictionary defined research

as “studious inquiry or examination ;especially investigation or experimentation aimed at the discovery and interpretation of facts, revision of accepted theories or laws in the light of new facts, or practical applications of such new or revised theories or laws”. The methodology applied to achieve this task varies from one environment to another. The methods of research may also different to various disciplines and topics.

The present study is designed to examine the status of maturity of an OSS used in libraries in order to propose the process of maturity as a new tool to the selection and adoption of OSS in libraries. To address the problem, the investigator has chosen a maturity model from among existing ones and an OSS for DLs from among many OSS designed for it. This chapter contains a description on the research basis, choosing case study as a method for research, justification for the selection of a particular maturity model for the study, choice for DSpace as a unit for case study and the methods followed for the collection and analysis of data.

3.2 Research Quest

The inferences obtained from the previous studies show that OSS is widely deployed in all fields of human life and that brings cost savings along with quality in technology. The OSS adoption in organizations depends on various factors. The maturity of OSS products is one of the important factors that influence OSS adoption. Maturity denotes the longevity of the product. Since institutions like libraries adopt software for long use, the future prospects of the product are highly important. Currently, OSS are the significant part of libraries and the use of OSS for digital libraries is prominent across the globe. The adoption of OSS in libraries is an important area of study and research. The survey of related studies revealed that the selection and adoption of OSS

in libraries are made on the basis of comparative studies of features of different software. The aspect of maturity of the product is left unattended. This study is oriented towards filling this research gap. Measuring maturity of an OSS helps to understand the critical issues on the growth, development and sustainability of a product. It brings more product knowledge and becomes an additional tool to library professionals in the selection of an appropriate OSS for the long term use in libraries. The theoretical understanding derived out of this study shall be a significant addition to the existing knowledge base of library and information science.

3.3 Justification for Case Study

Basically, a case study is an empirical enquiry of small study group, one individual case or one particular population. However, a "case" need not be concerned only with people, but an activity, program, or event that is bounded by place and time (Creswell, 1998). The present work is designed as a case study of DSpace. While there are many OSS being used for digital libraries (DLs), the investigator has deliberately selected DSpace since case studies help to narrow down a very broad field of research into one easily researchable topic giving more chance to focus on a specific case. The principle objective of case study is deep understanding (Woodside, 2010). The real business of case study is particularization, not generalization and case studies try to know a case well and not primarily as to how it is different from other (Stake, 1995). The case study method may provide some very important information that could not have been revealed any other way (Lawal, 2009, p.78). Hence the current research is planned as a case study to provide deep understanding on DSpace OSS from maturity perspectives.

3.4 Justification for the Selection of a Maturity Measurement Model

The identification of an instrument for research is very important for several reasons. There are three qualities of a good research instrument: validity, reliability and usability (Calmorin & Calmorin, 2007, p.51). Hence the identification of a valid, reliable and usable maturity measurement model was a real task for the current study. Previous studies have proved that traditional software maturity models cannot be directly applied to OSS as the method of design and development of OSS is different (Zhang, 2007). A search on databases of scholarly literature, online catalogues, Google books and Google Scholar bring information on literature on several models for OSS maturity. Search was also conducted for the evaluation of various maturity models. The study by Stol and Babar (2010) provided a comparison framework for open source software evaluation methods (p.389). After conducting a rigorous and systematic investigation for identifying all available OSS maturity models, they could list twenty models. The investigator's identification of various OSS maturity models is based on this study. Table 3.1 lists chronologically the twenty OSS maturity models identified by Stol and Babar.

Table 3.1 OSS Maturity Models

Sr.No	**Model Name**	Year
1	Capgemini Open Source Maturity Model	2003
2	Evaluation Framework for Open Source Software	2004
3	A model for Comparative Assessment of OS Products	2004
4	Navica Open Source Maturity Model	2004
5	Woods and Guliani's OSMM	2005
6	Open Business Readiness Rating (Open BRR)	2005

7	Atos Origin Method for Qualification and Selection of OSS	2006
8	Evaluation Criteria for Free/OSS Products	2006
9	A Quality Model for OSS Selection	2007
10	Selection Process of Open Source Software	2007
11	Observatory for Innovation and Technological Transfer on OSS	2007
12	Framework for OS Critical System Evaluation	2007
13	Balanced Scorecards for OSS	2007
14	Open Business Quality Rating	2007
15	Evaluating OSS through Prototyping	2007
16	A Comprehensive Approach for Assessing Open	2008
17	Software Quality Observatory for OSS-2008	2008
18	An operational approach for selecting open source components in a software development project	2008
19	QualiPSo Trustworthiness model -2008	2008
20	Open Source Maturity Model-2009	2009

The models listed in Table 3.1 were examined to understand the elements of maturity and the way these are being examined. It was found that all models have elements that seem related. The elements of maturity include; description of the software, support, documentation, training, product integration and professional services, functionality, quality, performance, support, community size and security. Among the twenty OSS maturity models shown above, the researcher has selected Woods and Guliani's OSMM-2005 for the current research based on the proposed qualities of flexibility, understandability, implementability, correctness and relevance (Salah, Paige & Cairns, 2014, p.319).

It is noted that none of the twenty models shown above are specifically designed to measure the maturity of OSS for libraries. However, the elements

of maturity described in the Woods and Guliani's OSMM-2005 are lengthy, descriptive and best match the factors of OSS in the field of library and information science. Moreover, the Woods and Guliani's OSMM-2005 is available in a published book at an affordable price across the globe and it has been widely referred in textbooks, websites, primary journals and online encyclopaedias. The model provides fifteen elements of maturity and under each element there is scope for deriving many sub elements which will add more study components for the researcher. Moreover the sub elements support for the collection of quantitative and qualitative data specific to research which will reduce the degree of subjectivity. The details of the model are discussed under the heading 3.6.

3.5 Justification for the Choice for DSpace as a Unit for Case Study

The two directories of open access repositories- Open DOAR and ROAR- list around 3045 digital repositories in the world. These repositories use various software packages. ROAR lists 32 software packages while Open DOAR list 155 software being used by these repositories. However, data from both repositories show that majority of open digital repositories are built using OSS. The prominent OSS include DSpace, EPrints, Fedora and OPUS. According to the official website of DSpace, as of December 2014, there are 1802 installations for DSpace across 117 countries representing six continents of the world. The present study has chosen DSpace as a case for research as it is the most widely adopted digital library software in the world. Previous data shows that the number of installations of DSpace at global level increases every year and it is expected that there would be a growing user community for the software in the future (Cherukodan, Santhosh Kumar & Humayoon

Kabir, 2013). The knowledge derived out of this study shall help libraries on decision making regarding the adoption of the software.

3.6 Woods and Guliani's Open Source Maturity Model-2005

There are two types of metrics used for software development: product metrics and process metrics. Product metrics are used to quantify characteristics of the product being developed. Process metrics are used to quantify characteristics of the process being used to develop the software (Jalote, 2008, p.12). The current study focuses on product maturity based on Woods and Guliani's Open Source Maturity Model-2005 (hereafter WGOSMM). The WGOSMM is providing a formal methodology to assess the maturity of an open source product.

3.6.1 The Elements of Maturity

The core of the WGOSMM is the fifteen elements of maturity that address the indispensable features of a mature software. The following are the elements of open source maturity;

- Leadership and culture
- Vitality of community
- Quality of end-user support
- Extent and scope of documentation
- Quality of packaging
- Momentum
- Quality of code and design
- Quality of architecture
- Testing practices

- Integration with other products
- Support for standards
- Quality of project site
- License type
- Potential for commercial conflicts
- Corporate commitment

The fifteen elements of maturity are the functional specifications of requirements of a mature OSS. A close examination of the elements of maturity reveals that they are the combination of software features and software environment. The leadership and culture, vitality of community, quality of end-user support, momentum, quality of project site, potential for commercial conflicts and corporate commitment are the factors belonging to software environment. The extent and scope of documentation, quality of packaging, quality of code and design, quality of architecture, testing practices, integration with other products, support for standards and license type are the features of software. These functional specifications are matched against a particular product. The WGOSMM provides detailed explanations for each element and directs the users of the model to examine several sub-elements. The fifteen major elements and the sub-elements coming under them form the research questions in this study. These research questions are answered through the collection of valid data from authoritative sources.

3.6.2 Maturity Score and Score Chart

The last part of the model attempts to quantify the maturity of an OSS in product criteria, use criteria and integration criteria. Product criteria are specific about the product itself. The age of the software is a product criterion. Use criteria are specific about what it takes to use the product that include

installation, configuration, support channels etc. Integration criteria are specific about the ability of the software to work in the adopted environment. For each criterion the model assigns a score of 1, 2, or 3. These scores and their descriptions are shown in the Table 3.2.

Table 3.2 WGOSMM Scoring Description

Score	Status	Description
1	Immature product	The product is lacking in several critical areas. It is not fit for adoption for a production
2	Reasonably mature	The product has sufficient quantity of features with a bright future. But it is weak on some areas.
3	Very mature	The product has a long and stable history, a broad and vibrant user community.

Table 3.2 shows the three possible status of an OSS. The criteria for scoring are given in Table 3.3, 3.4 and 3.5 under product criteria, use criteria and integration criteria respectively. The WGOSMM reduces the fifteen elements of maturity to twelve criteria under the product, use and integration criteria.

Table 3.3 WG-OSMM Score Chart for Product Criteria

Woods and Guliani's Open Source Maturity Model-2005				
Maturity Criteria	Score=1	Score=2	Score=3	Criteria Description
Product Criteria				
Age	6 months	6-months -2 years	2 years	OSS that are just getting underway are risky
Multiple Supported Platforms	One Platform	Many related platforms	Multiple heterogeneous platforms	Products that work on both Windows and Unix are more desirable.
Momentum	No release in last 6 months	Two releases in past year	Regular releases	This is key to helping separate vital products from ones that are withering.
Popularity	Unknown product	Viable alternative	Category leader	Popular OSS products are well tested and therefore more mature.
Design quality	Monolithic application	Multiple components	Well-defined API	This is key in determining the effort required to extend and adapt the product for use.

Table 3.3 shows five product criteria that include Age of the OSS, the ability of the software being installed on multiple operating systems, the frequency of release, popularity of the software and its design quality. An OSS that has completed two years is safe to adopt for a production environment

Table 3.4 WG-OSMM Score Chart for Use Criteria

Woods and Guliani's Open Source Maturity Model-2005				
Maturity Criteria	Score=1	Score=2	Score=3	Criteria Description
Use Criteria				
Setup cost	Poorly documented install process; poor documentation; help available from developers	Well-documented install process; reasonable documentation; help available from developers; help available in support forums	Well-documented install process; install wizards/scripts available; reasonable documentation; help available from developers; help available in support forums; third-party install services	Most products should require a setup effort of hours or days, not weeks or months.
Usage cost	Poor or nonexistent documentation; help available only through direct contact with developers	User manuals available; help available in support forums	Third-party training services available	This criterion is often overlooked when evaluating a product.
End-user support	No forums or mailing lists	Some forums or mailing lists	Well-run forums and mailing lists, with archives and search; third-party support options	User community (forums, mailing lists) and third-party support are vital to a product's success.

The Table 3.4 shows three Use criteria for a mature OSS. The set up cost, usage cost and end user support correspond to the documentation part as well as the community support of OSS.

Table 3.5 WG-OSMM Score Chart for Integration Criteria

Woods and Guliani's Open Source Maturity Model-2005			
Maturity Criteria	Score=1	Score=2	Score=3
Integration Criteria			
Modularity	Monolithic structure; possible but hard to extend	Multiple modules; possible to extend	Multiple modules, well-defined API; possible and easy to extend
Collaboration with other products	Unknown	Known cases of integration	Lots of integration documented
Standards compliance	Unknown or proprietary	Outdated	Current industry standards
Developer support	No forums or mailing lists	Some forums or mailing lists	Well-run forums and mailing lists with archives and search; third-party support options

Table 3.5 shows the four integration criteria to recognize a mature OSS. Modularity, collaboration with other products and standard compliance correspond to the ability of the software to integrate with other products. The developer support denotes the availability of support for the installation and maintenance of the software through live mailing list, archives of mailing list and third party intervention.

3.7 Method of Data Collection and Presentation of Data

The present work is designed as a case study. Case study research enables the researcher to draw upon many approaches to data collection (Swanson &

Holton, 2005, p.340). In a case study the researcher has to collect data in a variety of settings, under a variety of conditions for which the researcher has no control over (Lawal, 2009, p.78). The primary and secondary data for the study were collected through a combination of approaches. The major part of data for the research was collected using web content analysis. Content analysis is a systematic technique for coding symbolic content (text, images etc.) found in communication and semantic themes (Bauer, 2000). According to Berelson (1954) content analysis is a research technique for the objective, systematic, and quantitative description of the manifest content of communication. (p.489). Since the introduction of the first browser in 1993, the system of interlinked, hypertext documents known as the Web has grown to be the primary multimodal content delivery system on the Internet and today it is one of the largest content delivery vehicles in the history of the world that provide seemingly endless opportunities for research (Herring, 2006, p.233). The Web is a complex and rich mixture of old and new technologies. It provides many opportunities and challenges for researchers who apply content analysis to Web content (Kim & Kuljis, 2010). Web content analysis is the application of traditional content analysis techniques to the web (Herring, 2010, p235). One can use the Internet as a primary data collection resource in a number of ways for quantitative and qualitative data (Rudestam & Newton, 2014, p.287).

The Table 3.6 shows the list of websites from which data were obtained for the study.

Table 3.6 Websites used for Data Collection

Sr. No	URL	Description	Period Accessed
1	http://www.dspace.org/	Official websites of DSpace software	During 2010-2014
2	https://github.com/	World's largest Software code repository	December, 2014
3	http://sourceforge.net/	World's largest repository of OSS projects	During 2013-2014
4	http://www.duraspace.org/	Official websites of DuraSpace Organization	During 2010-2014
5	http://dspacedirect.org/	A hosted DSpace service	December, 2014
6	http://roar.eprints.org/	Registry of Open Access Repositories	During 2010-2014
7	http://www.opendoar.org/	Directory of Open Access Repositories	During 2010-2014
8	https://www.mail-archive.com	Mailing list archive	December, 2014
9	https://www.linkedin.com/	Professional network	January, 2014
10	https://scholar.google.co.in/	Scholarly works by authors	January, 2014
11	http://web.archive.org	Archives of websites	December, 2014

The Table 3.6 shows the major websites from where the investigator collected data. The major part of data was collected from the official websites of DSpace, SourceForg.net and GitHub repository. The biographical details of DSpace team leaders were collected from institutions' websites to which they were part and from the website of LinkedIn, a social network. The back issues of DSpace newsletter published during 2000-2002 (Four issues) and available with

www.mail.archive.com were examined to understand the historical milestones of DSpace project. Google Scholar was used to identify the scholarly contributions of DSpace team members and citations they received for their articles.

The Table 3.6 also shows the descriptions of the website and date of accessing data. Though these websites were visited many times by the investigator during the course of research, the final date is set as 2nd February 2015 because on this date the investigator revisited all web pages and ensured the existence of data on the site. The determination of the accuracy and consistency of data was easy as there are several alternate sources to verify it. The exploration of tacit knowledge was made with the help of email with team members of DSpace developer community.

The data collected were presented in narrative analysis with the help of figures, charts and tables of frequencies and percentages. Simple statistical techniques were employed as and when required for analysis and interpretations of data. Data were derived from DSpace mailing list archive by using a computer programming.

References

1. Bauer, M (2000), Classical content analysis: a review. In M.W. Bauer & G.Gaskell (Eds.), *Qualitative researching with text, image and sound: a practical handbook.* London: Sage. (pp.131-155).
2. Berelson, B. (1952). *Content analysis in communication research*. New York: Free Press.
3. Calmorin, L. P., & Calmorin, M. A. (2007). *Research methods and thesis writing*. Manila: Rex Book Store.
4. Cherukodan, S., Santhosh Kumar, G., & Humayoon Kabir, S. (2013). Using open source software for digital libraries: A case study of CUSAT. *The Electronic Library*, *31*(2), 217-225.
5. Creswell, J.W. (1998). *Qualitative inquiry and research design: Choosing among five traditions*. Thousand Oaks, CA: Sage.
6. Herring, S. C. (2010). Web content analysis: Expanding the paradigm. In Jeremy Hunsinger, Lisbeth Klastrup, Matthew M. Allen (Eds.) *International handbook of internet research* (pp. 233-249). Netherlands: Springer.
7. Jalote, P. (2008). *An integrated approach to software engineering*. (Ed.3.), New Delhi: Narosa.
8. Kim, I., & Kuljis, J. (2010, June). Applying content analysis to web based content.*Proceedings of the ITI 2010 32nd International conference on information technology interfaces, June 21-24, 2010, Cavtat,*

*Croatia.*Retrieved from http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=5546426

9. Lawal, I. (2009). *Library and information science research in the 21st century: A guide for practicing librarians and students*. Oxford: Chandos.
10. Mouly, G. J. (1963). *Educational research: The art and science of investigation*. Boston: Allyn and Bacon, Boston
11. Research. (n.d.). In *Merriam-Webster's online dictionary*. Retrieved from http://www.merriam-webster.com/dictionary/research
12. Rudestam, K. E., & Newton, R. R. (2014). *Surviving your dissertation: A comprehensive guide to content and process*. California: Sage Publications.
13. Salah, D., Paige, R., & Cairns, P. (2014). An Evaluation Template for Expert Review of Maturity Models. In Andreas, J., Pasi, K., Marco, K., Tomi, M. J. M. & Mikko, R. (Eds.) *Product-Focused Software Process Improvement* (pp. 318-321). Switzerland: Springer International Publishing.
14. Stake, R. E., & Savolainen, R. (1995). *The art of case study research*. Thousand Oaks, CA: Sage publications.
15. Stol, K. J., & Babar, M. A. (2010). A comparison framework for open source software evaluation methods. In Pär J. Ågerfalk, Cornelia Boldyreff, Jesús M. González-Barahona, Gregory R. Madey, John Noll(Eds.)*Open source software: New horizons* (pp. 389-394). Springer Berlin Heidelberg.
16. Swanson, R. A., & Holton, E. F. (2005). *Research in organizations: Foundations and methods in inquiry*, Oakland, CA: Berrett-Koehler.
17. Woodside, A. G. (2010). *Case study research: theory, methods and practice*. Bingley: Emerald.

18. Zhang, D. (2007, August). *Open source software maturity model based on linear regression and Bayesian analysis*. (Doctoral dissertation) Retrieved from http://repository.tamu.edu/bitstream/handle/1969.1/ETD-TAMU-1454/ZHANG-DISSERTATION.pdf?sequence=1 (accessed March 9, 2011).

.

Chapter 4 Open Source Software for Digital Libraries: The Status of DSpace

4.1 Introduction

The previous chapters of this study have given an introduction to the topic of research, related literature on the area and the methodological basis of research. This chapter tries to provide a theoretical base to the topic of research. Hence, this chapter is designed to have a better understanding of the application of OSS in libraries with special reference to DLs. The study covers various OSS for DLs with an in-depth examination of DSpace software.

OSS continues to make inroads into various fields of human activity from simple to mission critical applications challenging the traditional methods of software adoption and use. The factors behind the wide adoption and use of OSS in various organizations have been subjected to academic study and research (Ellis & Van Belle, 2009; Fitzgerald, 2011; Mijinyawa, 2008; Rossi, Russo & Succi, 2012). The factors of OSS adoption were understood in technological, organizational and environmental contexts. Generally they include the absence of license fee, free access to source code, availability of support options, perceptions of quality, facility of trial and test, potentials of customization, prospects of innovativeness, impact of government policies, and availability of IT infrastructure, commitment from organization and motivation from staff. However, while offering plenty of benefits, OSS is not free from risks and challenges (Ayala, Cruzes, Hauge & Conradi, 2011). Moreover, the possibility of success and sustainability is not equal to all OSS projects. Hence research is going on the field to understand the strong and stable OSS candidates.

4.2 History of OSS Adoption in Libraries

The factors that influenced the adoption of OSS in various organizations are also applicable to libraries. Daniel Chudnov's article published in 1999 on OSS and the future of libraries is regarded as the primary and seminal work on the subject. Chudnov (1999) observed that the community based development of OSS has a striking similarity to the economics of libraries and OSS can be used for organizing, communicating, preserving and giving access to information. He advocated the use of GNU/Linux operating system; Apache web server and MySQL database in libraries and asked librarians to help create OSS so that they would not be behind the curve of OSS movement. Initially the growth of OSS was very slow in libraries. Librarians at Yale's Cushing/Whitney medical library used Apache, MySQL and the PHP3 web scripting language for managing information about online resources and selected subject based Internet resources. Chudnov mentioned the use of Prospero, a web-based document delivery system created in 1999 by the staff of the Prior Health Sciences library at The Ohio State University. Breeding (2002) reported the OCLC's release of the source code of Java to its SiteSearch toolkit, code to Pairs search engine, and several other tools under OSS licenses. The George Mason University's OSCR (Open Source Course Reserves), an electronic course reserves system designed for academic libraries was another example. However, in those days library proponents like Breeding could not visualize a victory of OSS over commercial products in the integrated library system (ILS) arena like Linux and Apache made in operating system and web server arenas (Breeding, 2002). But in 2009, Breeding reported the market acceptance for three OSS ILS- Koha, OPALS and Evergreen- and predicted a fertile ground for OSS in the library community (Breeding, 2009). Currently there are more than two hundred OSS products being used in libraries. The foss4lib.org website has published 219

OSS specific to libraries that are applied around twenty two different areas of library services. The Figure 4.1 shows the eight major categories of OSS and the number of projects under it.

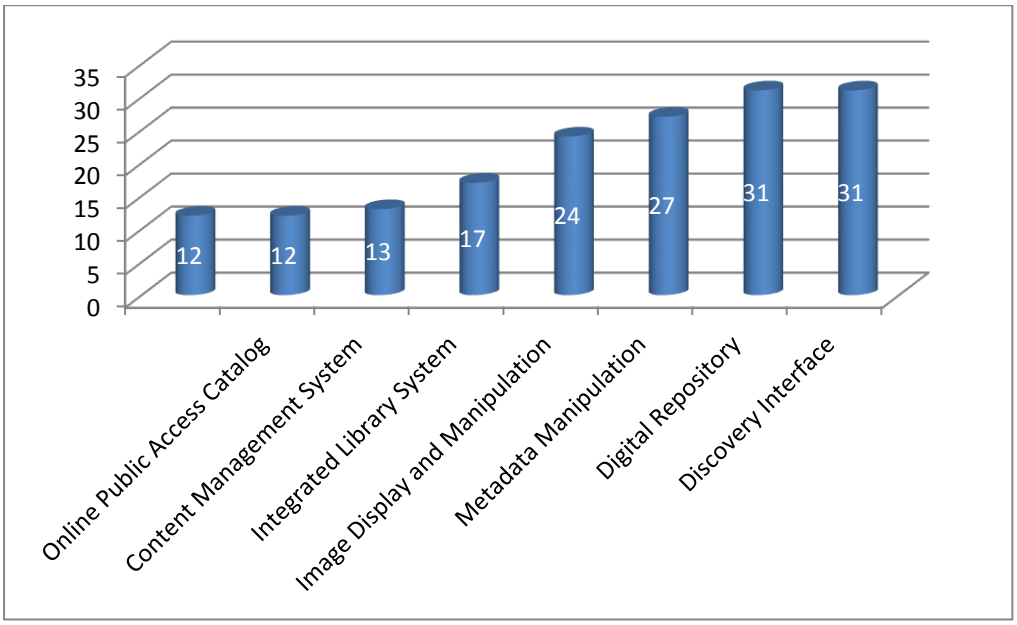

Figure 4.1 Major Category of Library OSS and Number of Projects

The rest of the categories are listed below with the number of projects under them in bracket. Archival Record Manager and Editor (7), Bibliography (7), Compilation (1), Content Management Framework (4), Course Management (1) Data Preservation and Management (5), Electronic Reserves (2) Interlibrary Loan (4), Journal Publishing (1), Knowledge Base (7), Link Resolver (4), Preservation Repository (2), Reference (5) and Subject Page Curation (2). This list helps us to understand the variety and extent of OSS projects in libraries. The study has further examined OSS for ILS and digital library as they are very important part of library service.

4.3 OSS for Integrated Library Systems (ILS)

The main type of software in use in libraries is ILS as every library needs an ILS. It is the most important and crucial tool that affect every aspect of the library service from providing a search interface to the library catalogue and to automating library tasks. ILS is also known as Library Management Systems (LMS). It is an automated system consisting of a number of functional modules, such as acquisitions, circulation, cataloguing, serials, an OPAC (Online Public Access Catalogue) and so on (Li, 2008). ILS makes library functions easier. The cost of proprietary ILS packages is so high that many libraries especially in developing countries cannot afford to purchase and maintain them. Cost has been reported as one of the main factors of adoption of OSS ILS (Singh, 2013).

Hence the availability of ILS in the OSS platform is a great boon to libraries. Apart from cost savings, OSS packages maintain quality under community inspection and involvement. The major ILS in the OSS arena include: ABCD, BiblioteQ, E-Library, Emilda, Evergreen, Gnuteca, Koha, NewGenLib, Next-LEnju, OPALS, Open Library Environment (OLE), OpenBiblio, OtomiGenX, PHP My Library, PHP MySQL, PMB, Senayan Library Management System (SLiMS) and Web Librarian. However, all OSS are not popular and widely adopted. In the United States and Canada, three open source ILS products dominate – Koha, OPALS and Evergreen (Breeding, 2009). And recent studies identify Koha and Evergreen as the most popular ILSs in the United States (Singh, 2013). While Evergreen and OPALS have not yet found wide adoption outside the United States and Canada, Koha finds use in libraries worldwide (Breeding, 2009). Apart from cost considerations, there are many other factors that influence the adoption of OSS ILS. This includes the support and involvement of library associations in imparting training and education

through workshops, the support of government, voluntary organizations, funding agencies and individual professionals. The trialability of OSS is an important feature for librarians in learning the working of the software and testing it on machines before being implemented. The availability of live CDs is another factor that accelerated the application of ILS in libraries.

4.4 The Concept of Digital Libraries

The practice of building and maintaining digital libraries (DLs) is an important part of modern librarianship. The positive trend towards DLs can be understood by going through the strategic plan documents of major libraries across the globe. This trend can be perceived as the response of library professionals towards adoption of OSS, provision of open access, the value of digitization and digital preservation. DLs offer a new role for library professionals as almost every type of information can be represented in digital form now. In a digital world the support for collection, access and preservation of digital content, tools and services are highly important while maintaining the traditional functions of libraries. Kahn and Wilensky (1995) define DLs as infrastructure that is open in its architecture and which supports a large and extensible class of distributed digital information services. The Digital Library Federation (1998) defined DLs as "organizations that provide the resources, including the specialized staff, to select, structure, offer intellectual access to interpret, distribute, preserve the integrity of, and ensure the persistence over time of collections of digital works so that they readily and economically available for use by a defined community or set of communities". DLs are global, multilingual repositories of data, knowledge, sound, and images inviting people everywhere to become users and learners (Griffin, 1998). DLs are extensions and augmentations of physical libraries (Marchionini & Fox,

1999). The fundamental characteristics of libraries – systematic access to information resources, the ideas represented by those resources, and sets of human stakeholders –also extend to DLs (Pomerantz, & Marchionini, 2007). The benefits of establishing a DL includes the provision of easy access to scholarly material generated by research community, long-term archiving of information and research output thereby preserving it for the future, the possibility of information to be widely and quickly disseminated so that it achieves the highest impact and increases the academic reputation of an institution (Pappalardo & Fitzgerald, 2007).

Digital objects are the basic building blocks of a digital library (Arms, 1995). The various items in a digital library are called digital objects. They include text, pictures, musical works, computer programs, databases, models and designs, video programs, and compound works combining many types of information. Digital objects are created in two ways: digitization and digital creation. Digitization is the process of creating digital equivalent of analogue materials by using devices such as scanner. Digital creation denotes the production of digital materials by using computers or digital cameras or video recorders. These types of digital materials are called born digital. DLs contain both born digital and digitized materials of diverse forms. Information stored in a digital object is called "content", which is divided into "data" and information about the data, known as "properties" or "metadata" (Arms, 1995).

DLs are known in different terms. The use of the words like institutional repositories (IRs), digital repositories, digital archives is prominent among scholars. The term IRs is relevant when a system captures, organize, disseminate and preserve the digital materials created by an institution like a university. When these repositories provide open access they are called open digital repositories. The digital collections of dissertations and theses were grouped under Electronic Theses and Dissertations (ETDs). However, all these

terms correspond to the concept of DLs. Smith (2008) shares the idea that IR is a digital library of local digital content including grey literature. Hence, as already mentioned, the investigator uses the word DL as an umbrella term to include and mean all entities that satisfy the requirements of technology and infrastructure in order to collect, organize, retrieve and preserve digital objects of many kinds.

4.5 OSS for DLs

DLs are complex systems involving hardware, software, developers and users. Digital libraries follow various architectures. The underlying architecture of a digital library has to support two main needs. It must provide methods for grouping digital library objects and must provide means for retrieval (Arms, 1995). Among other essential requirements, software is an important component of any DL. The process of selection and adoption of a software package among the many ones is also an important phase in the design of a DL. A group of software applications was created in the 1990s and early 2000s to facilitate the creation of DLs. In 1999s there was intense activity in the evolution of libraries; involving librarians and computer scientists in new collaborations (Helly, 1999). DLs were recognized as topic of computer science research the world over which led to the development of many OSS for DLs mainly by universities. By 2004, there were around ten OSS for creating DLs. The trend towards OSS for DLs can be understood from the "Guide to Institutional Repository Software" published by the Open Society Institute (OSI), New York, in 2004. The guide contained nine OSS for creating DLs. The Table 4.1 shows the name and developers of OSS listed by OSI.

Table 4.1 OSS for DLs in the year 2004

Sr.No	Software	Developer
1	Archimede	Laval University Library, Canada
2	ARNO	University of Amsterdam, Tilburg University, and the University of Twente
3	CDSware	European Organization for Nuclear Research
4	DSpace	MIT Libraries and Hewlett-Packard Lab
5	Eprints	University of Southampton
6	Fedora	University of Virginia and Cornell University
7	i-Tor	Innovative Technology-Applied- Netherlands
8	MyCoRe	University of Essen
9	OPUS	University of Stuttgart

Table 4.1 displays nine OSS for DLs and majority (77%) of them are contributions from Universities in the world. The OSI has not included Greenstone OSS in the list. The number of OSS based DLs has increased subsequently. The Directory of Open Access Repositories (Open DOAR) and the Registry of Open Access Repositories (ROAR) list DLs in the world that provide open access. As per ROAR, there are 32 software packages available for DLs. Further examination of the category of the software packages revealed that the majority of software listed by ROAR belongs to OSS. Out of 32 software 23 (71%) are OSS for DLs. The Open DOAR lists 155 software for DLs. However the determination of the category of software was difficult due to the non-availability of data. As of December 2014, the website of foss4lib.org lists 31 OSS for DLs. The Table 4.2 shows the alphabetical list of DL software, URLs and the name of developers.

Table 4.2 OSS for DLs in the year 2014

Sl.no	Name of the software	URL	Developers
1	Archimede	http://www.bibl.ulaval.ca/archimede/index.en.html	Lavel University Library, Canada
2	Archivematica	https://ww.archivematica.org/en/	Artefactual Systems Inc.
3	Atrium Digital Exhibits	https://jira.duraspace.org/browse/HYGALL/?selectedTab=com.atlassian.jira.jira-projects-plugin:summary-panel	DuraSpace
4	BitCurator	http://www.bitcurator.net/	School of Information and Library Science at the University of North Carolina and the Maryland Institute for Technology in the Humanities.
5	CollectionSpace	http://www.collectionspace.org/	LYRASIS
6	CONTENTdm Integration	https://www.drupal.org/project/contentdm	Dries Buytaert
7	Curate	https://github.com/ndlib/curate	on an alpha release.
8	Digitization Metadata Editor	http://dme.sourceforge.net	Not available

9	DMP Online	http://www.dcc.ac.uk/	Digital Curation Centre
10	DSpace	http://www.dspace.org/	MIT and HP Lab
11	DSpace-CRIS	http://cineca.github.io/dspace-cris/index.html	Cineca
12	EPrints	http://www.eprints.org/uk/	University of Southampton
13	eXtensible Text Framework	http://xtf.cdlib.org/	California Digital Library
14	Fedora	http://fedora-commons.org/	Cornell University
15	Goobi	https://www.goobi.org/en/	Saxon State and University Library Dresden
16	Greenstone	http://www.greenstone.org/	New Zealand Digital Library Project at the University of Waikato
17	Hydra	http://projecthydra.org/	Stanford University, University of Virginia & University of Hull
18	Invenio	http://invenio-software.org/	CERN (European Council for Nuclear Research)
19	IR+ (IR PLUS)	http://code.google.com/p/irplus/	University of Rochester

20	Islandora	http://islandora.ca/	University of Prince Edward Island's Robertson Library.
21	Kora	http://kora.matrix.msu.edu/promo_index.php	Matrix, the Center for Digital Humanities and Social Sciences at Michigan State University
22	Kramerius	http://krameriusdemo.mzk.cz/search/ (Demo site)	Czech libraries
23	MyCoRe	http://www.mycore.de/	University of Essen
24	Omeka	http://omeka.org/	Roy Rosenzweig Center for History and New Media, George Mason University
25	Open Harvester Systems	http://pkp.sfu.ca/ohs/	Public Knowledge Project
26	Open Library	https://openlibrary.org/	Internet Archive
27	ResCarta Toolkit	http://www.rescarta.org/	The ResCarta Foundation, Inc.
28	SobekCM	http://sobekrepository.org/	George A. Smathers Library at the University of Florida
29	Variations	http://variations.sourceforge.net/	Indiana University Digital

			Library Program.
30	VuDL	http://vudl.org/	Villanova University's Falvey Memorial Library
31	Weko	http://weko.at.nii.ac.jp/	National Institute of Informatics, Japan

The Table 4.2 shows that the number of OSS for DLs has increased from nine in 2004 to thirty one in 2014. These OSS vary with respect to their technologies, operating system compatibility, use of database, documentation, frequency of release, community strength, support options, ability to integrate with other software and standards etc. Since variations of these parameters influence the adoption of OSS in libraries, there is need to harness studies on these aspects.

4.6 The Prominent DL Software

The identification of the major software for DLs is having research value as they help the process of software comparison, selection and adoption. The data for which can be obtained from multiple sources. The Open DOAR and ROAR list open access DLs in the world along with the information on software being used. The list is limited to the DLs that are available on Internet. As of February 2015, Open DOAR lists 2867 DLs in the world distributed among 155 software packages that include both proprietary and OSS. On this date ROAR lists 3924 DLs among 32 software packages. Since these sources vary in numbering the DLs, the investigator collected data from one more source: http://maps.repository66.org/ prepared by Stuart Lewis who is a DSpace committer, developer and trainer. Lewis's list provides a

combined view of ROAR and OpenDOAR. Table 4.3 shows the number of DLs and the distribution of DLs among various OSS as per the three sources mentioned above. An average value has been assigned to get a combined view of three sources.

Table 4.3 Number of DLs and Share of DL Software

Sources	**No of DLs**	**DSpace**	**EPrints**	**Fedora**	**OPUS**	**Green-stone**	**OSS %**
OpenDOAR	2867	1230	390	38	71	55	60
ROAR	3924	1520	554	46	70	22	55
maps.repository66	3045	1225	468	40	26	-	57
Average Value	**3278**	**1325**	**470**	**41**	**55**	**38**	**58**

Table 4.3 shows that there are around 3278 DLs in the world that provide open access and visible over the Internet. These open DLs use both proprietary software and OSS. From the data obtained, it can be seen that majority (58%) of open DLs are built using OSS. DSpace, EPrints, Fedora, OPUS and Greenstone are the major OSS being used for open DLs in the world. And among the major OSS, DSpace occupies top position. This study further explores DSpace software.

4.7 DSpace Software- Historical Overview

Literature on DSpace is abundant in the world as it is the top among the software being used for DLs. Understanding the historical overview is important to know the mission and leaders behind it. DSpace was developed

by the Massachusetts Institute of Technology (MIT) Libraries and Hewlett-Packard Labs (HP) in 2002. The role of Ann Wolpert, the late, former Director of MIT libraries in building DSpace was remarkable. According to Wolpert, in the age of the Internet, a great research library could serve not only as a window into scholarly output for given members of university and research communities, but also as a window for the world at large into the scholarly enterprise. The idea for a system to address this issue was conceived in the late 1990 and it was materialized in 2000, when she achieved the MIT Libraries collaboration with HP to build DSpace as a multidisciplinary repository attached to an institution for the digital research and educational material produced by members of that institution. When the majority of born digital materials are never published by traditional means, such a repository becomes a vital part of library functions for the access and preservation of scholarly resources. Computer scientists recognized the reality of "bit rot", the process of data degradation or data decay, or the disappearance of large volume of information in born digital form within very short period of time. Tansley (2003) observed that digital data is vulnerable to rapidly changing standards for file formats, software applications and potential damage to the drums, disks, tapes and other media used to actually store the data. The researchers at HP investigated ways to tackle the problem and recognized that librarians are the most valuable allies than IT specialists in such a venture.

4.8 The Release of DSpace version 1.0

DSpace version 1.0 was officially released on November 4, 2002, under the terms of the BSD open source license. The DSpace team included MacKenzie Smith, Mary Barton, Margret Branschofsky, Julie Harford Walker, Greg McClellan, Mick Bass, Peter Breton, Peter Carmichael, William Cattey, Eric

Celeste, Dan Chudnov, Joyce Ng, David Stuve and Robert Tansley. The team members were a combination of library professionals and computer scientists from MIT libraries and HP labs except Peter Breton and Peter Carmichael who were developers at White Rabbit Software and PC Consulting respectively. DSpace was available for download from the sourceforge.net website.

MIT Libraries was the first to adopt DSpace followed by the Center for Technology, Policy & Industrial Development, Department of Ocean Engineering, Laboratory for Information and Decision Systems, and Sloan School of Management. DSpace offered service mainly to three categories of users. For end users, items deposited in DSpace can be accessed remotely via Internet. Contributors have the advantages of digital distribution and long-term preservation for their digital contents of text, audio, video, images, datasets and more. The institutions who adopt DSpace get a system to provide access to all the research of the institution through one interface. DSpace was designed to store digital objects that include articles, preprints, working papers, technical reports, conference papers, books, theses, data sets, computer programs, visual simulations and models. The software supports different format for these digital objects.

4.9 DSpace-Technical Features

The DSpace version 1.0 was designed to be installed on HP/UX, Linux, Mac OSX and Solaris platforms. The prerequisite software included Java 1.3- the Standard SDK, Java Libraries (JavaBeans, Java Servlet 2.3 and JSP 1.2, JavaMail API), Apache and Tomcat 4.0, PostgresSQL relational database, Lucene search engine, Jena (an RDF toolkit from HP Labs), OAICat from

OCLC, and several other useful software libraries. All leveraged components were also belonging to OSS.

DSpace platform was designed as a three layer architecture considering the development in future. It contained a storage layer at the bottom, business layer in the middle and service layer on the top. Figure 4.2 shows the architecture of DSpace based on the version 1.0.

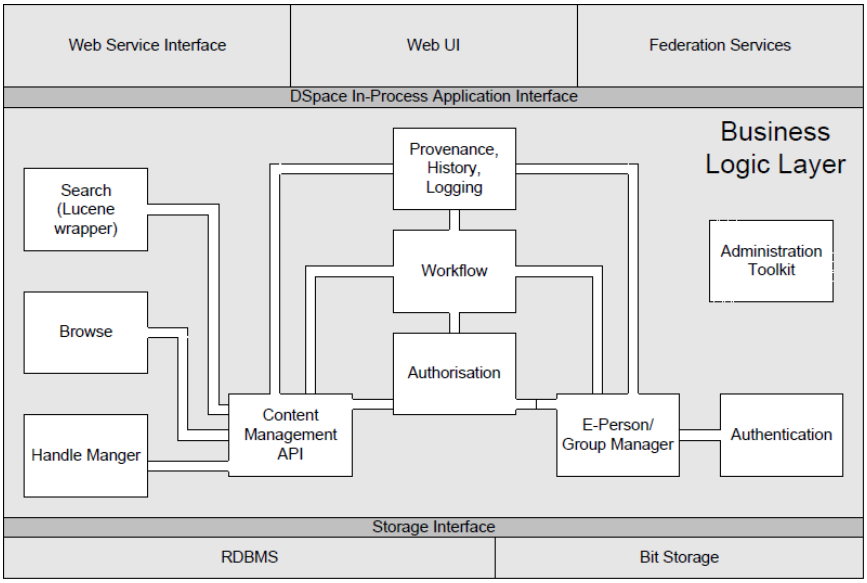

Figure 4.2 DSpace Architecture (Bass et al., 2002)

Figure 4.2 shows the three layers of functions. At the bottom, the storage layer preserves the contents and metadata. It is achieved through the service of PostgresSQL database or Oracle. The business logic layer manages the contents, e-persons (content submitters), authorization, and workflow. The application layer communicates with the users of the system that include web user interface, systems for interoperability and federation services. This architecture is the core of DSpace throughout all versions. The subsequent version of DSpace integrated several services upon user requirements.

4.10 DSpace Workflow

The basic structure of a DL built using DSpace reflects the structure of a university or educational institution. A university comprises of several departments for various faculties. There shall be several sub- components under a department like labs, library etc. The organization of digital collections in a university like environment should address the institutional structure to facilitate a systematic inflow of materials. DSpace satisfies this need by forming a community structure that corresponds to the departments of a university. The top level communities of DSpace look like the following;

Department of Arabic

Department of Biotechnology

Department of Botany

Department of Chemistry

Department of English

Department of History

Department of Library and information Science

Department of Malayalam

Department of Philosophy

Department of Psychology

DSpace further goes to create Sub-Communities under a community. The concept of Sub-Communities denotes the units under a community. It may correspond to a department library, department research lab, faculty members etc. from where digital collections come to the DL.

DSpace organizes various digital objects under a Collection. A collection is the final structural unit. A collection can be built either under a Community or under a Sub-Community. The concept of Community, Sub-Community and Collection is described below in Figure 4.3 based on investigator's practical knowledge in creating collection in Cusat Digital Library (dspace.cusat.ac.in).

Figure 4.3 Community Structure of DSpace

Figure 4.3 shows the formation of Community, Sub-Community and Collection structure in a DSpace system. Here, School of Engineering is a Community. The Division of Civil Engineering, Computer Engineering and Safety and Fire Engineering are Sub-Communities. The project reports under Civil Engineering, mini project reports, project reports and seminar reports under Division of Computer Engineering and the PhD Theses under Division of Safety and Fire Engineering are collections. Collection is the place where the digital files (items) are stored. Creation of a collection is essential to put digital objects. Collections may have varying policies and workflows determining authorisation for depositing digital objects and the stages of contents being archived. DSpace allots each Collection its own entry page that displays information, news and links reflecting the contents stored in the Collection.

4.11 Collection Building in DSpace

The process of collection building in a DSpace system is achieved through the Administrator and E-people. Administrator is one who has the powers to create, delete, edit or modify Communities, Sub Communities, Collections or Items. Heads of departments, project leaders, system administrators and librarians can be an Administrator. Those who are authorized by the Administrator to upload items in a DSpace system are known as an "E-Persons". Faculty members, library staff and project members may act as E-persons. The permission to add collection by an e-person is set at the time of creating a Collection. E-person may add contents directly to a collection or through a check/verify process by the Administrator. The default submission process in DSpace involves six steps. The first three steps are designed to describe the collection. The description level one includes the status of the file such as number of title, publication details and number of files to upload for an item. The description level two is the input form for author, title, series/report number, identifiers, type of the document and its language. The description level three is part of the input form for subject keywords, abstract, sponsors and other comments. The fourth step is intended for uploading the file. DSpace requires knowing the file format to properly archive and giving access to a file. When the system does not automatically recognize the file format, the e-person has to describe it. The fifth step is intended to verify the metadata and uploaded file. There is provision to go back in case the descriptions are incorrect and the file is wrongly selected. The sixth step contains a licence that is required by DSpace to reproduce, translate and distribute the submission worldwide. When the e-person clicks on the "Grant License" button, the uploading process becomes complete. The e-person will receive e-mail notification as soon as the completion of file uploads.

DSpace generates an MD5 checksum for every file it stores. A checksum is an error detection scheme. MD5 checksum for a file is a 128-bit value, something like a fingerprint of the file. DSpace uses this checksum internally to verify the integrity of files over time (a file's checksum is unique). When the item becomes a part of the DSpace it is assigned a persistent Uniform Resource Locator (URL). DSpace persistent URLs are registered with the Handle System, a comprehensive system for assigning, managing, and resolving persistent identifiers, known as "handles," for digital objects and other resources on the Internet. The Handle System is a technical development of the Corporation for National Research Initiates (CNRI).

4.12 Contents in DSpace

DSpace accepts a variety of digital objects that include both born digital and digitized from their analogue entity. DSpace is used to store scholarly articles, preprints, working papers, technical reports, theses, conference papers, paper clippings, books, computer programs, visualizations, simulations, multimedia contents, administrative records, journals, bibliographic datasets, images, audio files, video files, learning objects, web pages, data sets etc. These materials may be in different file formats. A file format is a standard way that information is encoded for storage in a computer file. File format is an important aspect related to the preservation of data in a digital library. DSpace recognizes and manages around seventy five file formats that include PDF, Word, JPEG, MPEG, TIFF etc. A file that is uploaded to DSpace is referred to as Bitstreams. After ingestion, files are stored on the file system as a stream of bits without the file extension. DSpace identifies two levels of digital preservation: bit preservation, and functional preservation. Through bit preservation a file remains exactly the same over time. In functional

preservation the file changes over time so that the material continues to be visible and usable in the same way. Some file formats are functionally preserved using straightforward format migration by DSpace. Users of DSpace create files in different formats. To accommodate all these formats, DSpace categorises file formats to three groups; supported, known, or unsupported. Supported formats will be subjected to functional preservation. Known formats are proprietary or binary formats the future of which is unpredictable. Unsupported formats are quite unknown to the system and there is no guaranty for functional preservation.

The default metadata schema in DSpace is Dublin Core (DC). DC schema has the potential to elicit enough information to represent a digital object. DSpace uses a qualified version of the DC schema. Every item in DSpace is represented by a DC record to provide descriptive, administrative and structural information. Descriptive metadata provides information about the scholarly contents of the item. Administrative metadata includes preservation metadata, provenance and authorization policy data. Structural Metadata is information about how to present an item to an end-user and the relationships between parts of the item. Both administrative metadata and structural metadata are generated automatically by the system while descriptive metadata is entered by the e-person.

4.13 Adoption of DSpace

The use of DSpace by a variety of institutions across the globe can be obtained from the official website of DSpace to understand the extent and scope of the system. As of December 2014, the website listed a total number of 1802 DSpace installations across 117 countries in the world. From the total number

of DSpace adoptions, it is possible to generate several kinds of lists such as continent-wise, country-wise, type of institution, DSpace version, relational database used, type of access, use cases, types of content and use of integrations or customizations. Figure 4.4 shows the continent-wise distribution of DSpace.

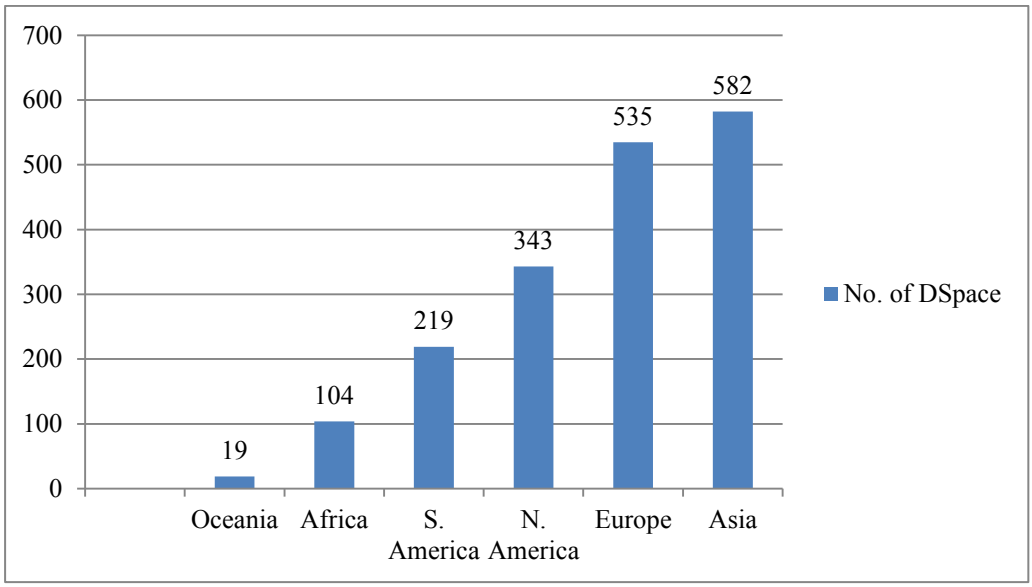

Figure 4.4 Distribution of DSpace Among Continents

Figure 4.4 displays the distribution of 1802 DSpace installations among six continents in the world. Asia has the largest number of DSpace (582) followed by Europe (535), North America (343), South America (219), Africa (104) and Oceania (19). Among Asian countries India has the largest number of DSpace systems.

Figure 4.5 presents the nine countries that have more than fifty installations of DSpace.

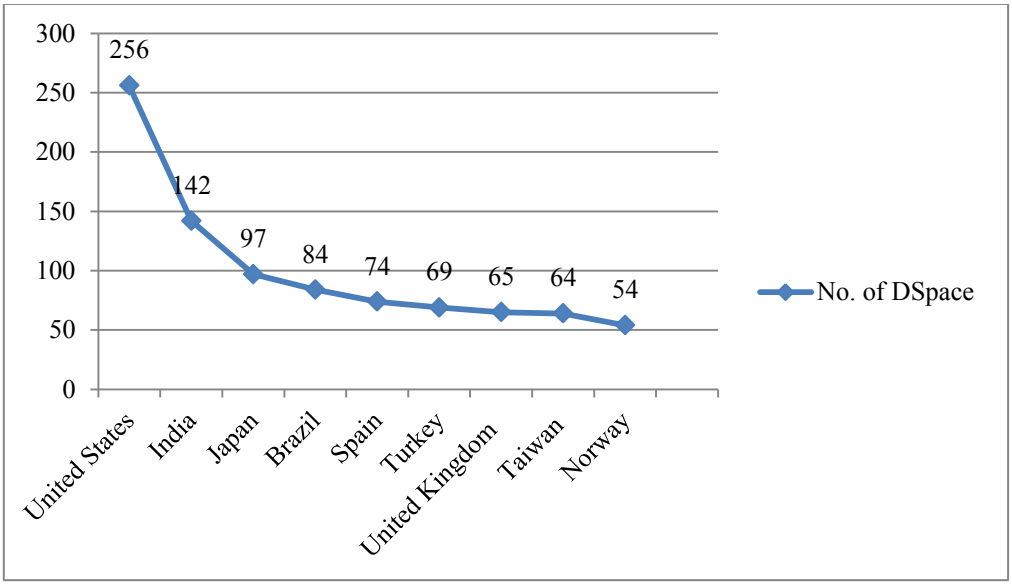

Figure 4.5 Distribution of DSpace Among Countries

The country-wise data presented in Figure 4.5 shows that United States is top with 256 (14.2%) DSpace websites. India has second position with 142 (7.8%) installations followed by Japan 97 (5.3%), Brazil 84 (5%), Spain 74 (4.2%), Turkey 69 (3.8%)United Kingdom 65 (3.6%), Taiwan 64 (3.6%), and Norway 54 (3%). The majority of DSpace adopters are educational institutions. It is relevant to note that India has the second position in the adoption of DSpace. There are several institutions in India that run a DL for archiving various kinds of documents. Apart from universities and educational institutions, banks, courts, parliament of India, and not-for-profit organizations adopt DSpace for archiving and providing open access to digital documents. Shodhganga, the Indian ETD project which accommodates around more than 30000 full text PhD theses of 160 Universities is running on DSpace software.

4.14 Conclusion

The OSS for libraries is a vibrant world. The adoption and use of OSS for meeting various library functions are increasing across the globe. Applications of OSS in libraries is an area of study and research over the world. While there are several OSS for libraries, the software for library automation and for building DLs occupy prominent place. While Koha, Evergreen and OPALS are prominently used for library automation, DSpace, EPrints, Fedora and Greenstone are well-known for DLs in the OSS domain. Among the OSS for DLs, the status of DSpace is found to be on top. While libraries are positively responding to the adoption of OSS for providing various digital contents, large numbers of libraries remain away from OSS technology due to factors ranging from lack of awareness to perception of maturity.

References

1. Arms, W. Y. (1995, July). Key concepts in the architecture of the digital library. *D-lib Magazine*, *1*(1). Retrieved from http://www.dlib.org/dlib/July95/07arms.html (accessed June 8, 2014).
2. Ayala, C. P., Cruzes, D. S., Hauge, O., & Conradi, R. (2011). Five facts on the adoption of open source software. *IEEE Software, 28*(2), 95-99.
3. Bass, M. J., Stuve, D., Tansley, R., Branschofsky, M., Breton, P., Carmichael, P., & Ng, J. (2002). DSpace–A sustainable solution for institutional digital asset services–spanning the information asset value chain: ingest, manage, preserve, disseminate. Internal reference specification: Technology and Architecture. Retrieved from http://web.mit.edu/dspace/live/implementation/design_documents/architecture.pdf (accessed June 8, 2014).
4. Breeding, M. (2002). The open source ILS: Still only a distant possibility. *Information Technology and Libraries*, *21*(1), 16.
5. Breeding, M. (2009). The viability of open source ILS. *Bulletin of the American Society for Information Science and Technology*, *35*(2), 20-25.
6. Chudnov, D. (1999). Open source software: The future of library systems? *Library Journal, 124* (13), 40-43.
7. Digital Library Foundation, (1999) "*A working definition of digital library*" Retrieved from http://www.diglib.org/about/dldefinition.htm (accessed June 5, 2014).

8. Ellis, J., & Van Belle, J. P. (2009, June). Open source software adoption by South African MSEs: Barriers and enablers. In *Proceedings of the 2009 Annual Conference of the Southern African Computer Lecturers' Association* (pp. 41-49). ACM.
9. Fitzgerald, B. (2011). Open source software adoption: Anatomy of success and failure. *International Journal of Open Source Software & Processes, 1* (1), 1-23.
10. Griffin, S. M. (1998, July/August). NSF/DARPA/NASA digital libraries initiatives: A program manager's perspective. *D-Lib Magazine, 4* (7/8). Retrieved from http://www.dlib.org/dlib/july98/07griffin.html (accessed June 9, 2014).
11. Helly, J. J., Elvins, T. T., Sutton, D., & Martinez, D. (1999). A method for interoperable digital libraries and data repositories. *Future Generation Computer Systems*, *16*(1), 21-28.
12. Kahn, R., & Wilensky, R. (1995). A framework for distributed digital object services. *International Journal on Digital Libraries*, *6*(2), 115-123.
13. Li, A. (2008). Investigation of ILS in Chinese main academic libraries. *Library Collections, Acquisitions, and Technical Services*, *32*(3), 115-120.
14. Marchionini, G. and Fox, E.A. (1999). Progress toward digital libraries: Augmentation through integration. *Information Processing & Management*, 35 (3), 219-225.
15. Mijinyawa, K. (2008). *An evaluation of open source software adoption by UK SMEs in the IT industry* (Doctoral dissertation). Retrieved from http://bura.brunel.ac.uk/bitstream/2438/4509/1/FulltextThesis.pdf (accessed June 5, 2013).

16. Pappalardo, K. M. & Fitzgerald, A. M. (2007). A guide to developing open access through your digital repository. Retrieved from http://eprints.qut.edu.au/9671/1/9671.pdf (accessed May12, 2013)
17. Pomerantz, J., & Marchionini, G. (2007). The digital library as place. *Journal of Documentation*, *63*(4), 505-533.
18. Rossi, B., Russo, B., & Succi, G. (2012). Adoption of free/libre open source software in public organizations: Factors of impact. *Information Technology & People*, *25*(2), 156-187.
19. Singh, V. (2013). Experiences of migrating to open source integrated library systems. *Information Technology and Libraries*, *32*(1), 36-53.
20. Smith, P. L. (2008). Where IR you? Using "open access" to extend the reach and richness of faculty research within a university. *OCLC Systems & Services: International Digital Library Perspectives*, *24*(3), 174-184.
21. Tansley, R (2003, August) *DSpace: Preserving digital data for the ages* Retrieved from http://www.hpl.hp.com/news/2003/july_sept/dspace.html (accessed July 8, 2013).

Chapter 5 Analysis

5.1 Introduction

This chapter stands for the analysis part of the study. Here, the investigator measures the maturity of DSpace software. The Woods and Guliani's OSMM-2005 model (hereafter WG-OSMM) is the instrument to measure the maturity of DSpace. The WG-OSMM involves fifteen major elements to determine the status of maturity of OSS. These elements form the research questions in this thesis. The elements of maturity are listed below;

- Leadership and culture
- Vitality of community
- Quality of end-user support
- Extent and scope of documentation

- Quality of packaging
- Momentum
- Quality of code and design
- Quality of architecture
- Testing practices
- Integration with other products
- Support for standards
- Quality of project site
- License type
- Potential for commercial conflicts
- Corporate commitment

These fifteen elements of maturity are the functional specifications of requirements of an OSS. They are a combination of technical features as well as software environment. The leadership and culture, vitality of community, quality of end-user support, momentum, quality of project site, potential for commercial conflicts and corporate commitment are related to the environment of a software. The extent and scope of documentation, quality of packaging, quality of code and design, quality of architecture, testing practices, integration with other products, support for standards and license type are the technical aspects of a software. These specifications are matched against DSpace. The WGOSMM offers detailed descriptions for each element in order to understand the meaning of the element. The investigator has recognized sub elements under each major element to aid collection and presentation of data. These sub elements are not subjective to the investigator, but derived from the descriptions given for each element by the WGOSMM model. Data for the study were collected from multiples sources with official

websites of DSpace being the major one. The following are the analysis of the study

5.2 Leadership and Culture

The first major element of maturity is OSS leadership and culture. WG-OSMM distinguishes the quality of leadership as one of the most important factors of OSS maturity. The WG-OSMM considers open source a living thing that needs on-going nourishment, encouragement, and care. And without leadership of some kind, an open source program will wither and eventually die. WG-OSMM also expects a strong, professional, respectful culture for a mature OSS. The WG-OSMM seeks to understand the following sub research questions on leadership and culture; Identification of Leadership, Quality of Leadership, Previous Experience, Participation in Forums and Conferences, Significant Contributions, Response to Questions or Suggestions and the Project Culture. The investigator examines each element in detail

5.2.1 Identification of Leadership

WG-OSMM confirms whether the leadership of an OSS project is identifiable. This question is relevant when many projects are started by individuals and left unfinished, unrefined and unusable for any purpose. The identification of leadership behind the DSpace project is not difficult as several documents, both online and print, are available for verification. This element is related to the historical milestones of a project. For reliable data, the investigator relied on the official website of DSpace hosted in 2001 and 2002. The leadership of DSpace can be divided into two categories; Institutional and Individual. Institutional leadership denotes the organizations behind DSpace software.

The individual leadership implies the diversity of individuals from different field of knowledge who joined DSpace project from time to time

5.2.1.1 DSpace: Institutional Leadership

DSpace is a joint project of Massachusetts Institute of Technology (MIT) Libraries and Hewlett Packard (HP) Lab begun in 2002. MIT is a well-known world-class educational institution established in 1861 in the United States. MIT is a top ranking institution in the QS World University Rankings as well as the Times Higher Education World University Rankings for several years. MIT has numerous academic departments, divisions, and degree-granting programs, as well as interdisciplinary centers, laboratories, and programs. MIT Libraries are the important part of MIT sharing the values and qualities of the parent institution. The MIT Libraries create and sustain an evolving information environment that advances learning, research, and innovation at MIT. They are committed to excellence in services, strategies, and systems that promote discovery, preserve knowledge, and improve worldwide scholarly communication (MIT Library website, 2014). The HP Lab is the leading multinational company headquartered in California, United States. The company is associated with hardware, software and services segment. It has 45 years of experience in the field.

The reputation of MIT and HP indicates that DSpace had the patronage of strong institutional leadership. The data available on the official website of DSpace as of 2014 shows that the DSpace leadership was shifted to DSpace Federation in 2004 and DSpace Foundation in 2007 and DuraSpace Organization in 2009. DSpace Federation was formed by a group of institutions and the DSpace Foundation was a non-profit organization and DuraSpace is an independent and non-profit organization providing leadership

to open source technologies. Currently, Debra Hanken KurtzMichele is the Chief Executive Officer of DuraSpace who was formerly Assistant Director of Information Technology Services and Head of Digital Experience Services at Duke University libraries and director of the Texas Digital Library (TDL).

5.2.1.2 DSpace: Individual Leadership

The identification and listing of individual leadership is limited to those who were part of DSpace software during the 2000-2002. The DSpace project was started from 2000 onwards. The earliest archived official website of DSpace was found to be on 2nd May, 2001 which shows the names of individuals who were the project team leaders and steering committee members of the project. The website was updated in 12th December 2011 showing the names of more team members. Table 5.1 shows the name, title and association of the project team members

Table 5.1 Project Team Leaders of DSpace (2001)

Sr.No	Name	Title	Association
1	Michael Bass	Project Leader	HP
2	Margret Branschofsky	Faculty Liaison	MIT
3	Peter Breton	Consultant	White Rabbit
4	William Cattey	Senior Developer	MIT
5	Joyce Ng	Research Assistant	MIT
6	David Stuve	Senior Developer	HP
7	Robert Tansley	Developer	HP
8	Mary Barton	Senior Marketing Development Manager	Analysis Group/Economics
9	Peter Carmichael	Consultant	PC-Consulting
10	Daniel Chudnov	Systems Curator	MIT
11	Julie Harford	Marketing Development Manager	Adero

Table 5.1shows the eleven project team members of DSpace. The project team was a combination of HP and MIT professionals with the support of members from consultant companies of software and business. Out of eleven, seven team members were from computer science and two each from business and library science respectively. Margret Branschofsky and Daniel Chudnov belonged to the library profession.

The steering committee of DSpace provided oversight of the overall project. There were six members in the steering committee. The Table 5.2 shows the names of steering committee members.

Table 5.2 Steering Committee of DSpace (2001)

Sr.No	Name	Association
1	Steve Brown	HP
2	Eric Celeste	MIT
3	William Wickes	HP
4	Ann Wolpert	MIT
5	Robin Gallimore	HP
6	Nick Wainwright	HP

Table 5.2 shows the names of six steering committee members. Steve Brown and William Wickes were from the computer science field and Eric Celeste and Ann Wolpert were having library science background. Wolpert, (who died on 2nd October 2013) was the director of MIT's libraries for seventeen years. She conceived the idea of a common, permanent repository platform for digital materials as a solution for preserving the intellectual heritage.

The presentation of data on the leadership leads to the inference that that DSpace had a support of two prestigious institutions and the individual leadership comprised of professionals from computer science, library science and business disciplines.

5.2.2 Quality of Leadership

The aspect of quality of leadership behind OSS is examined to verify whether they are serious developers with a strong understanding of technology. Data displayed on the Table 5.1 and 5.2 showed the names and association of project team members and steering committee members. It is disclosed that the leadership of DSpace came mainly from MIT and HP lab and on the basis of the reputation of both entities, the quality of the leadership can be confirmed.

The quality of the leadership can be further verified by examining their previous experience.

5.2.3 Previous Experience

The previous experiences and accomplishments of team leaders play an important role in the design of a new project and these attributes score for the maturity of software. The data regarding the previous experience of the team leaders of DSpace shows that all were having previous background in diverse fields and projects. Table 5.3 and 5.4 show the names of team leaders and steering group members arranged alphabetically with their previous experience.

Table 5.3 Previous Experiences of DSpace Team Leaders

Sl. No.	Name	Experience
1	Daniel Chudnov	Systems Architect, Cushing/Whitney Medical Library, Yale School of Medicine. New Haven, CT Programmer / Systems administrator University of Michigan Medical Center
2	David Stuve	Software engineer for HP. Team leader for a HP's ink jet printer division. Ran a consulting company that specialized in graphics tools and firmware for networked devices.
3	Joyce Ng	Experience with HP (knowledge management, Architecture Technology Group, Mountain View), ArsDigita, and Lotus.
4	Julie Harford	Strong background in product management and business planning. Senior Product Manager at Adero, an Internet content distribution service.

5	Margret Branschofsky	Professional librarian in academic science and engineering libraries at University of Cincinnati and MIT. Previous experience in library automation and computer programming.
6	Mary Barton	Senior Marketing Development Manager in the MIT Libraries Digital Library Research Group. Holds MBA from the MIT Sloan School of Management and has a background in finance and microeconomics.
7	Michael Bass	11 years of hardware and software design, and program management experience with Hewlett-Packard Company. Designed hardware and software contributing to HP's Precision Architecture microprocessors.
8	Peter Breton	A software developer since 1994 with White Rabbit Software company. Specialized in Web development in Java and XML. Worked with Swiss ISP, and with the Department of Education.
9	Peter Carmichael	A software developer since 1983. Specializing in Java, OOMD, Graphical User Interfaces, and Evolutionary Prototyping. Developed a Palm application for wireless order entry, and GUI and middleware design and development in Java & C++ for commercial and government projects.
10	Robert Tansley	PhD in the application of semiology to multimedia information at the University of Southampton in 2000. Designed and implemented the Eprints software. Involved in the specification and alpha-testing of the OAI-PMH
11	William Cattey	Senior Analyst Programmer for Information Systems at MIT. Part of Athena project at MIT

Table 5.4 Previous Experiences of DSpace Steering Group Members

Sl. No.	Name	Previous Experience
1	Ann Wolpert	Director of MIT Libraries since 1996 and a pioneer in digital libraries, open access, open courseware. Background in applied technology in libraries. Postgraduate in library science from University of Chicago.
2	Eric Celeste	Had combination of library and technology experience. Developed Uthink blog and Minnesota University Digital Conservancy. Technical leader of SHARE project.
3	Nick Wainwright	Research department manager at HP. Researcher in information infrastructure for future Internet services.
4	Robin Gallimore	Director of the Publishing Systems and Solutions Laboratory to create HP Labs' digital media solutions research program.
5	Steve Brown	Marketing manager at HP Labs.
6	William Wickes	Asst. Professor of Physics at Princeton University and University of Maryland. Development of advanced scientific calculators, portable computers and operating systems.

Table 5.3 and 5.4 show that DSpace had an experienced leadership and majority of them were having strong background in computer technologies. The library and information professionals and business experts were also having several years of experience in their field. This data also satisfies the sub element of quality of leadership discussed in section 5.2.2

5.2.4 Participation in Forums and Conferences

WG-OSMM looks for the degree to which a project leader or other team members participate in the many forums and mailing lists and their presence at important conferences to make significant contribution to the public knowledge base for the product. Team leaders of a mature product work for the promotion of the software. To examine this sub- research element, it is necessary to find out the mailing list maintained by the DSpace project. Both the official website of DSpace and the sourceforg.net website give the details of DSpace mailing list which are given in Table 5.5

Table 5.5 Status of DSpace Mailing list

Sr.No	Name	Purpose	Active Since	Status
1	DSpace-tech	Technology discussion list. Deals with technical support, installation, configuration, and customization	November 2002	Live
2	DSpace-general	General discussions, announcements about non-technical aspects like services, policies, legal issues, features and functions, etc.	August 2003	Live
3	DSpace-devel	For developers-discussions about the DSpace core code	April 2006	Live
4	DSpace-changelog	For developers-notifications of every commit to the codebase	August 2006	Live
5	DSpace-release	For developers/Others and discussion of the next release of the DSpace	May 2012	Live
6	DSpace-tickets	For Developers notification of any ticket activity from issue tracker	July 2014	Live

Table 5.5 shows that there are six mailing lists for DSpace. The archives of the list are available online with sourceforg.net website. DSpace-tech is the oldest mailing list started from November 2002 onwards followed by DSpace-general (2003), DSpace-devel and Dspace-changelog (2006), DSpace-release (2012) and DSpace-tickets (2014).

The examination of the participation of the team members in various mailing lists was achieved with the help of a simple computer programming to extract data from all mails archive. The presentation of data is limited to dspace-general and dspace-tech mailing lists since they exist for users and more mails are exchanged through these two forums. The Figure 5.1 and 5.2 show the names of team leaders and the number of mails posted by them in DSpace-tech and DSpace-general mailing list respectively.

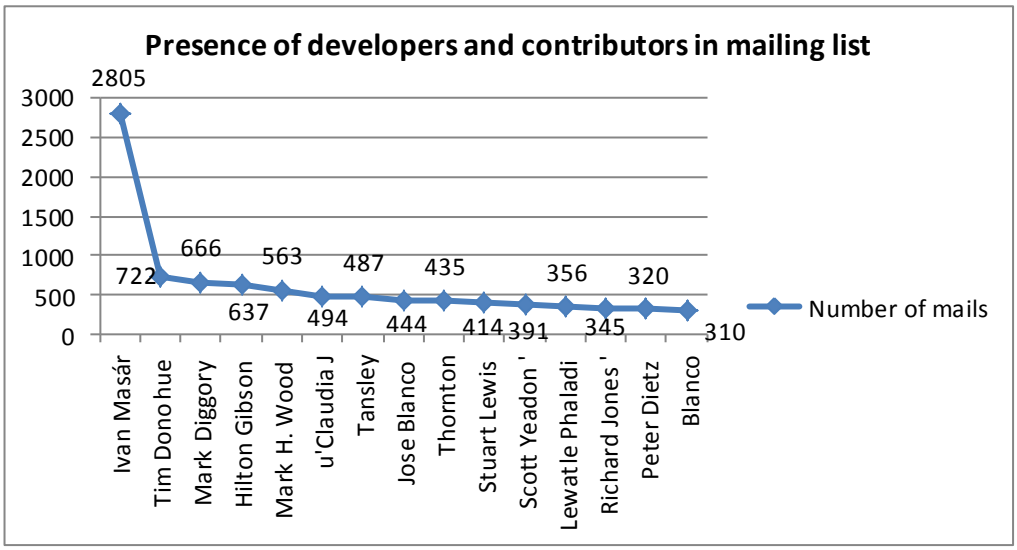

Figure 5.1 Presence of Leaders in DSpace-tech Mailing List

Figure 5.1 shows the names of 15 team members of DSpace who handled more mails. The names of team members mentioned in Table 5.1 and 5.2 are those who were leaders at the initial stage of DSpace. The names shown in

Figure 5.1 include leaders in different capacities from 2002 to 2014. Ivan Masar is top with 2805 mails followed by Tim Donohue, Mark Diggory, Hilton Gibson, Mark H.Wood, u'Claudia and Tansley. Ivan Masar is a DSpace committer's team member who joined the project in 2012. Tim Donohue is the technical lead for the DSpace project at DuraSpace from 2009 onwards. Mark Diggory was a member of development team since 2008. Tansley was a team member in the beginning of DSpace. All others mentioned in the figure 5.1 are associated with DSpace in various positions. There are a total of 35784 mails communicated through DSpace-tech mailing list during November 2002 to December 2014. Figure 5.2 shows the presence of ten DSpace leaders in the DSpace-general mailing list with the number of mails.

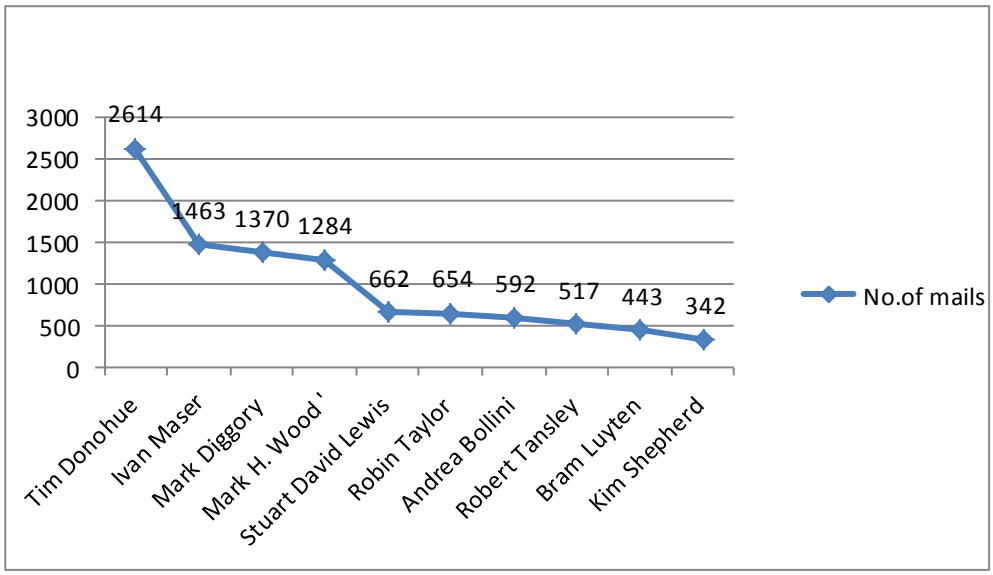

Figure 5.2 Presence of Leaders in DSpace-general Mailing List

There are 5862 mails communicated through DSpace-general mailing list during August 2003 to December 2014. Figure 5.2 displays the presence of ten DSpace team leaders in the mailing list. Leaders were very active in DSpace-general mailing list. Tim Donohue is top with 2614 mails followed by

Ivan Masar (1463), Mark Diggory (1370), Mark H. Wood (1284), Lewis (662), Taylor (654), Bollini (592), Tansley (517), Luyten (443) and Shepherd (342).

The examination of two DSpace mailing list showed that DSpace leaders belonging to different periods of the project were active in exchanging mails. Viewing from the maturity aspects of an OSS, it is very important that leaders of an OSS project are visible on its mailing list.

5.2.5 Significant Contributions of Team Leaders

The effort to make the project known to the public is important because lack of awareness inhibits the adoption and use of a product. There are several channels of communication to disseminate knowledge of an OSS. Conferences, workshops and publications are the three important ways of publicizing a project. DSpace leaders have extensively used these ways to popularize the software from 2002 onwards. The first paper on DSpace was presented in the 1st Joint Conference on Digital Libraries (JCDL) held at Roanoke, USA by Michael J. Bass and Margret Branschofsky. Daniel Chudnov published an article on DSpace in *serials* journal of November 2001 issue. These papers are still available online. Danile Chudnov along with Margret Branschofsky presented a paper on DSpace in the 2nd Joint Conference on Digital Libraries (JCDL) held at Portland, USA during July 14-18, 2002. MacKenzie Smith, Mary Barton, Mick Bass, Margret Branschofsky, Greg McClellan, Dave Stuve, Robert Tansley, and Julie Harford Walker presented a paper on DSpace at the 3rd ACM/IEEE-CS joint conference on Digital libraries held at Houston, Texas in 2003. A list of papers published by the leaders of DSpace has been given as Appendix A.

The approach of DSpace team leaders in attending conferences and presenting papers and the publishing of articles in journals is significant. Moreover, these resources are still available online on multiple locations.

5.2.6 Responses to Questions or Suggestions

The response of team leaders towards questions or suggestions from users for changes and new features is very important. Most open source developers communicate through mailing lists and they are a rich source of information which researchers can use to understand software processes and improve development practices (Shihab et al, 2010). The examination of the archive of six mailing list of DSpace shows that there are more than seventy five thousand mails that include questions, announcements, clarifications etc. The examination of a sample of answers provided by the DSpace leaders' showed that they were positive to questions and suggestions from users. The positive attitude of DSpace leaders can also be verified by the inspection of added features to DSpace from time to time based on the suggestions from user community.

5.2.7 Project Culture

Another crucial measure of an OSS project is its culture. This sub research question demands to examine the attitude of the project, its response to questions or suggestions for changes or new features and the degree of defensiveness.

The verification of data from mailing lists, DSpace documentation and from historical documents shows that DSpace software maintained a professional culture throughout its various stages of development. The DSpace project was materialized out of a commitment and strategy maintained at MIT. The

commitment was to manage the exponential growth of digital materials as they posed access and archiving problems. DSpace was developed out of the obligation that institutions could and should accept stewardship responsibility for wide-spread and long-term access to their intellectual output. On the strategic part it was decided to build a simple digital archiving solution that any academic institution could use with minimal configuration and customization. The MIT's commitment of openness of educational technology was extended to DSpace and it became an open source project.

When the DSpace 1.0 was released in November 2002, innovations in digital archiving were at the initial stage. Hodge (2000) observed that while there were traditions of stewardship and best practices that have become institutionalized in the print environment, many of these traditions were inadequate, inappropriate or not well known among the stakeholders in the digital environment. The project responded positively to suggestions for new features.

5.3 Vitality of Community

The vitality of community is the second major element of OSS maturity. Since OSS is a community driven project, it is essential to understand the vitality of community. Under this, WG-OSSM requires examining the following sub research questions; division of labour between developers and users, the size of the community, the liveliness of forums and the number and frequency of downloads. The investigator has attempted to examine each sub-element in detail.

5.3.1 Division of Labour

This element of maturity seeks to understand the relationship between project's developers and users. A mature product, as per WG-OSMM, keeps separate mailing list for users and developers and the releases of easy to use installation packages/ documentation for the users.

The Table 5.5 showed that DSpace maintains a total number of six mailing lists namely DSpace-tech, DSpace-general, DSpace-devel, DSpace-changelog, DSpace-release and DSpace-tickets. The Figure 5.3 displays the two types of DSpace mailing list.

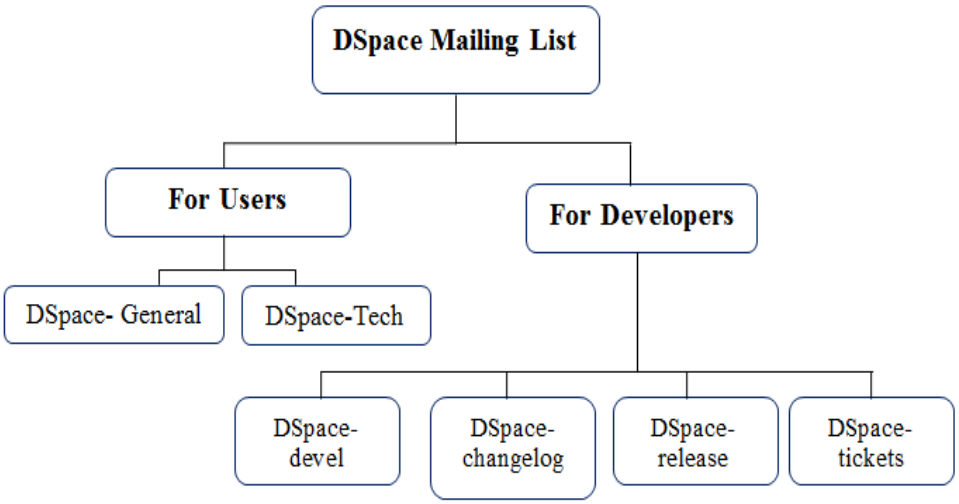

Figure 5.3 DSpace Mailing List for Users and Developers

Figure 5.3 shows six mailing lists of DSpace. Out of six, the two- DSpace-general and DSpace-tech – are devoted to users of DSpace. The remaining four mailing lists exist for the developers. While the DSpace-general is intended for general discussions and announcements about non-technical aspects like services, policies, legal issues, features and functions DSpace-tech provides

technical support and it is a platform for question and answers regarding installation, configuration and customization.

The availability of easy to use installation packages/ documentation for the users of the software is an essential part of OSS. Lack of good documentation affects the success and sustainability of any OSS. The absence of proper documentation has been regarded as a risk of an OSS project (Bell, Ng & Lambros, 2003). The official website of DSpace keeps documentation for all versions of the software in online and downloadable format. The practice of creation of easy-to-use installation packages or user documentation is very active in DSpace. Table 5.6 shows the availability of documentation associated with all stable versions of DSpace from 2002 onwards.

Table 5.6 DSpace Documentation Status

Version	**Date**	**Documentation**
DSpace 1.0.x	04-11-2002	Yes
DSpace 1.1.x	08-05-2003	Yes
DSpace 1.2.x	05-05-2005	Yes
DSpace 1.3.x	09-10-2005	Yes
DSpace 1.4.x	10-05-2007	Yes
DSpace 1.5.x	14-04-2009	Yes
DSpace 1.6.x	15-06-2010	Yes
DSpace 1.7.x	25-07-2013	Yes
DSpace 1.8.x	25-07-2013	Yes
DSpace 3.x	24-07-2013	Yes
DSpace 4.x	03-03-2014	Yes
DSpace 5.x	16-01-2015	Yes

Table 5.6 shows that releasing easy-to-use installation packages is core to DSpace project. It keeps documentation for all versions. The length of

documentation varies. The documentation for the first version of DSpace was around 10 pages whereas the latest versions contain user manuals having around 700-800 pages.

The two aspects examined under division of labour satisfy the requirement of a mature project. DSpace maintains separate list for users and developers and it has easy-to-use installation packages for all stable versions.

5.3.2 The Size of the Community

The size of the OSS community is an important factor for its success and sustainability. The examination of official data revealed that the DSpace community is a large entity comprising of several components. The Figure 5.4 shows the different group that form the size of DSpace community.

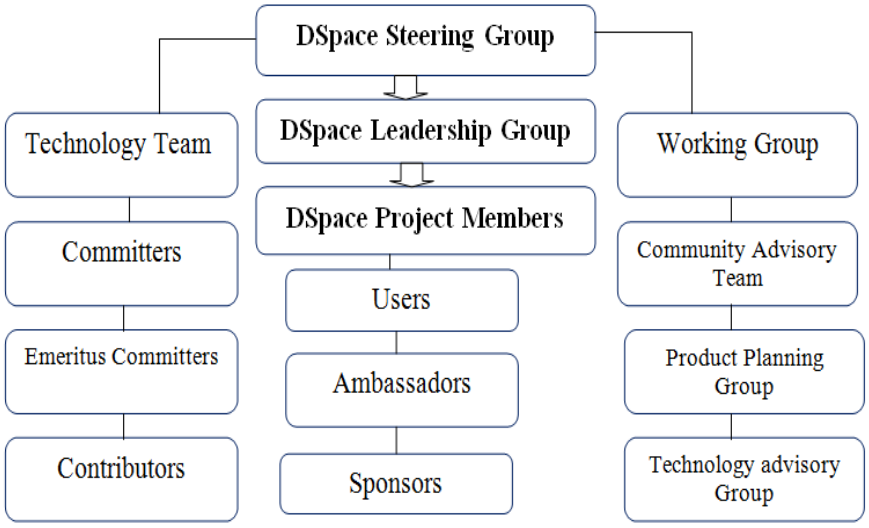

Figure 5.4 Size of DSpace Community

The Figure 5.4 shows that DSpace has a strong base of community structure with Steering Group at the apex. There are DSpace Steering Group,

Leadership Group, and Project Members at the core of the community. The other components include Committers, Emeritus Committers, Contributors, Community Advisory Team, Ambassador Programme Members, Sponsors and Users. The following part attempts to understand each component of DSpace community.

5.3.2.1 DSpace Steering Group

The DSpace Steering Group provides support for leadership and sets strategic direction for DSpace software. They oversee project operations and recommend annual budget allocations. Steering Group is nominated and elected by the DSpace Leadership Group. Currently there are 8 members in the group.

5.3.2.2 DSpace Leadership Group

The DSpace Leadership Group approves the overall priorities and strategic direction of the project presented by the Steering Group. It is a subset of the overall DSpace Members, selected based on their level of contribution to DSpace.

5.3.2.3 DSpace Project Members

DSpace Project Members are not directly involved with decisions regarding the DSpace platform. However, they may provide their feedback via member-directed surveys or similar.

5.3.2.4 Committers

The DSpace Committers are members authorized to commit change to the code repository. They are having the final responsibility for the form, architecture and design of DSpace going forward. They are applying code changes contributed by the larger DSpace development community. As of 31st December 2014 there are 23 members in the DSpace Committer Group who belong to diverse institutions. Eight members belong to different universities in the world and the five out of them are from library field. Twelve members belong to different firms that support DSpace.

5.3.2.5 Emeritus Committers

Emeritus Committers are those who, for one reason or another, are no longer able to contribute code to DSpace on a regular basis. As of 31st December 2014 there are 12 emeritus committers in DSpace who are providing advice and guidance for the software development.

5.3.2.6 Contributors

Contributors are the largest component of DSpace Community who report and fix a bug, provide a new feature, help with documentation, or otherwise contribute to the software product. As of 31st December 2014 there are 162 contributors to DSpace software who are comprised of individuals as well as institutions and have contributed to at least one version of DSpace.

5.3.2.7 Community Advisory Team (DCAT)

The DCAT stands for the service of repository managers and administrators at the global level. They are also representing the interest of DSpace end users

indirectly. It is a permanent Working Group that advises other DSpace project governance and leadership groups. DCAT solicits feedback through community-wide discussions, surveys, etc. to help ensure future software releases.

5.3.2.8 DSpace Product Planning Group

The Product Planning Group develops and maintains the DSpace Product Plan in conjunction with the DCAT and the Technology Advisory Group.

5.3.2.9 DSpace Technology Advisory Group

The DSpace Technology Advisory Group advises all groups on DSpace technology and architectural decisions. They help to research and/or prototype various implementation options, and recommend the "best of class" for implementation.

5.3.2.10 DSpace Ambassadors

The DSpace Ambassador is a volunteer in every country or region to be a point of contact for organizations just getting started with DSpace. They help new or potential users interested in adopting DSpace. Ambassadors are also encouraged to help build the DSpace user network within their country. Individuals with DSpace knowledge and experience may become an Ambassador. There are 48 Ambassadors for DSpace across the world.

5.3.2.11 DSpace Sponsors/Members

DSpace Members are leaders from university, research, library organizations, and others, who have made a financial commitment to DSpace project. There

are Platinum, Gold, Silver and Bronze members who contribute $20,000, $10000 and $5000 and $2500 respectively. There are currently three Platinum members, five gold members, six silver members and nine bronze members for DSpace.

5.3.2.12 DSpace Software Users

Data from the official website of DSpace shows that DSpace has a large community of users. As of October 2014, the official website lists a total number of 1781 DSpace installations spread across 117 countries in the world. Figure 5.5 shows the top nine countries having more than 50 DSpace installations in the world.

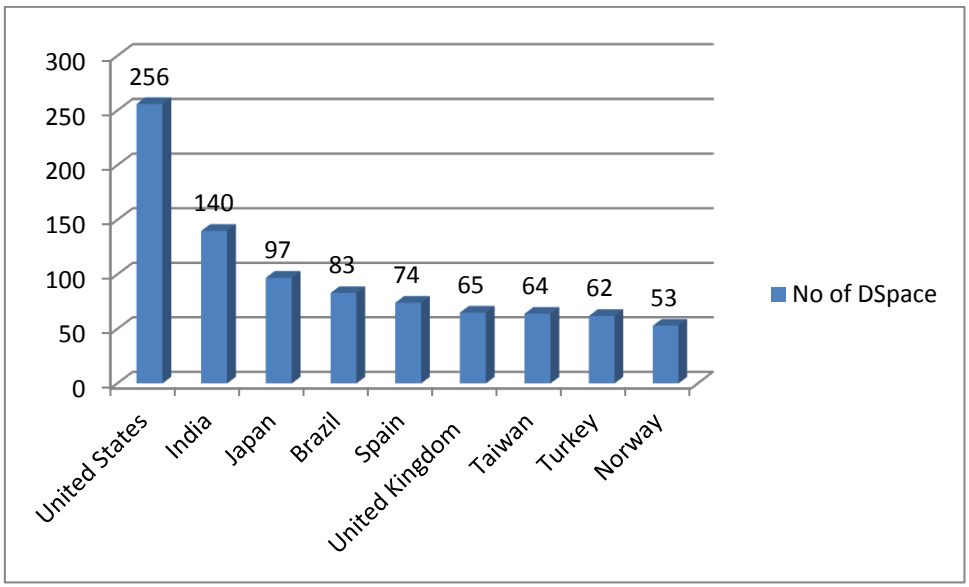

Figure 5.5 Countries Having More DSpace Installations

Figure 5.5 shows the ten countries that have more DSpace installations. United States is the major country having more DSpace installations followed by India, Japan, Brazil, Spain, United Kingdom, Taiwan, Turkey and Norway.

5.3.3 The Liveliness of Forums

The examination of mailing lists shown in Table 5.5 revealed that there are total six mailing lists for DSpace and the Figure 5.3 showed the existence of separate mailing list for users and developers. The number of emails exchanged through these mailing lists indicates the liveliness of forums. The Figure 5.6 shows the number of mails communicated through the six mailing lists from the beginning to December 2014.

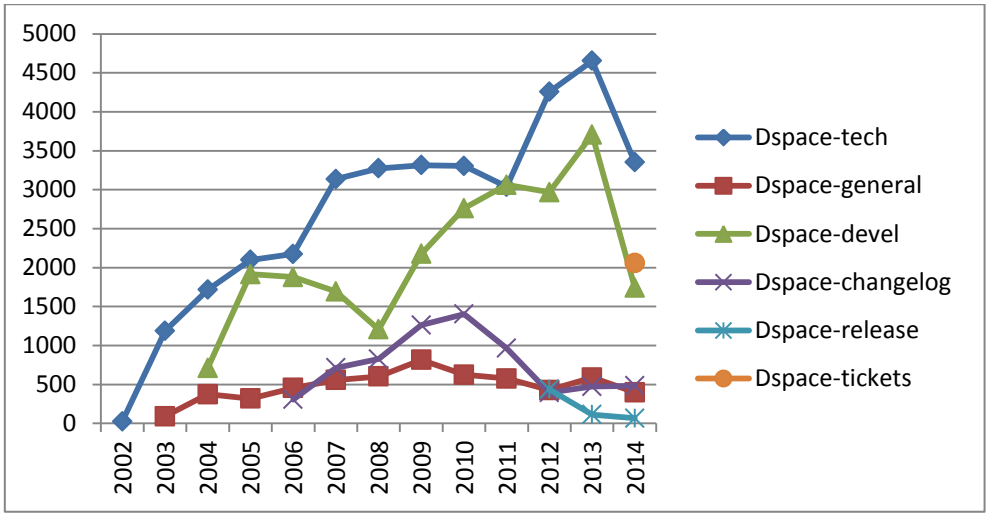

Figure 5.6 Liveliness of Forums

The Figure 5.6 shows that DSpace mailing list was very active in exchanging communication among leaders and users. The DSpace-tech mailing list contains more number of mails (35784), followed by DSpace-devel (23841), DSpace-changelog (6870), DSpace- general (5862), DSpace-tickets (2445) and DSpace-release (626). From the data obtained, it can be concluded that the DSpace mailing lists were very active. The involvement of DSpace team leaders in the two forums shown in Figure 5.1 and 5.2 adds more value to the liveliness of mailing lists.

5.3.4 The Frequency of Downloads

The number and frequency of downloads provide information on the status and popularity of a software product. The information on the download status of open source projects is available in the sourceforge.net website. The data for DSpace downloads were collected for the period of 2002 to 2014. Figure 5.7 shows the number of downloads of DSpace from November 2002 to December 2014.

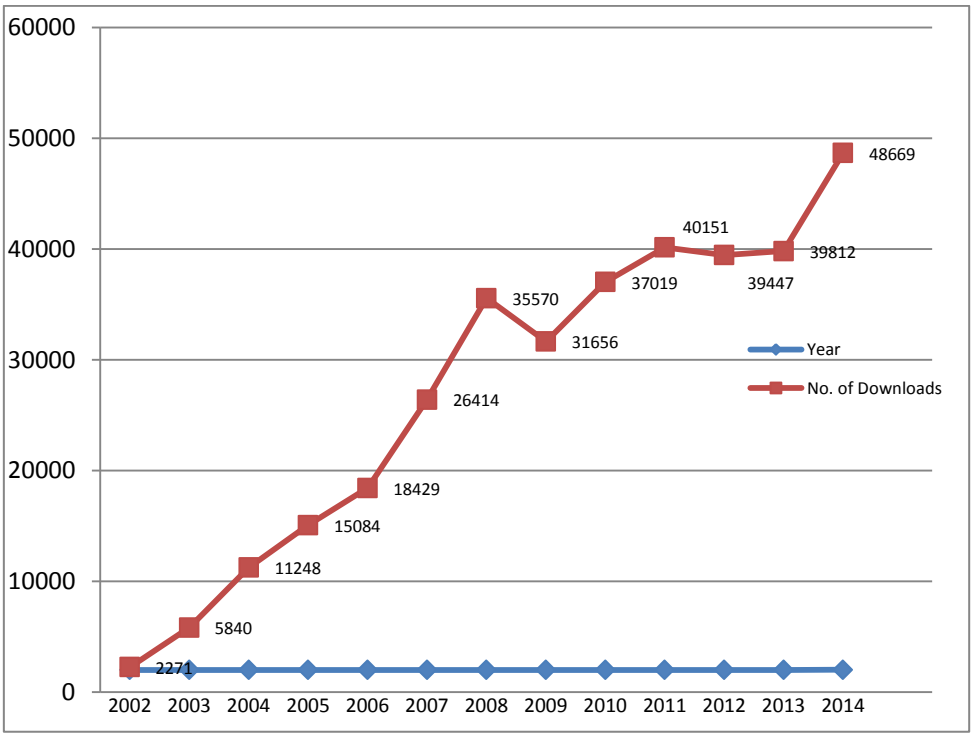

Figure 5.7 Download Statistics of DSpace

Figure 5.7 shows that DSpace received more than 3.5 lakhs downloads during 2002-2014. The statistics indicate a balanced growth of downloads for the project except in 2009. These download statistics correspond to various versions of DSpace. More downloads were registered in 2014 the years in which DSpace 4.x was released. Many factors influence the process of OSS

downloads. While every download would not result in an adoption of the software, the download statistics provide inference on the interest of users towards the software.

From the data obtained, it is possible to generate the frequency of downloads. The Figure 5.8 shows the frequency of DSpace downloads.

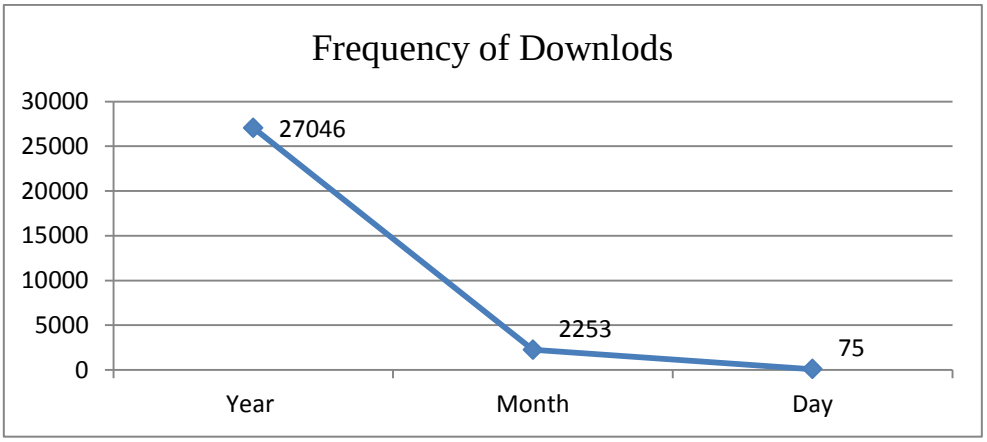

Figure 5.8 Frequency of Downloads

The Figure 5.8 show that DSpace is receiving an average 75 downloads per day, 2253 downloads per month and 27046 download per year. The download statistics of DSpace was compared with other OSS in the same category for the years 2014. The Figure 5.9 shows the download statistics of three OSS for DLs; DSpace, Greenstone and Fedora Commons.

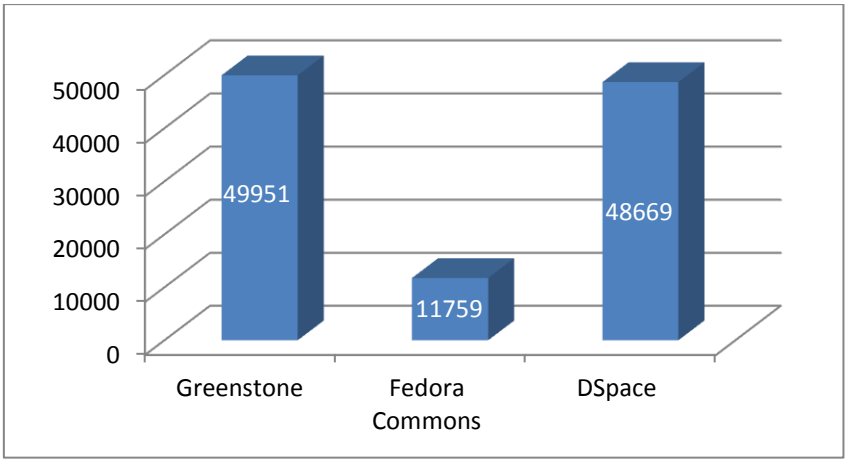

Figure 5.9 Download Statistics of Three OSS for DLs During 2014

Figure 5.9 shows that Greenstone and DSpace received more downloads during 2014 compared to Fedora Commons software.

5.4 Quality of End-User Support

The third major element of maturity, as per WG-OSMM, is the quality of end-user support. It is a key element for the understanding and installation of any OSS. The quality of end user support can be verified by examining the existence of active forums, well-maintained FAQs (Frequently Asked Questions), and online documentation that are available through a search engine. The investigator attempts to examine the extent of end user support applicable to DSpace.

5.4.1 Active Forums

WG-OSMM envisages the very public and free-ranging discussions among the lead developers, far-flung contributors and end users as one of the most compelling aspects of open source projects. The data on the availability of six

mailing list of DSpace was shown in Table 5.5. The Figures 5.1 and 5.2 provided data on the presence of DSpace leaders in DSpace-tech and DSpace-general mailing list. The Figure 5.6 displayed the liveliness of DSpace forums with a description of the total number of mails exchanged through these forums from the beginning to December 2014. A further exploration of the mailing forums is attempted here.

Figure 5.10 shows the six mailing lists and the number of mails exchanged from their beginning to December 2014.

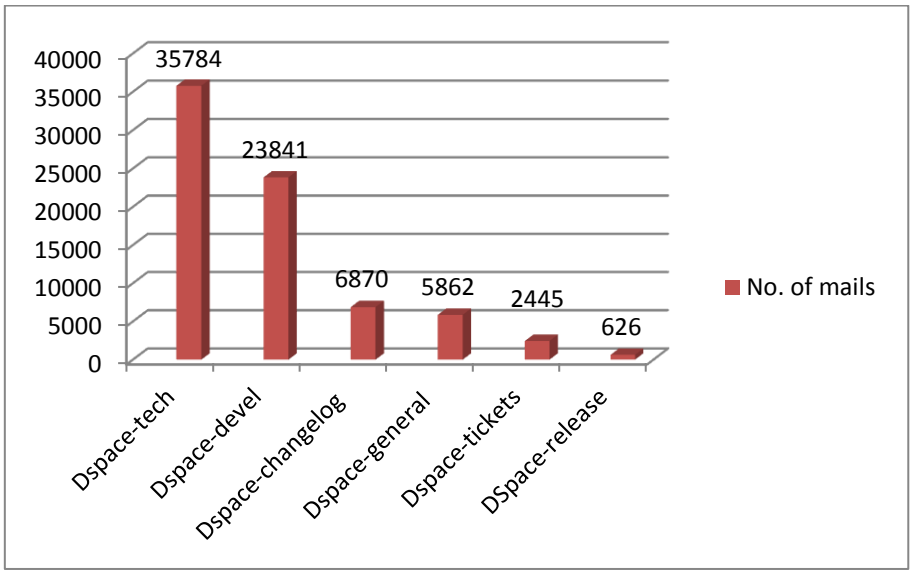

Figure 5.10 DSpace-Mailing List and Number of Mails

Figure 5.10 displays that among the six mailing list maintained by DSpace, DSpace-tech is the most active forum that deals with technical questions answered by the DSpace community. DSpace-devel stands for developers writing code for the DSpace. It also contains release information for DSpace committers. The DSpace-changelog is also for developers that handles notifications of every commit to the codebase of DSpace. DSpace-general deals with questions, announcements, and discussions about non-technical

aspects of DSpace including services, policies, legal issues, features and functions, etc. DSpace-tickets is a notification-only mailing list. Members of this mailing list receive an email notification whenever any ticket in the DSpace Issue Tracker (JIRA) is created/updated/closed. This was started in 2014 only and these issues were accomplished through DSpace-devel mailing list before. DSpace- release is meant for developers for planning and discussing issues on the future release of DSpace. Figure 5.11 shows the number of mails exchanged through DSpace-tech mailing list from 2002 to 2014.

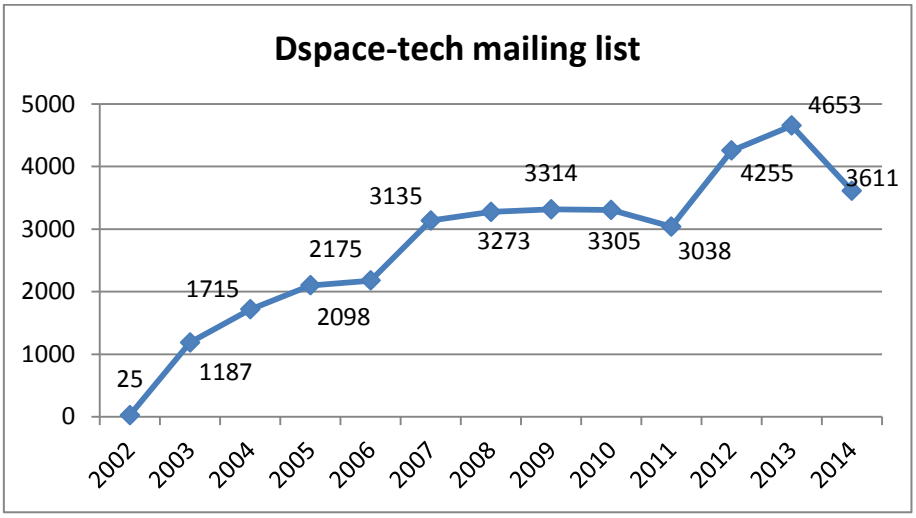

Figure 5.11 DSpace-tech Mailing List

Figure 5.11 shows that DSpace-tech mailing list contained less number of mails during 2002-2014. The number of mails increased to 1187 in 2003 to 1715 in 2004.

The movement of mails depends on many factors including the release of a new version, integration of a new service or applications to the new version etc. There were a total number of 35784 mails during 2002-2014 with a yearly average of 2752. mails. The examination of sample of mails revealed that the

discussions were more intensive on technical aspects of DSpace. There were around 2000 members who participated in the discussion at the level of lead developers, far-flung contributors and end users. The investigator could identify many users from India who participated in the discussions.

5.4.2 Archives of Questions and Answers

The examination of the details of DSpace mailing list on sourceforge.net website revealed the following facts regarding the availability of archives of questions and answers on DSpace. The Table 5.7 shows the names of the mailing list and the total number of mails being archived on the website as of December 2014 with the status of availability of online archive.

Table 5.7 Archives of Questions and Answers

DSpace Mailing List	No. of Questions and Answers	Online Archive
DSpace-tech	35784	Yes
DSpace-devel	23841	Yes
DSpace-changelog	6870	Yes
DSpace-general	5862	Yes
DSpace-tickets.	2445	Yes
DSpace-release	626	Yes

Table 5.7 shows that there are a total number of 74678 mails being archived from six mailing forums of DSpace from 2002 to 2014. The data collected from the sourceforge.net website reveals that all mails exchanged through the six mailing lists from the beginning to December 2014 are archived.

5.4.3 Availability of FAQs

Frequently Asked Questions (FAQs) is a common feature of websites of products and services. They are enquiries and answers supposed to be performed regularly in some context on a particular topic. FAQs help to save the time of readers as well as providers. The examination of the official website of DSpace revealed that FAQ is an essential component of the website. DSpace maintains two FAQs; EndUser FAQ and Technical FAQ. End User FAQ deals with questions on the working of DSpace, OSS development model and the ways to contribute to DSpace. Technical FAQ deals with technical questions on customization and code contribution.

5.4.4 Online Documentation

The availability of online documentation is an essential component of any OSS package since the development and growth of OSS are depending on the Internet. DSpace website provides links to various online documentations for the end users. The resources include online user manuals for various versions of DSpace, training materials, publications, results of community surveys, brochures and blogs. All the online documentations are available through search engines. The search for "documentation for DSpace software" was performed on Google, Bing, Yahoo, Ask.com, Aol.com, Wow.com, and it was found that all the six search engines bring the official webpage of DSpace on the first page displaying links to DSpace online manual. Hence it can be concluded that DSpace maintains online documentation and it is easily retrievable through search engines.

5.5 Extent and Scope of Documentation

The fourth major element of WG-OSMM requires examining the quality of a project's documentation. As per WG-OSMM the quality of documentation gives ideas about a project's work process and code quality. The extent and scope of documentation is determined by examining the following sub elements; Clarity in language, Historical milestones of code's development, Organization of documentation, Documentation archives, Contents of documentation and User manual.

5.5.1 Clarity in Language

A mature software's documentation contains instructions for installing, running, and fine-tuning the software written in reasonably clear English. The wiki.duraspace.org website provides documentation for installing DSpace on various platforms. The investigator examined the user manual for five versions of DSpace and found that they contain instructions for installing, running, and fine-tuning the software. It is written in simple and clear English language. The language used for DSpace documentation satisfies the qualities of simplicity and clarity.

5.5.2 Historical Milestones of Code's Development

WG-OSMM wants to check whether the historical milestones in the code's development are available for verification. All original code of DSpace is in the Java programming language. The examination of GitHub web site, the world's largest code repository, revealed that source codes of DSpace are available for downloading. Table 5.8 shows the DSpace version and availability of code in the GitHub.

Table 5.8 Availability of DSpace Code in GitHub

Version	Code in GitHub
DSpace 1.0.x	Yes
DSpace 1.1.x	Yes
DSpace 1.2.x	Yes
DSpace 1.3.x	Yes
DSpace 1.4.x	Yes
DSpace 1.5.x	Yes
DSpace 1.6.x	Yes
DSpace 1.7.x	Yes
DSpace 1.8.x	Yes
DSpace 3.x	Yes
DSpace 4.x	Yes
DSpace 5.x	Yes

Table 5.8 displays the DSpace versions 1.0.x to 5.x and the availability of source code for all versions. The codes can be viewed and downloaded by anyone. The codes are available in zip and tar.gz format to be used on Windows as well as on Linux based systems. The source codes are also available with the wiki.duraspace.org website where it is kept as binary form as well as full source release. These facts reveal the transparency of DSpace project which is a quality of a mature software.

5.5.3 Organization of Documentation

The WG-OSMM requires checking the way OSS project documentation is being organized. The recommended approach is organization of documentation by major version release, the latest being first and so forth. Moreover, the project needs to keep documentation for each release available because many users may still use older releases of the program. The

examination of the official website of DSpace revealed that all DSpace documentation is arranged in descending order. The documentation is available in online format as well in PDF format. The organization of documentation for DSpace satisfies the requirement for a mature OSS.

5.5.4 Documentation Archives

DSpace documentation is available with wiki.duraspace.org webpage which provides facility to browse online as well as to download the full document in PDF format. The sourceforge.net website is working as the archive for DSpace documentation. For DSpace codes, GitHub is the archive. Information about documentations can easily be located through various search engines.

5.5.5 User Manual

The WG-OSMM seeks to examine the presence of comprehensive user manual or reference guide as part of documentation which shall provide complete instructions for installing and configuring software. The official DSpace documentation published along with every stable version of the software is a comprehensive user manual. The investigator examined the contents of DSpace documentation for the last five versions. All are comprehensive and lengthy. Table 5.9 provides details about DSpace user manual.

Table 5.9 Availability of DSpace User Manual

Version	Date	Pages	Format	Installation	Configuration
DSpace 5.x	26-1-2015	779	pdf	2nd module	4th module
DSpace 4.x	28-7-2014	795	pdf	,,	,,
DSpace 3.x	23-6-2014	705	pdf	4th module	6th to 8th module
DSpace 1.8	7-06-2012	644	pdf	,,	,,
DSpace 1.7	28-3-2011	451	pdf	,,	,,

Table 5.9 shows the five user manuals of DSpace belonging to different versions. All manuals were authored by DSpace Developer Team. All manuals are lengthy documents with an average page length of 674. The second and fourth modules of user manual for DSpace 5.x and DSpace 4.x deal with complete instructions for installation and configuration of the software. The manuals for DSpace 1.7 to 3.x (3 versions) include instruction for installation in the 4th module and that for configuration in the 6th to 8th modules. The presence of comprehensive user manual is an essential part of DSpace documentation.

5.6 Quality of Installation Packaging

The fifth major element of maturity is the quality of installation packaging. WG-OSMM inquires the way an open source program is packaged. Some software are available only as source code. Users need to undertake the task of compiling source code into binary, executable form, which is a difficult task. Some projects provide both source and binaries, ready to be used for specific operating systems. A mature OSS is one that is having an installation package and that can be installed easily on many different platforms and

configurations. These elements are examined under the following sub elements; form of packaging and multiplatform compatibility.

5.6.1 Form of Packaging

The examination of the official website of DSpace revealed that DSpace is available both in binary and source code. It can be downloaded as binary release (no Java source code included) or as full source release from the file area in sourceforge.net website. The binary release of DSpace 5.x is having a file size of 580 MB and full source release is 8MB. DSpace source code is also available from GitHub repository.

5.6.2 Multiplatform compatibility

As per WG-OSMM, mature OSS products conform compatibility to multiple operating systems. DSpace is a multiplatform compatible software. It can be installed both on UNIX-like operating systems (Linux, HP-UX, Mac OSX, etc.) and on Windows platform. For this, installation files are kept in zip format for Windows, tar.gz format for UNIX-like systems. The user manuals for different versions of DSpace provide instructions for installation and customization of the software on Windows as well as UNIX-like operating systems. Data available on the official website of DSpace (DSpace user's registry) shows that there are DSpace installations in multiple platforms. Figure 5.12 shows the distribution of DSpace installations among various operating systems.

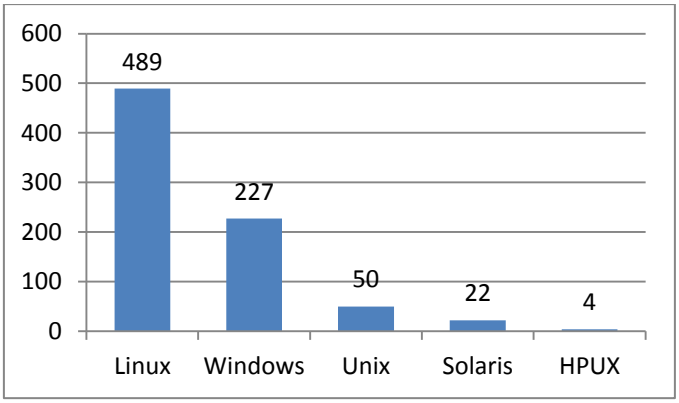

Figure 5.12 Distribution of DSpace on Various Platforms

Figure 5.12 shows that DSpace is installed on Linux, Windows, Unix, Solaris and HP-UX platforms by institutions across the globe. As of December 2014, out of 792 live DSpace installations across the globe (who provided information on the use of operating system), majority (489) are installed on Linux (62%) followed 227 Windows (29%), 50 Unix (6%), 22 Solaris (3%) and 4 HP-UX.

5.7 Momentum (Frequency of Releases)

The sixth major element of maturity is the frequency of software release. According to WG-OSMM the right release schedule depends largely on how stable and mature a project is. Lack of updates can be an indication of abandonment project. A well-managed release cycle indicates the presence of experienced technologists at work. The information on the frequency of release of DSpace software is available on the official website of DSpace, sourceforge.net, and GitHub. As of January 2015, 12 stable versions were released for DSpace. Figure 5.13 displays the stable DSpace versions and time of release.

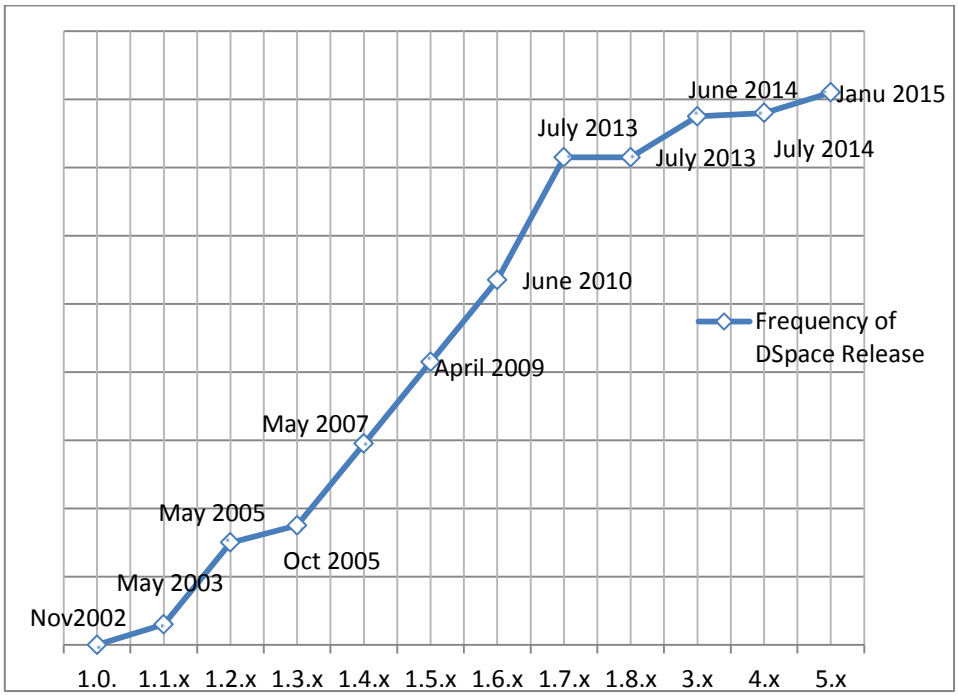

Figure 5.13 Frequency of DSpace Stable Releases

Figure 5.13 shows the release history of DSpace over 14 years. During this period there were 12 stable releases for DSpace with average 0.85 stable releases per year. The first version of DSpace (DSpace 1.0) was released in November 2002 the latest one (DSpace5.x) was released in January 2015. The Figure shows a frequency of six to thirty six months between the two stable versions.

5.7.1 Changes in new Release

WG-OSMM seeks to check out the release history of an OSS and examine if the new releases are mainly significant or more trivial. Ideally, new releases should be put forth only when substantial additions and changes have been made to a program, and not simply with every sprinkling of not-so-important changes.

To study the important changes made in each new release of DSpace, the investigator examined the user manual for the last four versions of DSpace. All documents contain a list of new features, bug fixes and improvements in the release note section. The following are the ten major new features added to the DSpace 5.x version released in January 2015.

1. Option to upgrade to 5.x from any previous version
2. Perform Batch Imports from user interfaces
3. Linked (Open) Data support via a new RDF interface
4. OAI-PMH improvements, including OpenAIRE v3 compliance
5. Enhanced Image and PDF Thumbnails using ImageMagick / Ghostscript
6. File downloads tracking using Google Analytics
7. All DSpace Objects having underlying metadata support
8. ORCID integration
9. Enhanced thumbnail quality
10. Batch import from various bibliographic formats

From the examination of additions to DSpace 5.x, it can be concluded that a DSpace version is updated incorporating several additional features.

5.8 Quality of Code and Design

The seventh major element of maturity is the quality of code and design. WG-OSMM asks to examine the organization of the code, the aspects of modularization, the grouping of modules, the naming convention and the facility to examine the high level structure and the labels being used. These aspects are separately examined under the following sub element;

5.8.1 Organization of DSpace code

Modularity in programming is a software design approach that subdivides a system into independent, interchangeable modules or skids. It is opposed to monolithic approach where the smallest component is the whole. In modular design, numerous small modules are written separately and these become an executable application programmme when compiled together. Modularity is advocated for the sake of changeability, independent development, and comprehensibility. Source code repositories are useful to developers as they help to consolidate systems' source code into a common place. This facilitates them to investigate the program structure and high level relationships exist between the source code components. Since its inception, DSpace software is consolidated in well-known open source repositories like GitHub and sourceforge.net.

The structural view of DSpace code is split mainly into three directory trees viz. DSpace source directory ([dspace-src]), install directory ([dspace]) and web deployment directory ([tomcat/webapps/dspace]). [dspace-src] contain all the source code to build various modules, license files and build property files. Upon installation, [dspace] directory is populated with configuration files, command line tools, libraries, and archive information. The web deployment directory [tomcat/webapps/dspace] contains the XML, JSP, Java classes and libraries necessary to run DSpace application. The aspect of modularity was examined by verifying data from the sourceforge.net and GitHub repository. The following are the conclusions drawn.

The DSpace source code is organized to cohere very strictly to the three-layer architecture. It consists of Application layer, Business logic, and Storage layer. DSpace follows a modular approach for code design. The codes are

independently designed and contributed by several contributors across the globe as branches. These are accepted according to a quality checklist and the DSpace Committers review the branches. Both GitHub and sourceforge.net keeps the codes of all versions of DSpace for examination and verification. The DuraSpace organization follows several checklists to ensure the DSpace code high manageable. The following are the checklist for code contribution;

1. Any changes must be Java 1.6 compliant
2. Contribution should be a "Pull Request" sent to GitHub repository
3. Ensure the code is commented and correctly formatted by IDE's format functions or using tools like Jacobe.
4. Code contribution must adhere to licensing requirements to be included.
5. User interface patches must be internationalized.
6. User interface patches must be XHTML-compliant and have a W3C WAI Conformance Level of "Double-A"
7. User Interface features are encouraged, not necessary, to support both XMLUI and JSPUI interfaces.
8. The patch must come with Documentation.
9. Examples or Use Cases should be submitted to help Committers understand and adequately test the patch prior to applying it to the core code
10. Any new features should be configurable
11. When adding new configuration parameters, name them appropriately.
12. Add appropriate WARN, INFO and DEBUG-level logging.
13. Retain backwards compatibility where possible.
14. No Database schema changes unless absolutely necessary

5.9 Quality of Architecture

The eighth major element of maturity is quality of architecture. WG-OSMM considers the quality of an open source program's architecture as an important measure of the code's maturity. By architecture, WG-OSMM means system components (such as classes in J2EE, PHP systems, modules in Perl), use of design patterns, and naming conventions.

The examination of the user manual for all versions provides data for understanding the architecture of DSpace. J2EE (Java 2 Platform Enterprise Edition) is not required for DSpace. Instead, DSpace uses the Java SE JDK (Standard Edition Java Development Kit) which is a platform to perform Java applications on desktops and servers and embedded environments. Java offers a rich user interface, performance, versatility, portability, and security that a digital library requires. Further, DSpace uses many Java classes and software libraries. The DSpace Web User Interface is the largest and most-used component in the application layer. It is built on Java Servlet and JavaServer Page technology which allow end-users to access DSpace systems over the web via web browsers. The other system components include Apache Maven, Apache Ant, PostgreSQL, Apache Tomcat and Perl. Apache Maven is used to manage a project's build, reporting and documentation from a central piece of information. Apache Ant is a Java build tool. PostgreSQL is the relational database (can be substituted by Oracle), Apache Tomcat is the servlet engine (can be substituted by Jetty, Caucho Resin or equivalent) and Perl, the web programming language.

5.9.1 DSpace Architecture

The DSpace system is organized into three layers, each of which consists of a number of components. The Figure 5.14 shows the system architecture of DSpace.

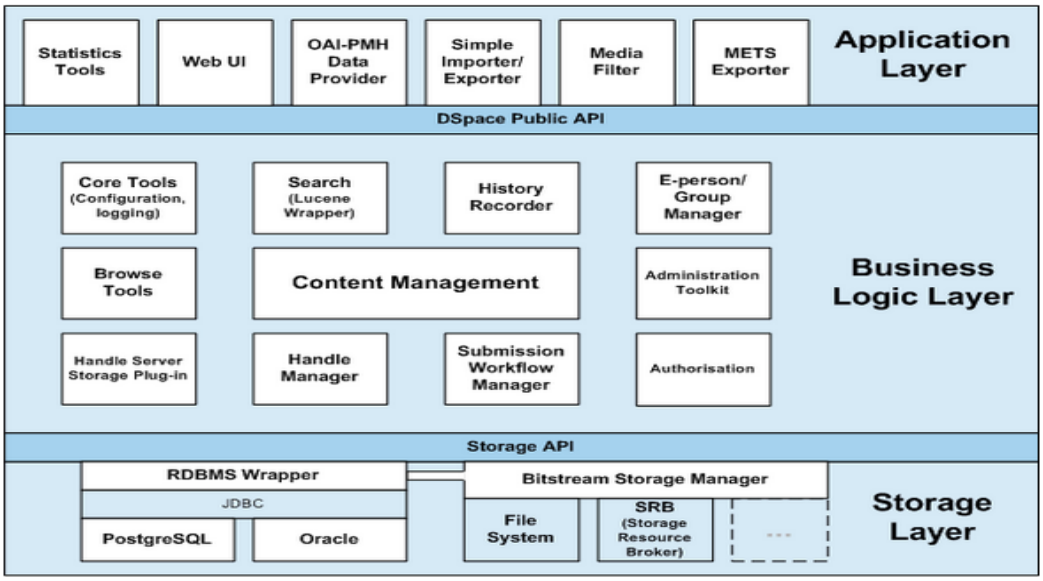

Figure 5.14 System Architecture of DSpace (Courtesy DSpace.org)

The Figure 5.14 shows the basic architecture of DSpace. It consists of application layer, logic layer and storage layer. Each layer performs fundamental functions. The application layer has components that communicate with the world outside of the individual DSpace installation, like the web user interface, Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH) service and Google analytics. The components of business logic layer deals with managing the content of the system, users of the archive (e-people), authorization, and workflow. The storage layer is responsible for physical storage of metadata and content.

5.9.2 Naming Convention in DSpace

In computer programming, naming convention denotes a set of rules to make programs more understandable and easier to read. Naming convention provides information about function of an identifier used in the program which helps better understanding of the code. Moreover, it reduces the effort required to read and understand the source code of software. DSpace follows the Java programming naming convention. The main identifiers used in Java are Packages, Classes, Interfaces, Methods and Variables. All these identifiers have predefined rules for naming and are strictly followed by DSpace committers. The checklist provided by DuraSpace (as detailed in section 5.8.1) ensures this point.

5.10 Testing Practices

The ninth major element of maturity is the testing practice of OSS. WG-OSMM observes that some open source code comes with automated, built-in testing facilities as standard features. The presence of unit tests is a key indicator of good design. The purpose of testing is to detect software failures and fix them possibly during early stages of development. Many OSS follow a test driven development framework in which tests are developed along with the code. Testing approaches are classified as follows:

- Unit tests: To test each method of an object for expected output in various situations.
- Functional tests: To test use-cases originated from the end user
- Integration tests: To test the interaction among the components within the system

- Regression tests: To test the software after a major code change
- Performance tests: To test the software behaviour under heavy load

To address this research element, the investigator depends the various test performed on DSpace by the Texas Digital Library (www.tdl.org) in 2010. They observed that;

1) DSpace is highly integrated and nearly impossible to separate from the database and file systems,

2) Creating unit test for all of DSpace is very time consuming. It is simpler to write a few functional tests that cover a wide set of features over the whole application. To address this problem the Texas Digital Library created a simple framework for adding both integration tests and functional tests. The main concept was to script the install of a test DSpace, with a full configuration and setup. Then they started DSpace in an embedded web server and then run through several scenarios just as a normal user would. The test was performed successfully for two DSpace-based projects.

The TDL test details were posted to DSpace-devel mailing list in April 2010 and there are several responses from the DSpace community leaders. A careful examination of the responses of the DSpace leaders indicate that the concept of testing facility of DSpace was an innovative idea and a good starting point for more automated testing (Unit Tests, Functional Tests, etc.) in general. Data for this element was collected from Tim Donohue who is a technical lead for the DSpace project at DuraSpace from 2009, by email communication. Donohue offered the following information. “Currently, DSpace has unit testing capability built in. They are included in the codebase under the "/src/test/java" subdirectories. They can be executed from the command line via: mvn package -Dmaven.test.skip=false. DSpace also uses a Continuous

Integration (CI) system which automatically runs all Unit Tests on any change in the codebase (this helps to ensure no changes break our tests). DSpace uses Travis CI for those purposes. The details are given at https://travis-ci.org/DSpace/DSpace/

Based on the knowledge provided by one of the DSpace technical experts, it can be concluded that DSpace has unit test facility inbuilt.

5.11 Integration with Other Products

The tenth major element of maturity is the software's integration with other products. WG-OSMM looks for the ability of an OSS to integrate with other products. A set of interdependencies cause programs to work on each other. Altering or ignoring certain dependencies between the previous set of applications and subsystems shall affect the ecological balance of different programmes. A mature software pays attention to the aspect of compatibility. The examination of documentation for various versions of DSpace brings information about the integration of DSpace with other products. DSpace is developed as a generic platform. Hence, its integration with other applications is very essential to make the platform fit for the intended use. The aspects of integration is discussed under the following sub elements; hardware integration, software integration.

5.11.1 Hardware Integration

DSpace can be installed on a modern personal computer, laptop or server class machine. The building of DSpace system for an institution having large number of users requires a server hardware having minimum 3 GB RAM and 20 GB Hard disc storage space to entertain searches, accesses, and downloads.

A high end production system that accommodates more than five lakhs documents requires 8 GB RAM and 1TB storage space.

5.11.2 Software Integration

WG-OSMM looks software integration as an important element of mature software. The changing of software dependencies over different versions of the project shall affect the process of upgrades. It is expected that when a new release of software or dependencies comes out, it will be tested with the other. To examine this element, the investigator depended on the contents of user manual of DSpace. DSpace is written in Java, a general purpose, concurrent and object oriented programming language from the Oracle Corporation. DSpace can therefore be installed on any operating system (UNIX-Linux, HP-UX, Windows, Mac OSX etc.). To examine the integration of DSpace with other products, the investigator examined the prerequisite software of DSpace for last five versions. Table 5.10 shows the list of five DSpace versions and the dependencies associated with them.

Table 5.10 DSpace Dependencies Across Various Versions

DSpace	**OS**	**Development Platform**	**Java Build Tool**	**Java Library**	**Relational Database**	**Servlet Engine**	**Lang-usage**
1.7.1	Multiple	Oracle Java JDK6	Apache Maven 2.2.x	Apache Ant 1.7 or later	Postgre SQL/ Oracle	Tomcat 5.5/6 or Jetty or Caucho Resin or similar	Perl
1.8	,,	,,	,,	Apache Ant 1.8 or later	,,	,,	,,
3.x	,,	JDK6 or 7	Apache Maven 2.2.x or higher	,,	,,	,,	,,
4.x	,,	JDK 7	Apache Maven 3.x	,,	,,	Tomcat 7 or Jetty or Caucho Resin or similar	,,
5.x	,,	JDK 7 or open JDK7	Apache Maven 3.0.5	,,	,,	,,	,,

Table 5.10 shows that DSpace uses the Apache Web server, the Tomcat Servlet engine, and the PostgreSQL relational database. All these tools are coming under open source license. DSpace used the same kind of pre-requisite software over the years across various versions. There is no change for the operating system compatibility of DSpace. There are only changes in the editions of development platform, Java build tool, Java library and Servlet engine. Moreover, when every new version of DSpace releases, there is provision for upgrading from any previous version.

5.12 Supports for Standards

The eleventh major element of maturity is the support for standards. WG-OSMM stresses the need for programs to use standard-based APIs (Application Programme Interface) along with dependencies. API is a set of routines, protocols, and tools for building software applications. Both commercial and OSS use various APIs supplied by a number of different projects or authors. WG-OSMM wants to check if all the APIs work together correctly.

The standards compatibility of DSpace has been documented in the official website, scholarly articles and book chapters produced by the DSpace leaders. The user manuals also provide information on APIs. The API and standards to which DSpace is compatible are listed in 5.12.1- 5.12.7.

5.12.1 Open Archives Initiative (OAI)

Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH) is a protocol developed by the Open Archives Initiative. It is used to harvest (or collect) the metadata descriptions of the records in a digital repository so that interoperability can be achieved among many archives. It provides an application-independent interoperability framework based on metadata harvesting. A harvester or collector is a client application that issues OAI-PMH requests. A harvester is operated by a service provider as a means of collecting metadata from repositories. DSpace has implemented OAI-PMH by using the OCLC OAICat and is exposing Dublin Core metadata for every item in the system.

5.12.2 Open Archives Initiative Object Reuse and Exchange (OAI-ORE)

OAI-ORE is a standard for the description and exchange of aggregations of Web resources. These aggregations, sometimes called compound digital objects, may combine distributed resources with multiple media types including text, images, data, and video. The goal of these standards is to expose the rich content in these aggregations to applications that support authoring, deposit, exchange, visualization, reuse, and preservation. This standard can be used to harvest content (bit streams and metadata) into DSpace from an external OAI-PMH or OAI-ORE server.

5.12.3 Simple Web-service Offering Repository Deposit (SWORD)

SWORD is a repository-standard ingest service using Atom Publishing Protocol. The embedded SWORD Client allows a user to copy an item to a SWORD server. This allows the DSpace installation to deposit items into another SWORD-compliant repository (including another DSpace install). The SWORD interface is configured within the main dspace.cfg file.

5.12.4 Web Distributed Authoring and Versioning (WebDAV)

It is an extension of the Hypertext Transfer Protocol (HTTP) that allows clients to perform remote Web content authoring operations. WebDAV is a proven and robust protocol that was designed for accessing and modifying resources and their metadata over a network.

5.12.5 OpenSearch

OpenSearch is a small set of conventions and documents for describing and using search engines, meaning any service that returns a set of results for a query. It allows publishing of search results in a format suitable for syndication and aggregation. It is commonly visible in modern web sites with search capability. It is used by Wikipedia, Facebook, CNN, etc. DSpace appears as a 'search-engine' to OpenSearch-aware software.

5.12.6 OpenURL

DSpace supports the OpenURL protocol in a rather simple fashion. If an institution has an SFX server, DSpace will display an OpenURL link on every item page, automatically using the Dublin Core metadata. Additionally, DSpace can respond to incoming OpenURLs. Presently it simply passes the information in the OpenURL to the search subsystem. A list of results is then displayed, which usually gives the relevant item (if it is in DSpace) at the top of the list.

5.12.7 Rich Site Summary (RSS)

RSS is a family of standard web feed formats to publish frequently updated information like blog entries, news headlines, audio, video. RSS removes the need for the user to manually check the website for new content. All RSS feed options are available in dspace.cfg. DSpace RSS feeds were designed to offer feeds for recent submissions for the entire repository, communities and collections. RSS feeds enable publishers to syndicate data automatically. A standard XML file format ensures compatibility with many different machines/programs. RSS feeds also help users to receive timely updates from favourite websites or to aggregate data from many sites.

5.12.8 Google Scholar Metadata Mappings

Google Scholar offers search for scholarly literature. The contents archived in DSpace systems are indexed by Google Scholar through automatic crawling by search robots. This helps the DSpace items retrievable easily to large audience. DSpace supports the standards for indexing by Google Scholar since version 1.7 onwards. The DSpace 4.0 contains several enhancements to these standards which were requested by the Google Scholar team. These included providing users (and web indexers) a way to browse content by the date it was added to DSpace ensuring the "dc.date.issued" field is set more accurately.

5.13 Quality of Project Site

The twelfth major element of maturity is the quality of OSS website. Since the success of OSS is very much based on Internet based communications and interactions, an excellent website is essential for any mature project. Different types of websites vary in their organization and display of contents. For a mature OSS, the website shall be educative, simple and easy to navigate. The result of the examination of the official website of DSpace is given below.

5.13.1 DSpace Website

DSpace maintains an official website (www.dspace.org) to communicate information regarding the software. As per Web Archive website, the history of DSpace website goes to 2nd May 2001. However, the purpose of the website at that time was to introduce DSpace as a service of MIT. The first version of DSpace was released on 4th November 2002. The option for downloading DSpace was visible on the website hosted on 14th November 2002. The information and links were organized under six headings that

included “What is DSpace”, “Technology platform”, “MIT Implementation”, “News”, “People” and “Contact Us”. As of January 2015, the official website of DSpace still functions as the central point of access for a variety information and services. The site is highly informative and all-embracing. It has organized information mainly under 5 headings. The Figure 5.15 shows the links that are provided under the five major headings of DSpace website.

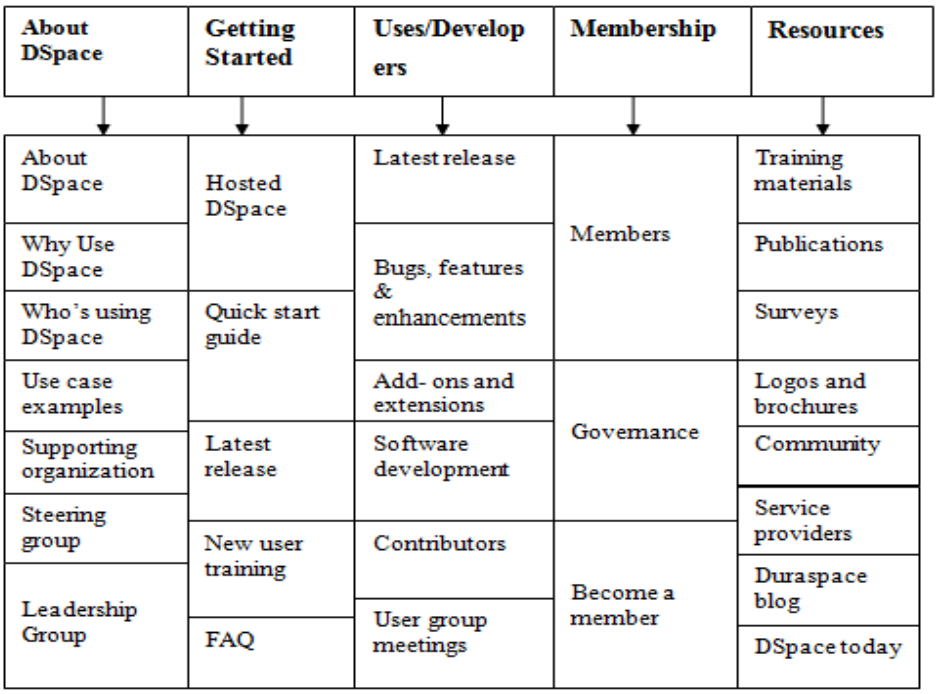

Figure 5.15 Contents of DSpace Official Website

Figure 5.15 shows that there are 34 links to search information on various aspects of DSpace software. The website also provides links *to take a tour*, *download current release*, *who's using DSpace*, *Hosted DSpace* and *DSpace news*. Apart from this, the website provides site map, facility to search the site content and link to contact the DuraSpace, the non-profit organizing that provides leadership to DSpace project. The site contents are licensed under Creative Commons.

5.13.2 Site Design

The examination of the contents and links available on the official website of DSpace revealed that it conforms to the degree of brevity and clarity expected of a mature OSS project by the WG-OSMM. The site is organized in a simple way to facilitate easy navigation by any visitor.

5.13.3 Educative

As per WG-OSMM a great site can make it easy for everyone to educate himself and find what he needs. The study has examined the details provided in the 34 links that appear on the home page of DSpace website. The examination revealed that the DSpace website is intended for educating the user on all aspects of the software. It provides knowledge covering basics to highly technical aspects of the software. A user of the software is able to view, read and download text and videos containing essential information on the software.

The link to training materials leads to a webpage that provides a variety of materials and resources developed by the DSpace user community which can be used as self-guided tutorials. This section gives information about DSpace Course that is intended for DSpace administrators or developers and includes 20 different modules on various topics. The contents were created by experts in the field who work as committers, developers and trainers of DSpace. This page also provides links to *DSpace Resources wiki* which contains latest technical information about DSpace platform. The wiki page holds documentation in downloadable format for all the versions of the DSpace.

5.14 License Type

The thirteenth major element of maturity is the type of OSS license. WG-OSMM considers direct access to the programmes' source code as the great attractions of open source. This is a vital form of insurance. Open source software employ a variety of licenses. In some cases, there might be no restriction on the use of the software for developing a new system that is based on certain software. But when it comes to distribute those applications and the underlying program in a commercial application, or to make the applications available through a public web site, users must pay a fee perhaps determined by how many servers are executing the software. Some licenses give the user freedom not only to work with the program to develop new applications, but also to distribute it at no charge. Still another level of license permits users to modify the software themselves, perhaps requiring that they contribute their improvements back to the open source project. The WG-OSMM requires checking the type of license.

Open source licenses are licenses that comply with the Open Source Definition of the Open Source Initiative (OSI). They allow software to be freely used, modified, and shared. DSpace is shared and distributed under Berkeley Software Distribution (BSD) license. DSpace attaches other open source tools to package with the DSpace applications and all these are freely available under an open-source license. However they don't carry the same license. The BSD license is permissive license which allows the source code to be copied or used without requesting the authors' permission and without cost. The distribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name DuraSpace nor the name of the DSpace Foundation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission. The following are the acceptable licensing models of DSpace.
- Apache License 2.0
- BSD
- Common Development and Distribution License (CDDL)
- Common Public License (CPL)
- GNU Library or "Lesser" General Public License (LGPL)
- MIT / X11 License
- Mozilla Public License

The following are the unacceptable licensing models of DSpace

- GNU General Public License (GPL)
- GNU Affero General Public License (AGPL)
- European Union Public Licence (EUPL)
- Any license which strictly forbids "sublicensing" as detailed at http://choosealicense.com/licenses/
- Any license which limits commercial use/redistribution of binary code

The DSpace license is aimed at commercial entities and service providers to be able to customize the entire codebase and redistribute/repackage/sell it in a binary form. GPL licenses is preventing this option and DSpace has included it under unacceptable license category

5.15 Potential for Commercial Conflicts

The fourteenth major element of maturity is the potential for commercial conflicts. WG-OSMM cautions the possibility of dubious legal status of software of all types in which code might infringe or simply appear to infringe on the intellectual property of a commercial company. This puts developers as well as users at risk of potential legal action by those commercial suppliers. Though this sort of conflict is a rare occurrence, it is desirable that open source projects focus mostly on integration with other open source projects.

The DSpace uses standard software as prerequisites from Java to Perl. The list below describes the third-party components and tools needed to run a DSpace server.

- Unix-Like Operating System or Microsoft Windows. Many distributions of Linux/Unix come with some of the dependencies pre-installed or easily installed via updates.
- Oracle Java JDK 7 or OpenJDK 7- OpenJDK (Open Java Development Kit) is a free and open source implementation of the Java programming language licensed under the GNU General Public License (GNU GPL) with a linking exception.
- Apache Maven- it is an open source project.
- PostgreSQL- a relational database. It is also an open source project
- Apache and Tomcat- both from the Apache Foundation and belong to open source category.
- Jena (an RDF toolkit from HP Labs
- OAICat from OCLC

It is found that all leveraged components and libraries attached to DSpace are also OSS and there is no possibility of commercial conflict with any software within DSpace.

5.16 Corporate Commitment

The fifteenth major element of maturity is the corporate commitment. Several open source projects, such as the Linux operating system and Apache Web Server, have enjoyed tremendous support from large, established computer companies, including IBM, Sun, HP, and Dell. IBM, in particular, has helped the Apache server effort with people and valuable source code. The support for a project from reputed corporations adds values to maturity. The supports received by DSpace from various corporate sectors are listed below.

5.16.1 Support for DSpace

DSpace is a joint project of Massachusetts Institute of Technology (MIT) Libraries and Hewlett Packard (HP) Lab begun in 2002. MIT is a well-known world-class university in the United States. The HP Lab is the leading multinational company headquartered in California, United States. The company is associated with hardware, software and services segment. It has 45 years of experience in the field. Moreover DSpace has received funding from Andrew Mellon Foundation, USA. Currently the DuraSpace looks after the project under the leadership of highly qualified professionals from across the globe. Over two-thirds of DSpace revenue are coming from sponsorship and hosted solutions. DSpace has several kinds of membership that include universities, university libraries and digital libraries. The corporate support for DSpace is sufficient to achieve the status of a mature OSS.

5.16.2 Third Party Support for DSpace

DSpace has many registered service providers across the globe who work with DuraSpace organisation. The official website of DSpace has given a list of nine firms who provide third party support for DSpace. They include Agronet from S.Korea, Arvo consulters from Spain, @mire from Belgium and USA, Cineca-from Italy, DSqure from India, Ibai from Spain, LongSight from USA, ProviderIT from Brazil and Open Repository from United Kingdom.

5.17 WG-OSMM – Maturity Scoring

The last part of the WG-OSMM attempts to quantify the maturity of an OSS in product criteria, use criteria and integration criteria. The details of these criteria are given below.

Product Criteria - Product criteria are specifics about the product itself. Since OSS products are often under rapid development, with major advances made in a few weeks to a few months, the model lists momentum as a criterion to offset the age criterion. Product criteria help to spot products that aren't mature enough today but are worthy of keeping an eye on.

Use Criteria - Use criteria are specifics about what it takes to use the product from day to day, from the effort of initial installation and configuration to the work required for daily upkeep and support mechanisms available to help in tailoring the product to an institution’s needs and fixing defects encountered.

Integration Criteria - Integration criteria are specifics about what it takes to make the product work in the institution’s environment. For each criterion the Model assigns a score of 1, 2, or 3: Table 5.11 shows the score and description of the score.

Table 5.11 WGOSMM Scoring Description

Score	Status	Description
1	Immature product	The product is lacking in several critical areas. It is not fit for adoption for a production use.
2	Reasonably mature	The product has sufficient quantity of features with a bright future. But it is weak on some areas.
3	Very mature	The product has a long and stable history, a broad and vibrant user community.

Table 5.12, 5.13 and 5.14 show the WGOSMM scoring chart under product criteria, use criteria and integration criteria.

Table 5.12 WG-OSMM Score Chart for Product Criteria

Woods and Guliani's Open Source Maturity Model-2005				
Maturity Criteria	Score=1	Score=2	Score=3	Criteria Description
Product Criteria				
Age	6 months	6-months -2 years	2 years	OSS that are just getting underway are risky
Multiple Supported Platforms	One Platform	Many related platforms	Multiple heterogeneous platforms	
Momentum	No release in last 6 months	two releases in past year	Regular releases	This is key to helping separate vital products from ones that are withering.
Popularity	Unknown product	Viable alternative	Category leader	Popular OSS products are well tested and therefore more mature.
Design quality	Monolithic application	Multiple components	Well-defined API	This is key in determining the effort required to extend and adapt the product for use.

Table 5.13 WG-OSMM Score Chart for Use Criteria

Woods and Guliani's Open Source Maturity Model-2005				
Maturity Criteria	Score=1	Score=2	Score=3	Criteria Description
Use Criteria				
Setup cost	Poorly documented install process; poor documentation; help available from developers	Well-documented install process; reasonable documentation; help available from developers; help available in support forums	Well-documented install process; install wizards/scripts available; reasonable documentation; help available from developers; help available in support forums; third-party install services	Most products should require a setup effort of hours or days, not weeks or months.
Usage cost	Poor or nonexistent documentation; help available only through direct contact with developers	User manuals available; help available in support forums	Third-party training services available	This criterion is often overlooked when evaluating a product.
End-user support	No forums or mailing lists	Some forums or mailing lists	Well-run forums and mailing lists, with archives and search; third-party support options	User community (forums, mailing lists) and third-party support are vital to a product's success.

Table 5.14 WG-OSMM Score Chart for Integration Criteria

Woods and Guliani's Open Source Maturity Model-2005				
Maturity Criteria	Score=1	Score=2	Score=3	Criteria Description
Integration Criteria				
Modularity	Monolithic structure; possible but hard to extend	Multiple modules; possible to extend	Multiple modules, well-defined API; possible and easy to extend	
Collaboration with other products	Unknown	Known cases of integration	Lots of integration documented	
Standards compliance	Unknown or proprietary	Outdated	Current industry standards	
Developer support	No forums or mailing lists	Some forums or mailing lists	Well-run forums and mailing lists with archives and search; third-party support options	

5.18 Measuring Maturity of DSpace

The last part of analysis attempts to measure the maturity of DSpace software against the quantitate values set by WG-OSMM. The scoring is based on the examination of fifteen major elements of maturity and various sub-elements coming under them. Table 5.15, 5.16. and 5.17 show the WG-OSMM maturity chart organized under product criteria, use criteria and integration criteria respectively. The first column contains criteria, the second, third and

fourth columns cover the status of a product, the fifth column provides explanation corresponding to DSpace against the criteria. The sixth column shows the score obtained by DSpace against each criterion. The value of score and status of software is described below.

I = Immature (Score1),
RM = Reasonably Mature (Score 2), and
VM =Very Mature (Score 3).

Table 5.15 WG-OSMM Score for DSpace Under Product Criteria

Criteria	I	RM	VM	DSpace	Score
Age	6 months	6-months -2 years	2 years	More than 2 years	3
Multiple Supported Platforms	One Platform	Many related platforms	Multiple heterogen-eous platforms	Run on Unix-like and Windows based platforms	3
Momentum	No release in last 6 months	two releases in past year	Regular releases	12 stable release in 14 years	3
Popularity	Unknown product	Viable alternative	Category leader	Category leader	3
Design quality	Monolith ic	Multiple components	Well-defined API	Well defined API	3

Table 5.15 contains the score obtained by DSpace for the criteria of Age, Multiple Supported Platforms, Momentum, Popularity and design quality. DSpace was developed in 2002. As of November 2014, it has crossed twelve years. The multiple supported platforms compatibility of DSpace has been discussed in section 5.6.2. The momentum of DSpace was explained in section 5.7. The popularity criterion has been given in section 4.6. DSpace is the category leader of OSS for DLs. Design quality criterion of DSpace was

given in section 5.8.1.DSpace received the highest score for all the product criteria of WG-OSMM. Table 5.16 shows the score for Use Criteria.

Table 5.16 WG-OSMM Score for DSpace Under Use criteria

Criteria	I	RM	VM	DSpace	Score
Setup cost	Poorly document-ed	Well-documented install process; reasonable documentation; help from developers; help from forums	Well-document ed, Support from forums, support from third party	Well-documented, Support from forums, support from third party	3
Usage cost	Poor or document ation; only direct contact with developers	User manuals available; help available in support forums	Third-party training services available	User manuals available; support forums Third-party training services available	3
End-user support	No forums or mailing lists	Some forums or mailing lists	Well-run forums and mailing lists, with archives and search; third-party support options	Six mailing lists, mail archive, support from DuraSpace	3

Table 5.16 contains the score obtained by DSpace for the criteria of Setup cost, Usage cost and End-user support. The aspects of documentation and third party support of DSpace were discussed in section 5.5 and 5.16.2 respectively.

DSpace has received the highest score for all criteria under the use criteria score of WG-OSMM.

Table 5.17 shows the score for DSpace under Integration Criteria.

Table 5.17 WG-OSMM Score for DSpace Under Integration Criteria

Criteria	**I**	**RM**	**VM**	**DSpace**	**Score**
Modularity	Monoli-thic structure ; possible but hard to extend	Multiple modules; possible to extend	Multiple modules, well-defined API; possible and easy to extend	Modular design, multiple modules, well defined API, easy to extend	3
Collaboratio n with other products	Unkno wn	Known cases of integration	Lots of integration document ed	Integration with Standard products	3
Standard Compliance	Unkno wn or propriet ary	Outdated	Current industry standard	Latest industry standard	3
Developer support	No forums or mailing lists	Some forums or mailing lists	Well-run forums and mailing lists with archives and search; third-party support options	Very active forums with archive to search. Support from DuraSpace and others	3

Table 5.17 contains the score obtained by DSpace for the criteria of Modularity, Collaboration with other products, Standard Compliance, Developer support. The modularity of DSpace has been given in 5.8.1. The ability of DSpace to integrate with other products and the Standard Compliance of DSpace were

discussed in section 5.11 and 5.12 respectively. The details of DSpace mailing forum have been given in 5.4 and 5.5. DSpace received the highest score for all criteria under the use criteria score of WG-OSMM.

The measuring of the maturity of DSpace open source software using the WG-OSMM is presented in Table 5.15, 5.16 and 5.17. The descriptive section examined fifteen elements of maturity and the sub -elements under it. The scoring chart of WG-OSMM contains twelve criteria of maturity under product criteria, use criteria, and integration criteria. DSpace has achieved the highest score for each criterion under product criteria, use criteria and integration criteria. Based on the examination of the descriptive elements and the scoring criteria, the investigator has found that DSpace is a mature OSS. The feasibility of DSpace for a DL has been examined by studies attempted previously. This study adds extensions to the previous studies. The outcome of the study is that while DSpace is a suitable OSS for DL, it is a mature software also. The findings of the study are summarized in the next chapter.

References

1. Bell, J. T., Ng, S., & Lambros, J.T. (2003). J2EETMopen source toolkit: Building an enterprise platform with open source tools. New Jersey: John Wiley & Sons.

2. Hodge, G. M. (2000). Best practices for digital archiving: An information life cycle approach. D-Lib magazine, 6(1). Retrieved From http://www.dlib.org/dlib/january00/01hodge.html (accessed May 12, 2014)

3. Shihab, E., Bettenburg, N., Adams, B., & Hassan, A. E. (2010). On the central role of mailing lists in open source projects: An exploratory study. In Nakakoji, K., Murakami, Y., & McCready, E. (Eds.). New Frontiers in Artificial Intelligence: JSAI-isAI 2009 Workshops, LENLS, JURISIN, KCSD, LLLL, Tokyo, Japan, November 19-20, 2009, Revised Selected Papers (Vol. 6284) (pp. 91-103). Heidelberg: Springer.

Chapter 6 Summary of Findings and Suggestions

6.1 Introduction

The assessment of maturity of software is an important area in the general software sector. The field of OSS also applies various models to measure software maturity. However, measuring maturity of OSS being used for several applications in libraries is an area left with no research so far. This study has attempted to fill the research gap. Measuring maturity of software contributes knowledge on its sustainability over the long term. Maturity of software is one of the factors that positively influence adoption. The investigator measured the maturity of DSpace software using Woods and Guliani's Open Source Maturity Model-2005. This chapter gives the

summary of the major findings of the study, areas of applications of the result of the study and suggestions for further research.

6.2 Findings

The major findings of the study are organized under two broader categories: DSpace software environment and DSpace software features.

6.2.1 DSpace Software Environment

1. DSpace is the category leader among the OSS for DLs.
2. DSpace is adopted in 117 countries in the world representing six continents.
3. The institutional and individual leadership of DSpace is identifiable.
4. DSpace has a strong leadership base and they are composed of experts from the field of computer science and library and information science.
5. Institutionally, DSpace was developed by MIT Libraries and Hewlett Packard Labs in 2002.
6. The Andrew W. Mellon Foundation supported the DSpace project.
7. The DuraSpace organization is providing leadership to DSpace
8. The DSpace team members actively participated in the many forums and mailing lists of DSpace and they provided significant contribution for the popularization of the software by attending conferences and presenting papers.
9. DSpace team members were positive to user queries.
10. DSpace software maintained a professional culture throughout its various stages of development.

11. The relationship between DSpace developers and users is found to be healthy.
12. DSpace Community involves Steering Group, Leadership Group and Project Members, Committers, Emeritus Committers, Contributors, Community Advisory Team, Ambassadors, Members, Sponsors and Users.
13. Several Universities and institutions support DSpace project.
14. DSpace has a regular release history having 12 releases over fourteen years beginning from November 2002 to January 2015.
15. DSpace project maintains a website that is highly informative and all-embracing. The home page provides links to 34 categories of contents. The site is simple and confirms to the degree of brevity and clarity expected of a mature software. The site contents are licensed under Creative Commons.
16. The release of easy to use installation packages/ documentation for the users is very part of DSpace project.
17. DSpace reduces the potential for commercial conflicts as all the prerequisite software for installation of DSpace belong to OSS.
18. DSpace has a total number of six mailing lists namely DSpace-tech, DSpace-general, DSpace-devel, DSpace-changelog, DSpace-release and DSpace-tickets.
19. DSpace maintains separate mailing list for users and developers
20. DSpace retains documentation for all versions of the software.
21. The total downloads for DSpace exceed 3 lakhs from 2002 to December 2014 with an average of 27000 downloads per year, 2200 per month and 75 per day.
22. DSpace mailing list is archived from 2002 onwards.
23. DSpace maintains a structured FAQ
24. DSpace keeps online documentation for all versions of DSpace.

25. DSpace documentations are organized by version.
26. DSpace user manuals contain sections that describe the installation of the software.
27. The understating of the historical milestones of DSpace code development is possible as all the source codes of all versions of DSpace are kept with GitHub code repository.
28. Third party support options are available for DSpace.

6.2.2 DSpace Software Features

1. Majority of the modules of DSpace is developed in Java programming language.
2. DSpace is available both in binary and source code forms.
3. DSpace is multiplatform compatible. It can be installed both on UNIX-like operating systems (Linux, HP/UX, Mac OSX, etc.) and on Windows platform.
4. DSpace incorporates new features and tools in every new release. And new release of DSpace is brought when there are significant additions to the existing features.
5. DSpace follows modularity in code design.
6. The basic architecture of DSpace is composed of three layers: application layer, business logic layer and storage layer.
7. The application layer contains components that communicate with the world outside of the individual DSpace system.
8. The business logic layer handles the content of the archive, users of the archive (e-people), authorization, and workflow.
9. The storage layer is responsible for physical storage of metadata and content.

10. DSpace maintains checklists for accepting code from code contributors to ensure quality.
11. DSpace follows a naming convention as per Java programming.
12. DSpace facilitates various testing practices.
13. DSpace maintains integration with hardware and software having current industry standards.
14. DSpace ensures the contents being easily and effectively indexed by major search engines.
15. DSpace conforms to the indexing guidelines recommended by Google Scholar.
16. DSpace uses the same prerequisite software over the years.
17. DSpace provides facility to upgrade from any previous version to the new version.
18. DSpace supports a number of standards and protocols for the exchange, preservation and use of digital data. It includes Open Archives Initiative (OAI), Metadata Harvesting (OAI-PMH), Open Archives Initiative Object Reuse and Exchange (OAI-ORE), Simple Web-service Offering Repository Deposit (SWORD), Web Distributed Authoring and Versioning (WebDAV), OpenSearch, OpenURL, Rich Site Summary (RSS).
19. DSpace is shared and distributed under Berkeley Software Distribution (BSD) license.
20. DSpace uses standard software as prerequisites from Java to Perl.
21. DSpace satisfies the requirement of all the fifteen elements of Woods and Guliani's Open Source Maturity Model-2005.
22. DSpace achieved the highest score for elements under product criteria, use criteria and integration criteria of Woods and Guliani's Open Source Maturity Model-2005.

23. Based on the examination of the descriptive elements and the scoring elements of Woods and Guliani's Open Source Maturity Model-2005, DSpace is a mature OSS.

6.3 Applications of the result of the Study

The result of the study can be applied in the selection of DSpace for DLs. The study can be used to gain in-depth knowledge on DSpace. The result of the study can also be used to develop understanding on the application of maturity measures to OSS in libraries.

6.4 Suggestions for further research

1. This study can be conducted on other open source software used in libraries such as Koha, EPrints, Greenstone, NewGenLib etc.

2. The maturity of OSS being used in libraries can be made by using other maturity measurement models

3. Further exploration on the factors influencing the adoption of open source software in libraries with special reference to the perception of maturity of software.

Bibliography

Adewumi, A., Omoregbe, N., Misra, S., & Fernandez, L. (2013,). Quantitative quality model for evaluating open source web applications: Case study of repository software. In IEEE 16th International Conference on Computational Science and Engineering (CSE), 2013 (pp. 1207-1213). Retrieved from http://ieeexplore.ieee.org/stamp/ stamp.jsp?tp=&arnumber=6755361 (accessed March 5, 2014).

Ahmed, S. S., & Al-Baridi, S. (2012). An overview of institutional repository developments in the Arabian Gulf region. OCLC Systems & Services, 28(2), 79-89.

Aichernig, B. K. (2006,) Qualification of open source software for e-government. e-Maco report 2008, Version 1.0, Retrieved from http://www.emacao.gov.mo/documents/20/report28.pdf (accessed June 5, 2014).

Akbari, M., & Peikar, S. R. H. (2014, September). Evaluation of free/open source software using OSMM model case study: WEBGIS and spatial database. Advances in Computer Science: an International Journal3 (5), 34-43 Retrieved from http://www.acsij.org/documents/v3i5/ACSIJ-2014-3-5-521.pdf (accessed October 5, 2014).

Alayon, S. B., Nemiz, E. S., Superio, D. L., de la Peña, J. G., & Pacino, L. G. (2013). The development of an institutional repository at the Aquaculture Department of the Southeast Asian Fisheries Development Center,

Philippines. Program: Electronic Library and Information Systems, 47(4), 384-398

Al Zeheimi, S., Zeki, A. M., Razi, M., Jalaldeen, M., Zain, S. M., & Abubakar, A. (2014). Perceptions of library and information science community towards open source software adoption in libraries of Oman. In Information and Communication Technology for The Muslim World (ICT4M), 2014 The 5th International Conference on (pp. 1-6). IEEE. Retrieved from http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7020591 (accessed June 5, 2014).

Amollo, B. A. (2013). Feasibility of adaptation of open source ILS for libraries in Kenya: A practical evaluation. The Electronic Library, 31(5), 608-634.

Anandan, C. (2009). Product management. New Delhi: Tata McGraw-Hill Education.

Applewhite, A. (2003). Should governments go open source? IEEE Software, 20(4), 88-91.

Arms, W. Y. (1995, July). Key concepts in the architecture of the digital library. D-lib Magazine, 1(1). Retrieved from http://www.dlib.org/dlib/July95/07arms.html (accessed June 8, 2014).

Ayala, C. P., Cruzes, D. S., Hauge, O., & Conradi, R. (2011). Five facts on the adoption of open source software. IEEE Software, 28(2), 95-99.

Barve, S., & Dahibhate, N. B. (2012). Open source software for library services. DESIDOC Journal of Library & Information Technology, 32(5), 401-408.

Barwick, J. (2007). Building an institutional repository at Loughborough University: Some experiences. Program: Electronic Library and Information Systems, 41(2), 113-123.

Bauer, M (2000), Classical content analysis: A review. In M.W. Bauer & G. Gaskell (Eds.), Qualitative researching with text, image and sound: a practical handbook. London: Sage.

Bell, J. T., Ng, S., & Lambros, J.T. (2003). J2EETMopen source toolkit: Building an enterprise platform with open source tools. New Jersey: John Wiley & Sons.

Berelson, B. (1952). Content analysis in communication research. New York: Free Press.

Bisson, C., West, J., & Eby, R. (2007, May/June). Open source software for libraries. Library Technology Report. American Library Association Tech Source. Retrieved from http://maisonbisson.com/oss4lib/ (accessed June 5, 2014).

Boulanger, A. (2005). Open-source versus proprietary software: Is one more reliable and secure than the other?. IBM Systems Journal, 44(2), 239-248.

Breeding, M. (2002). The open source ILS: Still only a distant possibility. Information Technology and Libraries, 21(1), 16.

Breeding, M. (2009). The viability of open source ILS. Bulletin of the American Society for Information Science and Technology, 35(2), 20-25.

Bretthauer, D. (2001, December 12). Open source software: A history. Retrieved from http://digitalcommons.uconn.edu/cgi/viewcontent.cgi?article=1009&context=libr_pubs (accessed June 6, 2014).

Brewer, E., Demmer, M., Du, B., Ho, M., Kam, M., Nedevschi, S., & Fall, K. (2005). The case for technology in developing regions. Computer, 38(6), 25-38.

Buchanan, K. & Krasnoff, B. (2005). Can open source software save school libraries time and money? Knowledge Quest, 33(3), 32-34.

Cabinet Office, UK (2012, April 27). All about open source: An introduction to open source software for Government IT, version 2.0. Retrieved from https://www.gov.uk/ government/ uploads/system/uploads/attachment_data/file/78959/All_About_Open_Source_v2_0.pdf (accessed June 6, 2014).

Calmorin, L. P., & Calmorin, M. A. (2007). Research methods and thesis writing. Manila: Rex Book Store.

Candela, L., Castelli, D., & Pagano, P. (2011). History, evolution and impact of digital libraries. In Iglezakis, I., Synodinou, T., & Kapidakis,S. (Eds.). E-publishing and digital libraries: Legal and organizational issues, (pp.1-30). Hershey: IGI Global.

Cervone, H. F. (2006). Some considerations when selecting digital library software. OCLC Systems & Services: International Digital Library Perspectives, 22(2), 107-110.

Chen, C. M., Chen, Y. T., Hong, C. M., Liao, C. W., & Huang, C. M. (2012). Developing a Taiwan library history digital library with reader knowledge archiving and sharing mechanisms based on the DSpace platform. The Electronic Library, 30(3), 426-442.

Cherukodan, S., Santhosh Kumar, G., & Humayoon Kabir, S. (2013). Using open source software for digital libraries: A case study of CUSAT. The Electronic Library, 31(2), 217-225.

Choi, N. (2014). The application profiles and development characteristics of library open source software projects. Library Hi Tech, 32(2), 260-275.

Chudnov, D. (1999). Open source software: The future of library systems? Library Journal, 124 (13), 40-43.

Clark, B. K. (1997). The effects of software process maturity on software development effort. (Doctoral dissertation) Retrieved from

http://sunset.usc.edu/~bkclark/Research/ PMAT990406.pdf.(accessed May 2, 2005)

Classen, H. W. (2007). A practical guide to software licensing for licensees and licensors: Analyses and model forms (2nded.). Chicago, IL: American Bar Association.

Comino, S. & Manenti, F. M. (2005). Government policies supporting open source software for the mass market. Review of Industrial Organization, 26(2), 217-240.

Creswell, J.W. (1998). Qualitative inquiry and research design: Choosing among five traditions. Thousand Oaks, CA: Sage.

Damsgaard, J. & Karlsbjerg, J. (2010). Seven principles for selecting software packages. Communications of the ACM, 53(8), 63-71.

De', Rahul. (2009) Economic impact of free and open source software - A Study in India. Retrieved from

http://www.iimb.ernet.in/~rahulde/RD_FOSSRep2009.pdf (accessed June 7, 2014).

Devakos, R., & Toth-Waddell, A. (2008). Ontario government documents repository D-space pilot project. OCLC Systems & Services, 24(1), 40-47.

Digital Library Federation (2011, January 11). A working definition of digital library. Retrieved from http://old.diglib.org/about/dldefinition.htm (accessed June 5, 2014).

Doctor, G., & Ramachandran, S. (2008). DSpace@ IBSA: Knowledge sharing in a management institute. VINE, 38(1), 42-52.

DoD (2009, October 16). Clarifying guidance regarding open source software (OSS). Retrieved from http://dodcio.defense.gov/Portals/0/Documents/FOSS/2009OSS.pdf (accessed June 7, 2014).

Ebert, C. (2008). Open source software in industry. IEEE Software, 25(3), 52-53.

Elliott, M. S., & Scacchi, W. (2008). Mobilization of software developers: The free software movement. Information Technology & People, 21(1), 4-33.

Ellis, J., & Van Belle, J. P. (2009, June). Open source software adoption by South African MSEs: Barriers and enablers. In Proceedings of the 2009 Annual Conference of the Southern African Computer Lecturers' Association (pp. 41-49). ACM.

Fageria, N. K. (1992). Maximizing crop yields. New York: CRC Press.

Fitzgerald, B. (2006). The transformation of open source software. MIS Quarterly, 30 (3) 587–598.

Fitzgerald, B. (2011). Open source software adoption: Anatomy of success and failure. International Journal of Open Source Software & Processes, 1 (1), 1-23.

Forge, S. (2006). The rain forest and the rock garden: The economic impacts of open source software. Info, 8(3), 12-31.

Ghosh, M. (2009). E-theses and Indian academia: A case study of nine ETD digital libraries and formulation of policies for a national service. The International Information & Library Review, 41(1), 21-33.

Ghosh, R.A. (2006, November 20). Economic impact of open source software on innovation and the competitiveness of the Information and Communication Technologies (ICT) sector in the EU. Retrieved from http://ec.europa.eu/ enterprise/sectors/ict/files/2006-11-20-flossimpact_en.pdf t (accessed June 7, 2014).

Golden, B. (2005). Succeeding with open source. Boston: Addison-Wesley Professional.

Griffin, S. M. (1998, July/August). NSF/DARPA/NASA digital libraries initiatives: A program manager's perspective. D-Lib Magazine, 4 (7/8). Retrieved from http://www.dlib.org/dlib/july98/07griffin.html (accessed June 7, 2014).

Gurusamy, K. (2011, February 21). Open source software adoption in the Australian public sector (Doctoral dissertation). Retrieved from http://www.canberra.edu.au/researchrepository/file/12add23b-cbf0-eab2-1906-90d50c1017a2/1/full_text.pdf (accessed June 5, 2014).

Han, Y. (2004). Digital content management: The search for a content management system. Library Hi Tech, 22(4), 355-365.

Hasan, N. (2009, October). Issues and challenges in open source software environment with special reference to India. In Proceedings of International Conference on Academic Libraries (ICAL-2009), University of Delhi, Delhi. Retrieved from http://crl. du. ac. in/ical09/papers/index_files/ical-43_144_ 317_1_RV.pdf (accessed June 9, 2014).

Helly, J. J., Elvins, T. T., Sutton, D., & Martinez, D. (1999). A method for interoperable digital libraries and data repositories. Future Generation Computer Systems, 16(1), 21-28.

Hendrickson, M., OReilly, T. & Magoulas, R. (2012). Economic impact of open source on small business: A case study. Sebastopol, CA: O'Reilly Media.

Herring, S. C. (2010). Web content analysis: Expanding the paradigm. In Jeremy Hunsinger, Lisbeth Klastrup, Matthew M. Allen (Eds.) International handbook of internet research (pp. 233-249). Netherlands: Springer.

Hodge, G. M. (2000). Best practices for digital archiving: An information life cycle approach. D-Lib magazine, 6(1). Retrieved From http://www.dlib.org/dlib/ january00/01 hodge.html (accessed May 12, 2014)

Hoe-LianGoh, D., Chua, A., AnqiKhoo, D., Boon-HuiKhoo, E., Bok-Tong Mak, E., & Wen-Min Ng, M. (2006). A checklist for evaluating open source digital library software. Online Information Review, 30(4), 360-379.

Holck, J., Larsen, M.H.& Pedersen, M.K. (2005). Managerial and technical barriers to the adoption of open source software. In X. Franch & D. Port (Eds.), COTS-based software systems: 4th International Conference, ICCBSS 2005, Bilbao, Spain, February 2005 (pp. 289–300). Heidelberg: Springer.

Hulse, B., Cheverie, J. F., & Dygert, C. T. (2007). ALADIN research commons: A consortial institutional repository. OCLC Systems &Services, 23(2), 158-169.

Jose, S. (2007, February 14). Adoption of open source digital library software packages: A survey. Retrieved fromhttp://eprints.rclis.org/8976/1/Sanjojose.pdf (accessed March 12, 2012).

Isitan, K. (2011, May). An approach to establish a software reliability model for different free and open source software development paradigms. Retrieved from https://tampub.uta.fi/bitstream/handle/10024/82597/gradu05074.pdf?sequence=1 (accessed June 8, 2014).

Jalote, P. (2008). An integrated approach to software engineering. (Ed.3.), New Delhi: Narosa.

Joki, S. M. E. (2007). PEPIA: A Norwegian collaborative effort for institutional repositories. OCLC Systems & Services, 23(2), 204-209.

Jones, P., Day, M., & Ball, A. (2009). Institutional repositories should be built on open source software. Bulletin of the American Society for Information Science and Technology, 35(4), 22-26.

Jones, R. D. (2004). DSpace vs. ETD-db: Choosing software to manage electronic theses and dissertations. Retrieved from: http://www.ariadne.ac.uk/issue38/jones (accessed March 2, 2012).

Kahn, R., & Wilensky, R. (1995). A framework for distributed digital object services. International Journal on Digital Libraries, 6(2), 115-123.

Kim, I., & Kuljis, J. (2010, June). Applying content analysis to web based content. Proceedings of the ITI 2010 32nd International conference on

information technology interfaces, June 21-24, 2010, Cavtat, Croatia.Retrieved from http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=5546426 (accessed May, 2014)

Karmakar, G. S., Das, R., & Thakuria, J. (2010, February 18). Institutional repositories in India: A comparative study of Dspace and Eprints. Retrieved fromhttp://ir.inflibnet.ac.in/bitstream/1944/983/1/43.pdf (accessed July 5, 2012).

Khode, S., & Chandel, S. S. (2015). Adoption of open source software in India. DESIDOC Journal of Library & Information Technology, 35(1) p. 30-40.

Khondhu, J., Capiluppi, A., &Stol, K. J. (2013). Is it all lost? A study of inactive open source projects. In Etiel P., Giancarlo, S., Nabil, E. I. & Alberto, S. (Eds.), Open source software: quality verification (pp. 61-79). Heidelberg: Springer.

Krishnamurthy, M. (2008). Open access, open source and digital libraries: A current trend in university libraries around the world. Program: Electronic Library and Information Systems, 42(1), 48-55.

Krishnamurthy, S. (2003, February). An analysis of open source business models. Retrieved from http://faculty. washington.edu/sandeep/d/bazaar.pdf (accessed June 6, 2014).

Kumar, V. (2009). Comparative evaluation of open source digital library packages. Retrieved from https://drtc.isibang.ac.in/ bitstream/handle/1849/441/comparative_evaluation_DL_vinit.pdf?sequence =1(accessed May 6, 2010).

Lam, K. T., & Chan, D. L. (2007). Building an institutional repository: Sharing experiences at the HKUST Library. OCLC systems & services, 23(3), 310-323.

Lawal, I. (2009). Library and information science research in the 21st century: A guide for practicing librarians and students. Oxford : Chandos.

Laxminarsaiah, A., & Rajgoli, I. U. (2007). Building institutional repository: An overview. OCLC Systems & Services, 23(3), 278-286.

Li, A. (2008). Investigation of ILS in Chinese main academic libraries. Library Collections, Acquisitions, and Technical Services, 32(3), 115-120.

Li, Y., Tan, C. H., & Yang, X. (2013). It is all about what we have: A discriminant analysis of organizations' decision to adopt open source software. Decision Support Systems, 56, 56-62.

Lihitkar, S. R., & Lihitkar, R. S. (2012). Open source software for developing digital library: Comparative study. DESIDOC Journal of Library & Information Technology, 32(5), 393-400.

Lundell, B., Lings, B., & Syberfeldt, A. (2011). Practitioner perceptions of open source software in the embedded systems area. Journal of Systems and Software, 84(9), 1540-1549.

Lynch, C. A. (2003). Institutional repositories: Essential infrastructure for scholarship in the digital age. Portal: Libraries and the Academy, 3(2), 327-336.

Macan, B., Vanesa Fernández, G., & Stojanovski, J. (2013). Open source solutions for libraries: ABCD vs Koha. Program, 47(2), 136-154.

Madalli, D. P. (2003). A digital library of library and information science using DSpace, paper presented at the DRTC Workshop on the Semantic Web, March 23-25, Bangalore Retrieved from https://drtc.isibang.ac.in/handle/1849/56 (accessed November 2, 2004).

Madalli, D. P., Barve , S. and Amin, S.(2012). Digital preservation in open-source digital library software. The Journal of Academic Librarianship 38(3), P. 161–164.

Marchionini, G. and Fox, E.A. (1999). Progress toward digital libraries: Augmentation through integration. Information Processing & Management, 35 (3), 219-225.

Marsan, J., & Paré, G. (2013). Antecedents of open source software adoption in health care organizations: A qualitative survey of experts in Canada. International Journal of Medical Informatics, 82(8), 731-741.

Marsan, J., Paré, G., & Beaudry, A. (2012). Adoption of open source software in organizations: A socio-cognitive perspective. The Journal of Strategic Information Systems, 21(4), 257-273.

Masrek, M. N., & Hakimjavadi, H. (2012). Evaluation of three open source software in terms of managing repositories of electronic theses and dissertations: A comparison study. Journal of Basic and Applied Scientific Research, 2(11), 10843-10852.

Meeker, H. J. (2008). The open source alternative: Understanding risks and leveraging opportunities. New Jersey: John Wiley & Sons.

Midha, V., & Palvia, P. (2012). Factors affecting the success of open source software. Journal of Systems and Software, 85(4), 895-905.

Mijinyawa, K. (2008). An evaluation of open source software adoption by UK SMEs in the IT industry (Doctoral dissertation). Retrieved from http://bura.brunel.ac.uk/bitstream/2438/4509/1/FulltextThesis.pdf (accessed June 5, 2013).

MITRE (2011). Open source software, Systems engineering guide. Retrieved from: http://www.mitre.org/publications/systems-engineering-guide/enterprise-engineering/engineering-informationintensive-enterprises/open-source-software (accessed June 8, 2014).

Mittal, R., & Mahesh, G. (2008). Digital libraries and repositories in India: an evaluative study. Program: Electronic Library and Information Systems, 42(3), 286-302.

Mockus, A., Fielding, R. T., & Herbsleb, J. D. (2002). Two case studies of open source software development: Apache and Mozilla. ACM Transactions on Software Engineering and Methodology (TOSEM), 11(3), 309-346.

Morgan, E. L. (2002). Possibilities for open source software in libraries. Information Technology and Libraries, 21(1), 12.

Mouly, G. J. (1963). Educational research: The art and science of investigation. Boston: Allyn and Bacon.

Mtsweni, J., & Biermann, E. (2008, October). An investigation into the implementation of open source software within the SA government: An emerging expansion model. In Proceedings of the 2008 annual research conference of the South African Institute of Computer Scientists and Information Technologists on IT research in developing countries: riding the wave of technology (pp. 148-158). ACM.

Müller, T. (2011). How to choose a free and open source integrated library system. OCLC Systems & Services, 27(1), 57-78.

Mutula, S., & Kalaote, T. (2010). Open source software deployment in the public sector: a review of Botswana and South Africa. Library Hi Tech, 28(1), 63-80.

Nagy, D., Yassin, A.M. & Bhattacherjee, A. (2010). Organizational adoption of open source software: Barriers and remedies. Communications of the ACM, 53(3), 148-151.

Nash, J. C. (2010). Directions for open source software over the next decade. Futures, 42(4), 427-433.

NRCFOSS (2013). Open source maturity model for a department/enterprise/institution. Retrieved from ttp://www.nrcfoss.au-kbc.org.in/maturity/ (accessed June 7, 2014).

O'Neill, A. (2012), Open source software. In Ruth Chadwick (Ed.) Encyclopedia of applied ethics. (2nd ed.). (pp.281-287) Retrieved from http://www.sciencedirect.com/ science/article/pii/B9780123739322000636 (accessed June 5, 2014).

Pappalardo, K. M. & Fitzgerald, A. M. (2007). A guide to developing open access through your digital repository. Retrieved from http://eprints.qut.edu.au/9671/1/9671.pdf (accessed May12, 2013).

Paulk, M.C, Curtis M, Chrissis, B. & Weber, C. V. (1993). Capability maturity model for software, version 1.1. Technical report. Retrieved from https://resources.sei.cmu.edu/ asset_files/TechnicalReport/1993_005_001_16211.pdf (accessed June 7, 2014).

Pomerantz, J., & Marchionini, G. (2007). The digital library as place. Journal of Documentation, 63(4), 505-533.

Poulter, A. (2010). Open source in libraries: An introduction and overview, Library Review, 59(9), 655-661.

Pressman, R.S. (2005). Software engineering: A practitioner's approach. (6th ed.). New York: McGraw-Hill.

Randhawa, S. (2012). Open source software for creation of digital library: A comparative study of Greenstone digital library software & DSpace. Indian Journal of Library and Information Science, 6(3), 42-52.

Ratib, O. & Rosset, A. (2006). Open-source software in medical imaging: Development of OsiriX. International Journal of Computer Assisted Radiology and Surgery, 1(4), 187-196.

Raymond, E.S. (2001) Cathedral and the bazar: Musings on Linux and open source by an accidental revolutionary. (Rev.ed.) CA: OReilly Media, Inc.

Raza, A., Capretz, L. F., & Ahmed, F. (2012). An open source usability maturity model (OS-UMM). Computers in Human Behavior, 28(4), 1109-1121.

Research. (n.d.). In Merriam-Webster's online dictionary. Retrieved from http://www.merriam-webster.com/dictionary/research (accessed June 2, 2014).

Rossi, B., Russo, B., & Succi, G. (2012). Adoption of free/libre open source software in public organizations: Factors of impact. Information Technology & People, 25(2), 156-187.

Rudestam, K. E., & Newton, R. R. (2014). Surviving your dissertation: A comprehensive guide to content and process. California : Sage Publications.

Salah, D., Paige, R., & Cairns, P. (2014). An evaluation template for expert review of maturity models. In Andreas, J., Pasi, K., Marco, K., Tomi, M. J. M. &Mikko, R. (Eds.) Product-focused software process improvement (pp. 318-321). Switzerland: Springer International Publishing.

Sawant, S. (2011). IR system and features: Study of Indian scenario. Library Hi Tech, 29(1), 161-172.

Schneider, K. G. (2009). The thick of the fray: Open source software in libraries in the first decade of this century. Bulletin of the American Society for Information Science and Technology, 35(2), 15-19.

Seadle, M., & Greifeneder, E. (2007). Defining a digital library. Library Hi Tech, 25(2), 169-173.

Sen, R., Singh, S. S., & Borle, S. (2012). Open source software success: Measures and analysis. Decision Support Systems, 52(2), 364-372.

Shaikh, M. & Cornford, T. (2011, November). Total cost of ownership of open source software: A report for the UK Cabinet Office supported by open forum Europe. Retrieved from http://eprints.lse.ac.uk/39826/1/Total_cost_of_ownership_of_ open_source_software_%28LSERO%29.pdf. (accessed June 7, 2014).

Sharma, A., & Adkins, R. (2005). OSS in India. In Chris DiBona, Mark Stone, Danese Cooper (Eds.), Open Sources 2.0: The Continuing Evolution (pp.189-198), CA : OReilly Media.

Shihab, E., Bettenburg, N., Adams, B., & Hassan, A. E. (2010). On the central role of mailing lists in open source projects: An exploratory study. In Nakakoji, K., Murakami, Y., & McCready, E. (Eds.). New Frontiers in Artificial Intelligence: JSAI-isAI 2009 Workshops, LENLS, JURISIN, KCSD,

LLLL, Tokyo, Japan, November 19-20, 2009, Revised Selected Papers (Vol. 6284) (pp. 91-103). Heidelberg: Springer.

Shoeb, M. Z. H. (2010). Developing an institutional repository at a private university in Bangladesh. OCLC Systems & Services, 26(3), 198-213.

Singh, V. (2013). Experiences of migrating to open source integrated library systems. Information Technology and Libraries, 32(1), 36-53.

Smith, P. L. (2008). Where IR you? Using "open access" to extend the reach and richness of faculty research within a university. OCLC Systems & Services: International Digital Library Perspectives, 24(3), 174-184.

Smith, R. (2002). Adapting a new technology to the academic medical library: Personal digital assistants. Journal of the Medical Library Association, 90(1), 93-94.

Spinellis, D., & Giannikas, V. (2012). Organizational adoption of open source software. Journal of Systems and Software, 85(3), 666-682.

Spiro, L. (2009, January). Archival management software. Retrieved from http://www.clir.org/pubs/reports/spiro/ spiro_Jan13.pdf (accessed March 5, 2012).

Sourceforgnet (2014) Retrieved from http://sourceforge.net/ (accessed June 1, 2014).

Stahl, M. T. (2005). Open-source software: Not quite endsville. Drug Discovery Today, 10(3), 219-222.

Stake, R. E., & , R. (1995). The art of case study research. Thousand Oaks, CA: Sage publications.

Stol, K. J., & Babar, M. A. (2010). A comparison framework for open source software evaluation methods. In Pär J. Ågerfalk, Cornelia Boldyreff, Jesús M. González-Barahona, Gregory R. Madey, John Noll(Eds.)Open source software: New horizons (pp. 389-394). Heidelberg: Springer.

Subramaniam, C., Sen, R., & Nelson, M. L. (2009). Determinants of open source software project success: A longitudinal study. Decision Support Systems, 46(2), 576-585.

Sutradhar, B. (2006). Design and development of an institutional repository at the Indian Institute of Technology Kharagpur. Program: Electronic Library and Information Systems, 40(3), 244-255.

Swanson, R. A., & Holton, E. F. (2005). Research in organizations: Foundations and methods in inquiry, Oakland, CA: Berrett-Koehler.

Tansley, R (2003, August) DSpace: Preserving digital data for the ages Retrieved from http://www.hpl.hp.com/news/2003/ july_sept/dspace.html (accessed July 8, 2013).

Tennant, R. (2000, January). The role of open source software. Library Journal, 125(1), 36. Retrieved from http://search.proquest.com/abicomplete/docview/196834140/fulltextPDF/B6DB42E755B44E0PQ/5?accountid=39226 (accessed June 7, 2014).

The concise Oxford English dictionary (2011). New Delhi: Oxford University Press.

The Oxford Dictionary of English (3rd ed.) (2010). New Delhi: Oxford University Press.

The New International Webster's Comprehensive Dictionary of the English Language. (1996) Naples: Trident Press International.

Tramboo, S., Shafi, S. M., & Gul, S. (2012). A study on the open source digital library software's: Special reference to DSpace, EPrints and Greenstone. International Journal of Computer Applications 59(16), 1-9. Retrieved from http://arxiv.org/ftp/arxiv/papers/1212/1212.4935.pdf (accessed June 9, 2014).

UNESCO (2010). Free and open source software. Retrieved from http://www.unesco.org/new/en/communication-and-information/access-to-knowledge/free-and-open-source-software-foss/browse/5/ (accessed May 2, 2014).

Välimäki, M., & Oksanen, V. (2005). The impact of free and open source licensing on operating system software markets. Telematics and Informatics, 22(1), 97-110.

Venkatesh, V., & Davis, F. D. (2000). A theoretical extension of the technology acceptance model: Four longitudinal field studies. Management Science, 46(2), 186-204.

Vimal Kumar, V., & Jasimudeen, S. (2012). Adoption and user perceptions of KOHA library management system in India. Annals of Library and Information Studies, 59(4), 223-230.

Von Hippel, E. (2001). Learning from open-source software. MIT Sloan Management Review, 42(4), 82-86.

Winter, M., & Bowen-Chang, P. (2010). Dealing with DSpace: The experience at the university of the West Indies, St Augustine. New Library World, 111(7/8), 320-332.

Woods, D., & Guliani, G. (2005). Open source for the enterprise: Managing risks, reaping rewards. New Delhi: O'Reilly Media.

Woodside, A. G. (2010). Case study research: theory, methods and practice. Bingley: Emerald.

Yeates, S. (2005, November 1). Documentation issues in open source, OSS Watch. Retrieved from http://oss-atch.ac.uk/resources/archived/documentation (accessed June 7, 2014).

Yin, R. K. (1994). Case study research: Design and methods. (2nded.). CA: Sage Publications.

Young, B. (2001). Forward. In E.S. Raymond, Cathedral and the bazar: Musings on Linux and open source by an accidental revolutionary, (Rev.ed.) CA: OReilly Media, Inc.

Yuan, E.K.K. (2009, January). The adoption of open source software by Singaporean companies. Retrieved from http://eprints.qut.edu.au/20496/1/Edmund_Koh_Ker_Yuan_Thesis.pdf (accessed May 8, 2010).

Zhang, D. (2007, August). Open source software maturity model based on linear regression and bayesian analysis. (Doctoral dissertation) Retrieved from http://repository.tamu.edu/ bitstream/handle/1969.1/ETD-TAMU-1454/ZHANG-DISSERTATION.pdf?sequence=1(accessed June 8, 2014).

Appendix A Significant Scholarly Contributions of DSpace Team Leaders

Tansley, R., Bass, M., Stuve, D., Branschofsky, M., Chudnov, D., McClellan, G., & Smith, M. (2003, May). The DSpace institutional digital repository system: Current functionality. In Proceedings of the 3rd ACM/IEEE-CS joint conference on Digital libraries (pp. 87-97). IEEE Computer Society.

Tansley, R., Bass, M., & Smith, M. (2003). DSpace as an open archival information system: Current status and future directions. In Research and advanced technology for digital libraries (pp. 446-460). Springer Berlin Heidelberg.

Tansley, R., Smith, M., & Walker, J. H. (2005). The DSpace open source digital asset management system: challenges and opportunities. In Research and Advanced Technology for Digital Libraries (pp. 242-253). Springer Berlin Heidelberg.

Smith, M., Rodgers, R., Walker, J., &Tansley, R. (2004). DSpace: A year in the life of an open source digital repository system. In Research and Advanced Technology for Digital Libraries (pp. 38-44). Springer Berlin Heidelberg.

Tansley, R., Bass, M., Branschofsky, M., Carpenter, G., McClellan, G., &Stuve, D. (2006). DSpace System Documentation. Retrieved July, 4, 2007.

Tansley, R., Bass, M., Stuve, D., Branschofsky, M., Chudnov, D., McClellan, G., & Smith, M. (2003, May). The DSpace institutional digital repository system: current functionality. In Proceedings of the 3rd ACM/IEEE-CS joint conference on Digital libraries (pp. 87-97). IEEE Computer Society.

Smith, M. (2002). DSpace: an institutional repository from the MIT Libraries and Hewlett Packard Laboratories. In Research and Advanced Technology for Digital Libraries (pp. 543-549). Springer Berlin Heidelberg.

Reilly, W., Wolfe, R., & Smith, M. (2006). MIT's CWSpace project: packaging metadata for archiving educational content in DSpace. International Journal on Digital Libraries, 6(2), 139-147.

Bass, M. J., Stuve, D., Tansley, R., Branschofsky, M., Breton, P., Carmichael, P., & Ng, J. (2002). DSpace–A sustainable solution for institutional digital asset services–spanning the information asset value chain: ingest, manage, preserve, disseminate. Internal reference specification: Functionality. Retrieved February, 9, 2003.

Witten, I. H., Bainbridge, D., Tansley, R., Huang, C. Y., Don, K., & Hamilton, N. Z. (2005). A bridge between Greenstone and DSpace. D-Lib magazine, 11(9), 1082-9873.

Bass, M. J., Stuve, D., Tansley, R., Branschofsky, M., Breton, P., Carmichael, P., & Ng, J. (2002). DSpace--Internal Reference Specification: functionality. Retrieved July, 21, 2003.

Bass, M. J., Stuve, D., Tansley, R., Branschofsky, M., Breton, P., Carmichael, P., & Ng, J. (2002). DSpace Technology and Architecture. Internal Reference Specification. Hewlett-Packard Company and Massachusetts Institute of Technology.

Tansley, R. (2007, October). DSpace Architecture: an update. In Proc. DSpace User Group Meeting.─ Rome, Italy.─ Oct (Vol. 2007).

Appendix B List of Papers Published

International Journals

1. Cherukodan, S., Santhosh Kumar, G., & Humayoon Kabir, S. (2013). Using open source software for digital libraries: A case study of CUSAT. The Electronic Library, 31(2), 217-225.

2. Cherukodan, S., & Humayoon Kabir, S. Adoption of DSpace by Indian Libraries (Submitted to Program: Electronic Library and Information Systems (Emerald insight) On 24th August 2014. Status- accept).

International Conferences

3. Sheeja, N.K and Cherukodan, S. (2011) The Development and promotion of ETDs in Kerala. Proceedings of 8th International CALIBER: Towards Building a Knowledge Society: Library as Catalyst for Knowledge Discovery and Management, Goa University, 2-4 March 2011pp. 156-163. Ahmadabad : INFLIBNET.

4. Cherukodan, S., & Humayoon Kabir, S (2011) Using open source for digital repositories in India. Proceedings of 8th International CALIBER: Towards Building a Knowledge Society: Library as Catalyst for Knowledge

Discovery and Management, Goa University, 2-4 March 2011 (Abstract). Ahmadabad: INFLIBNET.

5. Cherukodan, S. & Sheeja N.K (2010) Archiving news in a university using open source software. Proceedings of IFLA International Newspaper Conference 2010 on Digital Preservation and Access to News and Views, February 25-28, 2010, pp.273-279 New Delhi: IFLA.

National Conference

6. Cherukodan, S. & Santhosh Kumar, G. (2011). CUSAT Digital Library: Issues and strategies. Proceedings National Seminar at University of Calicut, Library 2.0 and Information Management. pp. New Delhi: Atlantic.

7. Cherukodan S. & Humayoon Kabir, S.(2011) Making e-contents in-house: some strategies. In Proceedings of KLA National Seminar, Thiruvanadapuram 14-16 July 2011.

8.Cherukodan, S. & Sheeja N.K A Digital Library Proposal for Engineering Libraries, Proceedings of PLANNER 2010 on Re-engineering of Library and Information Services in Digital Era, Tezpur University Assam.February 18-20, 2010, pp.414-418. Ahmadabad: INFLIBNET.

9. Sheeja N K, Mathew, S.K. & Cherukodan S. (2015). Software for managing digital repositories in the world. Paper presented in National Conference on Knowledge Discovery and Management (NCKDM) 30–31 January 2015 , at University of Calicut, 2015.

Book Chapter

10. Cherukodan, S., & Humayoon Kabir, S. (2015). Adoption and visibility of digital repositories: a study of DSpace in India.In Devarajan, G., Abdul Majeed, K. C., Dineshan Koovakkai, Suresh Kumar, P. K., & Beena, S. (Eds.). (2015). Blended libraries and information centres: A blueprint for the development of information profession in India: Essays in honour of Dr. K. P. Vijayakumar. Trivandrum: Southern Bookstar.